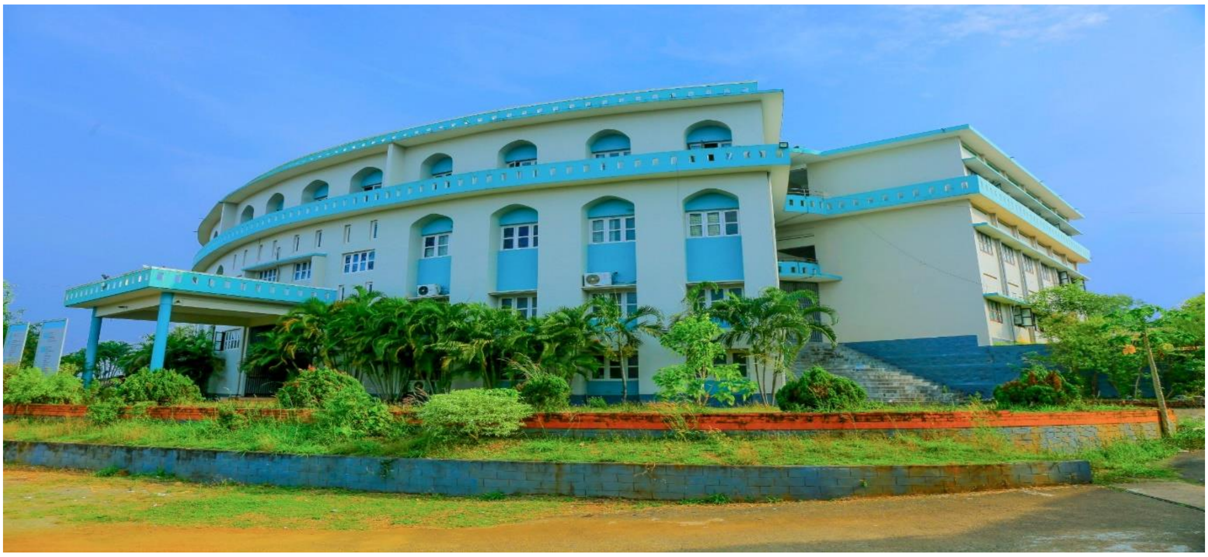
LEARNING OUTCOMES-BASED CURRICULUM FRAMEWORK (LOCF) FOR UNDERGRADUATE EDUCATION

B.Sc. COMPUTER SCIENCE

DEPARTMENT OF E COMPUTER SCIENCE



EMEA College of Arts and Science, Kondotty

Kumminiparamaba PO, Malappuram Dt. Kerala – India PIN :673638; Aided by
Govt. of Kerala & Affiliated to the University of Calicut,
Re-accredited with A Grade (3.13 CGPA)
Website: https://emeacollege.ac.in/ |
E-mail: mail@emeacollege.ac.in |
Phone: 0483 2712030

CONTENTS

INTRODUCTION

EMEA College of Arts and Science, Kondotty, is fast emerging as a resourceful destination for higher studies in Malabar, spreading the fragrance of education in the society. The college offers up-to-date, advanced, and job-oriented programmes in the vast expanding horizon of humanities, commerce, and science and technology. The college – affiliated to the University of Calicut, is dedicated to nurturing academic excellence, fostering a culture of research and innovation, and promoting community engagement. Established with a commitment to high-quality education and holistic development, the College aligns its programs with the Learning Outcomes-Based Curriculum Framework (LOCF), ensuring that students acquire not only subject expertise but also skills relevant to real-world applications.

From an LOCF perspective, EMEA College's curriculum prioritizes outcome-based learning, aiming to produce graduates equipped with critical thinking, effective communication, cultural sensitivity, and social responsibility. The College's pedagogical approach integrates both theoretical knowledge and practical experience, creating a learning environment that responds to the dynamic needs of today's society. By mapping program outcomes (POs) and course outcomes (COs) in alignment with UGC's LOCF guidelines, EMEA College ensures that each course contributes to a cohesive learning journey that enhances employability, research capability, and lifelong learning.

In its pursuit of excellence, EMEA College fosters a supportive academic community that encourages students to engage deeply with their disciplines, appreciate diverse perspectives, and contribute meaningfully to their communities. Through this LOCF-aligned curriculum, the College aims to prepare its graduates to meet global challenges while remaining rooted in local values and responsibilities.

VISION AND MISSION OF THE COLLEGE

Vision

EMEA College envisions creating a transformative educational environment that inspires personal growth, social responsibility, and academic excellence. The College aims to become a beacon of higher learning that empowers students to lead meaningful lives, equipped with the knowledge and skills to contribute positively to society.

Mission

Identifying and developing the talent of the youth and moulding them into useful citizens with due emphasize on right character formation is the avowed mission of EMEA College. The fulfilment of this lofty goal is the basis of educational programes formulated and pursued by the institution. The mission of EMEA College of Arts and Science includes the following core objectives:

1. **Quality Education**: To provide high-quality, inclusive education that fosters intellectual and personal growth, enabling students to reach their fullest potential.
2. **Social Responsibility**: To cultivate a sense of responsibility toward the community, encouraging students to engage in social initiatives and contribute to societal well-being.
3. **Research and Innovation**: To promote a culture of research and innovation, encouraging critical inquiry, creative problem-solving, and continuous learning.
4. **Skill Development**: To equip students with essential life skills and competencies that enhance their employability and adaptability in a dynamic global environment.
5. **Community Empowerment**: To support the development of the local community through outreach and extension activities, addressing social and economic challenges.
6. **Sustainable Practices**: To foster sustainability and inclusivity within the College, embracing practices that promote environmental consciousness and ethical responsibility.

VISION AND MISSION OF THE DEPARTMENT OF COMPUTER SCIENCE

Vision

To be a leader in computing education and research, fostering innovation and advancing technology for societal benefit. The department aspires to produce highly skilled professionals with a commitment to lifelong learning, ethical values, and the ability to adapt to emerging trends in computer science.

Mission of the Department of Computer Science

1. **Provide Quality Education**: To deliver a high-quality, innovative curriculum that balances theoretical foundations and practical applications, enabling students to develop the analytical, technical, and problem-solving skills essential in computing.

2. **Promote Research and Development**: To encourage faculty and students to engage in cutting-edge research, fostering an environment that supports creativity, critical thinking, and the pursuit of knowledge to address real-world challenges.
3. **Industry Collaboration and Skill Development**: To build partnerships with industry leaders, ensuring that students gain exposure to contemporary technologies and acquire relevant skills, preparing them to excel in professional environments.
4. **Ethical and Inclusive Learning**: To cultivate an inclusive environment that values ethical practices, social responsibility, and respect for diversity, helping students become conscientious contributors to society.
5. **Lifelong Learning and Adaptability**: To instil a passion for lifelong learning, empowering students to continuously update their knowledge and adapt to evolving technologies and industry demands.

Core Values

The Department of Computer Science upholds the following core values:

Core values are fundamental beliefs or guiding principles that represent the heart of an organization, institution, or department. They reflect what is important to the organization and guide both its internal conduct and its relationship with the external community. For a Computer Science department, typical core values might include:

1. **Integrity** – Commitment to honesty, transparency, and ethical conduct in all interactions and endeavors.
2. **Innovation** – Encouraging creative thinking and development of new technologies to address real-world challenges.
3. **Excellence** – Striving for the highest quality in teaching, research, and service, with a focus on continuous improvement.
4. **Collaboration** – Emphasizing teamwork, both within the department and with external partners, to foster a supportive learning environment.
5. **Lifelong Learning** – Promoting a culture where continuous learning is valued, enabling students and faculty to stay current in a rapidly evolving field.
6. **Diversity and Inclusion** – Committing to a respectful, inclusive environment where diverse perspectives are valued and everyone has equal opportunities to succeed.

These core values help shape the mission and vision of the department, driving its goals and the learning experience for students.

INTRODUCTION TO THE LEARNING OUTCOMES-BASED CURRICULUM FRAMEWORK (LOCF) FOR THE B.Sc COMPUTER SCIENCE PROGRAM

The **Learning Outcomes-Based Curriculum Framework (LOCF)** for the B.Sc Computer Science Programme at EMEA College of Arts and Science, Kondotty, is designed to align with the guidelines set by the University Grants Commission (UGC) and reflects the institution's commitment to outcome-driven education. This framework emphasizes a student-centered approach, where learning outcomes define the competencies, skills, and values students are expected to achieve upon completing the program.

The **LOCF** for the B.Sc. Computer Science program focuses on developing essential knowledge, skills, and competencies in students to meet both academic and industry needs. This curriculum emphasizes a comprehensive foundation in computer science, covering principles of programming, data structures, and system design while also fostering computational thinking and analytical skills. It prepares students for industry demands with a focus on professional standards, ethical responsibilities, and problem-solving proficiency. The LOCF further encourages adaptability to technological advancements, promoting lifelong learning and innovation. By focusing on these core areas, the program aims to produce graduates equipped with the technical, ethical, and logical skills necessary for success in diverse computer science fields.

By mapping Programme Outcomes (POs) and Course Outcomes (COs) for each course, the LOCF framework provides a clear structure for educational goals, teaching methodologies, and assessment strategies, enhancing coherence and relevance across the curriculum. Ultimately, it aims to produce well-rounded graduates prepared for various professional and academic opportunities, enabling them to contribute meaningfully to society.

GRADUATE ATTRIBUTES FOR THE B.Sc. COMPUTER SCIENCE PROGRAMME

The **Graduate Attributes for the B.Sc. Computer Science Programme** emphasize essential, specific skills and values to be developed by students:

1. **In-depth Technical Knowledge**: Students gain a comprehensive understanding of computer science fundamentals, including software development, database management, networks, and algorithms, ensuring they are technically proficient.

2. **Innovative Problem-Solving**: Graduates acquire advanced analytical skills that empower them to design, implement, and refine solutions for complex computing challenges through creative and efficient coding practices.
3. **Ethical and Social Responsibility**: The curriculum instills a strong sense of ethics, encouraging students to approach technological advancement responsibly, with an awareness of societal impact and privacy concerns.
4. **Adaptability to Technological Change**: The program nurtures flexibility, equipping graduates to adapt and respond effectively to the fast-evolving tech landscape and engage in continuous professional development.
5. **Effective Communication and Collaboration**: Students learn to convey technical information effectively and work cohesively in multidisciplinary teams, preparing them for collaborative industry environments.
6. **Global and Cultural Awareness**: Emphasizing the global nature of technology, the program encourages students to understand diverse cultural perspectives, aligning their work with international standards and addressing global issues responsibly. These attributes ensure that graduates are not only prepared to succeed in their careers but also contribute meaningfully to the tech industry and society at large.

GRADUATE ATTRIBUTES – DEPARTMENT OF COMPUTER SCIENCE

The **Graduate Attributes for the Department of Computer Science** focus on developing students into well-rounded, competent professionals with a strong foundation in both technical skills and personal values. Here’s a summary of key attributes fostered within a Computer Science program:

1. **Technical Expertise**: Graduates are expected to have a solid foundation in core computer science areas like programming, algorithms, data structures, and systems architecture, preparing them to handle diverse computational tasks effectively.
2. **Problem-Solving Ability**: The program encourages critical thinking and innovative problem-solving skills, enabling graduates to tackle complex computing challenges and design efficient, practical solutions.

3. **Ethical Responsibility**: Emphasis is placed on ethics and social responsibility, ensuring that graduates understand the broader impact of technology and act responsibly in their professional roles.
4. **Adaptability and Lifelong Learning**: Given the dynamic nature of technology, graduates are taught to be adaptable and to value continuous learning, making them capable of evolving with emerging trends and technologies.
5. **Communication and Teamwork**: Strong interpersonal and communication skills are developed, allowing graduates to work effectively within teams, present ideas clearly, and collaborate across disciplines.
6. **Global Awareness**: Graduates gain an appreciation for the global implications of technology, fostering a mindset of inclusivity and cultural sensitivity in their work.

These attributes ensure that Computer Science graduates are prepared to meet industry demands, contribute meaningfully to society, and lead with integrity in their professional journeys.

PROGRAMME OUTCOME	
PROGRAMME – B.Sc. COMPUTER SCIENCE	
PO1	Knowledge Acquisition: Demonstrate a profound understanding of knowledge trends and their impact on the chosen discipline of study.
PO2	Communication, Collaboration, Inclusiveness, and Leadership: Become a team player who drives positive change through effective communication, collaborative acumen, transformative leadership, and a dedication to inclusivity.
PO3	Professional Skills: Demonstrate professional skills to navigate diverse career paths with confidence and adaptability.
PO4	Digital Intelligence: Demonstrate proficiency in varied digital and technological tools to understand and interact with the digital world, thus effectively processing complex information.
P05	Scientific Awareness and Critical Thinking: Emerge as an innovative problem-solver and impactful mediator, applying scientific understanding and critical thinking to address challenges and advance sustainable solutions.
PO6	Human Values, Professional Ethics, and Societal and Environmental Responsibility: Become a responsible leader , characterized by an unwavering commitment to human values , ethical conduct, and a fervent dedication to the well-being of society and the environment.

MINIMUM CREDIT REQUIREMENTS OF THE DIFFERENT PATHWAYS IN THE THREE-YEAR PROGRAMME IN CUFYUGP

Sl. No.	Academic Pathway	Major	Minor/ Other Disciplines	Foundation Courses AEC: 4 MDC: 3 SEC: 3 VAC: 3	Intern -ship	Total Credits	Example
		Each course has 4 credits		Each course has 3 credits			
1	Single Major (A)	68 (17 courses)	24 (6 courses)	39 (13 courses)	2	133	Major: Computer Science + six courses in different disciplines in different combinations
2	Major (A) with Multiple Disciplines (B, C)	68 (17 courses)	12 + 12 (3 + 3 = 6 courses)	39 (13 courses)	2	133	Major: Computer Science + Mathematics and Physics
3	Major (A) with Minor (B)	68 (17 courses)	24 (6 courses)	39 (13 courses)	2	133	Major: Computer Science Minor: Electronics
4	Major (A) with Vocational Minor (B)	68 (17 courses)	24 (6 courses)	39 (13 courses)	2	133	Major: Computer Science Minor: Data Science/Web Technology
5	Double Major (A, B)	A: 48 (12 courses) B: 44 (11 courses)	-	12 + 18 + 9	2	133	Computer Science and Statistics Double Major
			The 24 credits in the Minor stream are distributed between the two Majors. 2 MDC, 2 SEC, 2 VAC and the Internship should be in Major A. Total credits in Major A should be 48 + 20 = 68 (50% of 133) 1 MDC, 1 SEC and 1 VAC should be in Major B. Total credits in Major B should be 44 + 9 = 53 (40% of 133)				
Exit with UG degree / Proceed to fourth year with 133 credits							

B.Sc. COMPUTER SCIENCE HONOURS PROGRAMME

COURSE STRUCTURE FOR PATHWAYS 1 – 4

1. Single Major
2. Major with Multiple Disciplines
3. Major with Minor
4. Major with Vocational Minor

Semester	Course Code	Course Title	Total Hours	Hours/ Week	Credits	Marks		
						Internal	External	Total
1	CSC1CJ101/ CSC1MN100	Fundamentals of Computers & Computational Thinking	75	5	4	30	70	100
	XXX1MNXXX	Minor Course 1	75	5	4	30	70	100
	XXX1MNXXX	Minor Course 2	75	5	4	30	70	100
	ENG1FA101 (2)	Ability Enhancement Course 1	60	4	3	25	50	75
	XXX1FA102 (2)	Ability Enhancement Course 2	45	3	3	25	50	75
	XXX1FM105	Multi-Disciplinary Course 1	45	3	3	25	50	75
		Total		**25**	**21**			**525**
2	CSC2CJ101/ CSC2MN100	Fundamentals of Programming(C Language)	75	5	4	30	70	100
	XXX1MNXXX	Minor Course 3	75	5	4	30	70	100
	XXX1MNXXX	Minor Course 4	75	5	4	30	70	100
	ENG2FA103 (2)	Ability Enhancement Course 3	60	4	3	25	50	75
	XXX2FA104 (2)	Ability Enhancement Course 4	45	3	3	25	50	75
	XXX2FM106	Multi-Disciplinary Course 2	45	3	3	25	50	75
		Total		**25**	**21**			**525**
3	CSC3CJ201	Software Project Management	60	4	4	30	70	100
	CSC3CJ202/ CSC3MN200	Data Structures and Algorithm	75	5	4	30	70	100
	XXX1MNXXX	Minor Course 5	75	5	4	30	70	100
	XXX1MNXXX	Minor Course 6	75	5	4	30	70	100
	XXX3FM107 (2)	Multi-Disciplinary Course 3 – Kerala Knowledge System	45	3	3	25	50	75
	ENG3FV108 (2)	Value-Added Course 1	45	3	3	25	50	75
		Total		**25**	**22**			**550**
4	CSC4CJ203	Database Management System	75	5	4	30	70	100
	CSC4CJ204	Python Programming	75	5	4	30	70	100

	CSC4CJ205	Computer Networks	75	5	4	30	70	100
	ENG4FV109 (2)	Value-Added Course 2	45	3	3	25	50	75
	XXX4FV110(2)	Value-Added Course 3	45	3	3	25	50	75
	ENG4FS111 (2)	Skill Enhancement Course – 1 (P)	60	4	3	25	50	75
		Total		**25**	**21**			**525**
5	CSC5CJ301	Data Mining	60	4	4	30	70	100
	CSC5CJ302	Object Oriented Programming (Java)	75	5	4	30	70	100
	CSC5CJ303	Full Stack Web Development	75	5	4	30	70	100
	CSC5EJ305	Elective Course 1 in Major	60	4	4	30	70	100
	CSC5EJ306	Elective Course 2 in Major	60	4	4	30	70	100
	CSC5FS112	Skill Enhancement Course 2 - Introduction to Digital Marketing	45	3	3	25	50	75
		Total		**25**	**23**			**575**
6	CSC6CJ304/ CSC8MN304	Digital Electronics and Computer Architecture	60	4	4	30	70	100
	CSC6CJ305/ CSC8MN305	Principles of Operating Systems	75	5	4	30	70	100
	CSC6CJ306/ CSC8MN306	Introduction to Artificial Intelligence & Machine Learning	75	5	4	30	70	100
	CSC6CJ311	Elective Course 3 in Major	60	4	4	30	70	100
	CSC6CJ312	Elective Course 4 in Major	60	4	4	30	70	100
	CSC6FS113	Skill Enhancement Course 3 - Project Implementation	45	3	3	25	50	75
	CSC6CJ349	Internship in Major (Credit for internship to be awarded only at the end of Semester 6)	60		2	50	-	50
		Total		**25**	**25**			**625**
Total Credits for Three Years					**133**			**3325**
7	CSC7CJ401	Theory of Computation	60	4	4	30	70	100
	CSC7CJ402	System Security	60	4	4	30	70	100
	CSC7CJ403	Advanced Data Structures and Algorithms	75	5	4	30	70	100
	CSC7CJ404	Block Chain Technology	60	4	4	30	70	100
	CSC7CJ405	Internet of Things	75	5	4	30	70	100
		Total		**22**	**20**			**500**

8	CSC8CJ406	Compiler Design	60	4	4	30	70	100
	CSC8CJ407	Client-Server Architecture	60	4	4	30	70	100
	CSC8CJ408	Parallel Computing	60	4	4	30	70	100
	OR (instead of Core Courses CSC8CJ406, CSC8CJ407 and CSC8CJ408 in Major)							
	CSC8CJ449**	Project (in Honours programme)	360	13	12	90	210	300
	CSC8CJ499**	Research Project (in Honours with Research programme)	360	13	12	90	210	300
	CSC8EJXXX* /CSC8MN406	Elective Course 5 in Major / Minor Course 7	60	4	4	30	70	100
	CSC8EJXXX* /CSC8MN407	Elective Course 6 in Major / Minor Course 8	60	4	4	30	70	100
	CSCEJXXX* /CSC8MN408	Elective Course 7 in Major / Minor Course 9 / Major Course in any Other Discipline	60	4	4	30	70	100
	OR (instead of Elective Course 7 in Major, in the case of Honours with Research Programme)							
	CSC8CJ 489	Research Methodology	60	4	4	30	70	100
		Total		**25**	**24**			**600**
Total Credits for Four Years					**177**			**4425**

.

Choose any four elective courses (two in fifth and two in sixth semester) from the basket of electives with specialization

Choose three elective courses in semester 8 from elective basket with no specialization

CREDIT DISTRIBUTION FOR PATHWAYS 1 – 4

1. Single Major
2. Major with Multiple Disciplines
3. Major with Minor
4. Major with Vocational Minor

Semester	Major Courses	Minor Courses	General Foundation Courses	Internship/ Project	Total
1	4	4 + 4	3 + 3 + 3	-	21
2	4	4 + 4	3 + 3 + 3	-	21
3	4 + 4	4 + 4	3 + 3	-	22
4	4 + 4 + 4	-	3 + 3 + 3	-	21
5	4 + 4 + 4 + 4 + 4	-	3	-	23
6	4 + 4 + 4 + 4 + 4	-	3	2	25
Total for Three Years	**68**	**24**	**39**	**2**	**133**
7	4 + 4 + 4 + 4 + 4	-	-	-	20
8	4 + 4 + 4	4 + 4 + 4	-	12	24
Total for Four Years	**88 + 12 = 100**	**36**	**39**	**2**	**177**

DISTRIBUTION OF MAJOR COURSES IN COMPUTER SCIENCE FOR PATHWAYS 1 – 4

1. Single Major
2. Major with Multiple Disciplines
3. Major with Minor
4. Major with Vocational Minor

Semester	Course Code	Course Title	Hours/ Week	Credits
1	CSC1CJ101 /CSC1MN100	Fundamentals of Computers & Computational Thinking	5	4
2	CSC2CJ101 /CSC2MN100	Fundamentals of Programming (C Language)	5	4
3	CSC3CJ201	Software Project Management	4	4
	CSC3CJ202 /CSC3MN200	Data Structures and Algorithms	5	4
4	CSC4CJ203	Database Management System	5	4
	CSC4CJ204	Python Programming	5	4

	CSC4CJ205	Computer Networks	5	4
5	CSC5CJ301	Data Mining	4	4
	CSC5CJ302	Object Oriented Programming(Java)	5	4
	CSC5CJ303	Full Stack Development	5	4
	CSC5EJ305	Elective Course 1	4	4
	CSC5EJ306	Elective Course 2	4	4
6	CSC6CJ304 /CSC8MN304	Digital Electronics and Computer Architectures	4	4
	CSC6CJ305 /CSC8MN305	Principles of Operating Systems	5	4
	CSC6CJ306 /CSC8MN306	Introduction to AI and ML	5	4
	CSC6CJ311	Elective Course 3	4	4
	CSC6CJ312	Elective Course 4	4	4
	CSC6CJ349	Internship in Major	-	2
Total for the Three Years				**70**
7	CSC7CJ401	Theory of Computation	4	4
	CSC7CJ402	System Security	4	4
	CSC7CJ403	Advanced Data Structures and Algorithms	5	4
	CSC7CJ404	Blockchain Technology	4	4
	CSC7CJ405	Internet of Things	5	4
8	CSC8CJ406	Compiler Design	4	4
	CSC8CJ407	Client-Server Architecture	4	4
	CSC8CJ408	Parallel Computing	4	4
	OR (instead of Core Courses CSC8CJ406, CSC8CJ407 and CSC8CJ408 in Major)			
	CSC8EJXXX* /CSC8MN406	Elective Course 5	4	4
	CSC8EJXXX * /CSC8MN407	Elective Course 6	4	4
	CSC8EJXXX * /CSC8MN408	Elective Course 7	4	4
	CSC8CJ449**	Project Work (in Honours Programme)/ Project with Research	12	12
	CSC8CJ499**	Research Project (in Honours with Research Programme)		
	OR (instead of Elective Course 7 in Major, in the case of Honours with Research Programme)			

	CSC8CJ 489	Research Methodology	4	4
Total for the Four Years				**114**

ELECTIVE COURSES IN COMPUTER SCIENCE WITH SPECIALISATION

Group No.	Sl. No.	Course Code	Title	Semester	Total Hrs	Hrs/ Week	Credits	Marks		
								Internal	External	Total
1		**DATA SCIENCE**								
	1	CSC5EJ305a	Mathematical and Statistical Foundation for Data Science	5	60	4	4	30	70	100
	2	CSC5EJ306a	Exploratory Data Analysis	5	60	4	4	30	70	100
	3	CSC6CJ311a	Introduction to Data Warehousing and Big Data	6	60	4	4	30	70	100
	4	CSC6CJ312a	Advanced Python for Data Science	6	60	4	4	30	70	100
2		**AI and ML**								
	1	CSC5EJ305b	Machine Learning Algorithms	5	60	4	4	30	70	100
	2	CSC5EJ306b	Knowledge Engineering	5	60	4	4	30	70	100
	3	CSC5EJ311b	Soft Computing	6	60	4	4	30	70	100
	4	CSC5EJ312b	Deep Learning	6	60	4	4	30	70	100
3	**Cloud Computing**									
	1	CSC5EJ305c	Cloud Computing	5	60	4	4	30	70	100
	2	CSC5EJ306c	Security and Privacy in Cloud	5	60	4	4	30	70	100
	3	CSC6CJ311c	Storage Technologies	6	60	4	4	30	70	100
	4	CSC6CJ312c	Virtualization	6	60	4	4	30	70	100

ELECTIVE COURSES IN COMPUTER SCIENCE WITH NO SPECIALISATION

Sl. No.	Course Code	Title	Semester	Total Hrs	Hrs/ Week	Credits	Marks		
							Internal	External	Total
1	CSC8EJ401	Microprocessor and its Applications	8	60	4	4	30	70	100
2	CSC8EJ402	System Software	8	60	4	4	30	70	100
3	CSC8EJ403	Social Network Analysis	8	60	4	4	30	70	100
4	CSC8EJ404	Advanced Distributed Computing	8	60	4	4	30	70	100
5	CSC8EJ405	Cyber Forensic	8	60	4	4	30	70	100
6	CSC8EJ406	Ethical Hacking	8	60	4	4	30	70	100
7	CSC8EJ407	Expert System and fuzzy logic	8	60	4	4	30	70	100

GROUPING OF MINOR COURSES IN COMPUTER SCIENCE

The Minor courses given below should not be offered to students who have taken Computer Science as the Major discipline. They should be offered to students from other major discipline

(Title of the Minor: **COMPUTER SCIENCE**)

Group No.	Sl. No.	Course Code	Title	Semester	Total Hrs	Hrs/ Week	Credits	Marks		
								Internal	External	Total
1	**Foundation of Computer Programming** (preferable for Physic and Electronics students)									
	1	CSC1MN 101	Exploring Computer Basics & Computational Thinking	1	75	5	4	30	70	100

	2	CSC2MN 101	Foundations of C Programming	2	75	5	4	30	70	100
	3	CSC3MN 201	Python Programming	3	75	5	4	30	70	100
2	**Data Science and AI** (preferable for Mathematics and Data Science as complementary course)									
	1	CSC1MN 102	Python Programming	1	75	5	4	30	70	100
	2	CSC2MN 102	Introduction to Data Science	2	75	5	4	30	70	100
	3	CSC3MN 202	Introduction to AI and Machine Learning	3	75	5	4	30	70	100
3	**Data Analysis and Visualization** (preferable for Statistics, Econometrics, and Economics students)									
	1	CSC1MN 103	Data analysis using Spreadsheet	1	75	5	4	30	70	100
	2	CSC2MN 103	Fundamentals of SPSS and R programming	2	75	5	4	30	70	100
	3	CSC3MN 203	Data Visualisation using Python	3	75	5	4	30	70	100
4	**Computing Skills and Programming Fundamentals** (preferable for Chemistry, and Industrial Chemistry students)									
	1	CSC1MN	Computer Essentials with	1	75	5	4	30	70	100

		104	Word Processing & Presentation							
	2	CSC2MN 104	Web Design Trends and Techniques	2	75	5	4	30	70	100
	3	CSC3MN 204	Programming fundamentals using C	3	75	5	4	30	70	100
5	**General Computing Principles** (preferable for Humanities, Commerce, Public Administration, and Travel and tourism students)									
	1	CSC1MN 105	Introduction to IT	1	75	5	4	30	70	100
	2	CSC2MN 105	Efficient Office Dynamics	2	75	5	4	30	70	100
	3	CSC3MN 205	Mastering Content Management Systems	3	75	5	4	30	70	100
6	**Fundamentals of Computer Science** (preferable for Microbiology students)									
	1	CSC1MN 106	Computer Fundamentals with MS Excel,_SPSS	1	75	5	4	30	70	100
	2	CSC2MN 106	Fundamentals of the System software, Networks and DBMS	2	75	5	4	30	70	100
	3	CSC3MN 206	Python Programming	3	75	5	4	30	70	100

7	**General awareness in Computer** (for any sream)									
	1	CSC1MN 107	Computer Hardware Assembly	1	75	5	4	30	70	100
	2	CSC2MN 107	Exploring Cyber security in social media	2	75	5	4	30	70	100
	3	CSC3MN 207	Emerging Trends in Computer Science	3	75	5	4	30	70	100

GROUPING OF VOCATIONAL MINOR COURSES IN COMPUTER SCIENCE

(Title of the Vocational Minor: **DATA SCIENCE)**

Group No.	Sl. No.	Course Code	Title	Semester	Total Hrs	Hrs/ Week	Credits	Marks		
								Internal	External	Total
1		**DATA SCIENCE**								
	1	CSC1VN101	Computational Mathematics in Data Science	1	75	5	4	30	70	100
	2	CSC2VN101	Introduction to Data Science	2	75	5	4	30	70	100
	3	CSC3VN201	Data Analysis and Visualisation Using Spreadsheets	3	75	5	4	30	70	100
	4	CSC8VN401	Predictive Modelling	8	60	4	4	30	70	100

Group No.	Sl. No.	Course Code	Title	Semester	Total Hrs	Hrs/ Week	Credits	Marks		
								Internal	External	Total
2		**Artificial Intelligence**								
	1	CSC1VN102	Statistical Foundations for Artificial Intelligence	1	75	5	4	30	70	100
	2	CSC2VN102	Foundations of	2	75	5	4	30	70	100

			Artificial Intelligence							
	3	CSC3VN202	Automation and Robotics	3	75	5	4	30	70	100
	4	CSC8VN402	Expert Systems and Fuzzy Logic	8	60	4	4	30	70	100

i. Students in Single Major pathway can choose course/courses from any of the Minor/ Vocational Minor groups offered by a discipline other than their Major discipline.

ii. Students in Major with Multiple Disciplines pathway can choose as one of the multiple disciplines, all the three courses from any one of the Minor/ Vocational Minor groups offered by any discipline, other than their Major discipline. If they choose one of the Minor/ Vocational Minor groups offered by their Major discipline as the first one of the multiple disciplines, then their choice as the second one of the multiple disciplines should be any one of the Minor/ Vocational Minor groups offered by a discipline other than the Major discipline. If the students choose any one of the Minor/ Vocational Minor groups in Computer Science as given above, then the title of the group will be the title of that multiple discipline.

(iii). Students in Major with Minor pathway can choose all the courses from any two Minor groups offered by any discipline. If the students choose two Minor groups in Computer Science (three courses from foundations of computing and three courses from foundations of data analytics) as given above, then the title of the Minor will be **Computer science.**

(iv). Students in Major with Vocational Minor pathway can choose all the courses from any two Vocational Minor groups offered by any discipline. If the students choose a Vocational Minor group in Computer Science as given above, then the title of the Vocational Minor will be **Data Science and AI**

DISTRIBUTION OF GENERAL FOUNDATION COURSES IN COMPUTER SCIENCE

Semester	Course Code	Course Title	Total Hours	Hours/ Week	Credits	Marks		
						Internal	External	Total
1	CSC1FM105	Data Analysis and Visualization Through Spread Sheet	45	3	3	25	50	75
2	CSC2FM106	Digital Empowerment Through Ethical Standards	45	3	3	25	50	75
3	CSC3FV108(1)	Introduction to cyber laws	45	3	3	25	50	75
4	CSC4FV109(2)	Introduction to Content Management Systems	45	3	3	25	50	75
5	CSC5FS112	Introduction to Digital Marketing	45	3	3	25	50	75
6	CSC6FS113	Project Implementation	45	3	3	25	50	75

COURSE STRUCTURE FOR BATCH A1(B2)
IN PATHWAY 5: DOUBLE MAJOR

A1: 68 credits in COMPUTER SCIENCE (Major A) ***B1: 68 credits in Major B***

A2: 53 credits in COMPUTER SCIENCE (Major A) ***B2: 53 credits in Major B***

The combinations available to the students: (A1 & B2), (B1 & A2)

Note: Unless the batch is specified, the course is for all the students of the class

Semester	Course Code	Course Title	Total Hours	Hours/ Week	Credits	Marks		
						Internal	External	Total
1	CSC1CJ101 / CSC1MN100	Fundamentals of Computers & Computational Thinking/Minor in Computer Science	75	5	4	30	70	100
	XXX1CJ101	Core Course 1 in Major B –	60/ 75	4/ 5	4	30	70	100
	CSC1CJ102 / CSC2CJ102 / CSC4CJ203*	Database Management System (for batch A1 only)	75	5	4	30	70	100
	ENG1FA101 (2)	Ability Enhancement Course 1	60	4	3	25	50	75
	XXX1FA102(2)	Ability Enhancement Course 2	45	3	3	25	50	75

	CSC1FM105	Multi-Disciplinary Course 1 – Data Analysis and Visualisation Through Spreadsheets	45	3	3	25	50	75
		Total		**24/ 25**	**21**			**525**
2	CSC2CJ101 / CSC2MN100	Fundamentals of Programming Language/ Minor in Computer Science	75	5	4	30	70	100
	XXX2CJ101	Core Course 2 in Major B –	60/ 75	4/ 5	4	30	70	100
	XXX2CJ102 / XXX1CJ102	Core Course 3 in Major B – (for batch B2 only)	60/ 75	4/ 5	4	30	70	100
	ENG2FA103 (2)	Ability Enhancement Course 3	60	4	3	25	50	75
	XXX2FA108(2)	Ability Enhancement Course 4	45	3	3	25	50	75
	CSC2FM106	Multi-Disciplinary Course 2 – Digital Empowerment Through Ethical Standards	45	3	3	25	50	75
		Total		**23 – 25**	**21**			**525**
3	CSC3CJ201	Core Course 4 in Major – Software Project Management	60	4	4	30	70	100
	CSC3CJ202/ CSC3MN200	Core Course 5 in Major – Data Structures and Algorithms	75	5	4	30	70	100
	XXX3CJ201	Core Course 4 in Major B	60/ 75	4/ 5	4	30	70	100
	XXX3CJ202	Core Course 5 in Major B	60/ 75	4/ 5	4	30	70	100
	XXX3FM106	Multi-Disciplinary Course 1 in B –	45	3	3	25	50	75
	CSC3FV108(1)	Value-Added Course Introduction to cyber laws	45	3	3	25	50	75
		Total		**23 – 25**	**22**			**550**
4	CSC4CJ204	Core Course 6 in Major –Python Programming	75	5	4	30	70	100
	XXX4CJXXX	Core Course 6 in Major B	60/ 75	4/ 5	4	30	70	100
	CSC4CJ205	Core Course 7 in Major – Computer networks	75	5	4	30	70	100
	CSC4FV109(2)	Value-Added Course Introduction to content management system	45	3	3	25	50	75
	XXX4FV110	Value-Added Course 1 in B	45	3	3	25	50	75

	CSC4FS112	Skill Enhancement Course Introduction to Digital Marketing	45	3	3	25	50	75
		Total		**23/ 24**	**21**			**525**
5	CSC5CJ302	Core Course 8 in Major – Object Oriented Programming	75	5	4	30	70	100
	XXX5CJXXX	Core Course 7 in Major B –	60/ 75	4/ 5	4	30	70	100
	CSC5CJ303	Core Course 9 in Major – Full Stack Development	60	4	4	30	70	100
		Elective Course 1 in Major	60	4	4	30	70	100
	XXX5CJXXX	Elective Course 1 in Major B	60	4	4	30	70	100
	XXX5FSXXX	Skill Enhancement Course 1 in B	45	3	3	25	50	75
		Total		**24/ 25**	**23**			**575**
6	CSC6CJ305/ CSC8MN305	Core Course 10 in Major – Operating System/minor	75	5	4	30	70	100
	XXX6CJXXX	Core Course 8 in Major B –	60/ 75	4/ 5	4	30	70	100
	XXX6CJXXX	Core Course 9 in Major B – (for batch B2 only)	60	4	4	30	70	100
		Elective Course 2 in Major Computer Science	60	4	4	30	70	100
	XXX6EJXXX	Elective Course 2 in Major B	60	4	4	30	70	100
	CSC6FS113	Skill Enhancement Course 3 – Project Implementation	45	3	3	25	50	75
	CSC6CJ349	Internship in Major Computer Science (Credit for internship to be awarded only at the end of Semester 6)	60		2	50	-	50
		Total		**24/ 25**	**25**			**625**
Total Credits for Three Years					**133**			**3325**

For batch A1(B2), the course structure in semesters 7 and 8 is the same as for pathways 1 – 4, except that the number of the core and elective courses is in continuation of the number of courses in the two categories completed at the end of semester 6.

* The course code of the same course as used for the pathways 1 – 4

CREDIT DISTRIBUTION FOR BATCH A1(B2)
IN PATHWAY 5: DOUBLE MAJOR

Semester	Major Courses in Computer Science	General Foundation Courses in Computer Science	Internship/ Project in Computer Science	Major Courses in B	General Foundation Courses in B	AEC	Total
1	4 + 4	3	-	4	-	3 + 3	21
2	4	3	-	4 + 4	-	3 + 3	21
3	4 + 4	3	-	4 + 4	3	-	22
4	4 + 4	3 + 3	-	4	3	-	21
5	4 + 4 + 4	-	-	4 + 4	3	-	23
6	4 + 4	3	2	4 + 4 + 4	-	-	25
Total for Three Years	**48**	**18**	**2**	**44**	**9**	**12**	**133**
	68			**53**		**12**	**133**
	Major Courses in Computer Science	Minor Courses					
7	4 + 4 + 4 + 4 + 4	-			-	-	20
8	4 + 4 + 4	4 + 4 + 4	12		-	-	24
Total for Four Years	**88 + 12 = 100**	**12**					**177**

COURSE STRUCTURE FOR BATCH B1(A2)
IN PATHWAY 5: DOUBLE MAJOR

A1: 68 credits in Computer Science (Major A) ***B1: 68 credits in Major B***

A2: 53 credits in Computer Science (Major A) ***B2: 53 credits in Major B***

The combinations available to the students: (A1 & B2), (B1 & A2)

Note: Unless the batch is specified, the course is for all the students of the class

Semester	**Course Code**	**Course Title**	**Total Hours**	**Hours/ Week**	**Credits**	**Marks**		
						Internal	**External**	**Total**
1	**XXX**1CJ101	Core Course 1 in Major B –	75	5	4	30	70	100

	CSC1CJ101	Fundamentals of Computers & Computational Thinking	60/ 75	4/ 5	4	30	70	100
	XXX1CJ 102 / XXX2CJ 102	Core Course 2 in Major B – (for batch B1 only)	60/ 75	4/ 5	4	30	70	100
	ENG1FA101(2)	Ability Enhancement Course – 1 (P) (E)	60	4	3	25	50	75
	XXX1FA102(2)	Ability Enhancement Course – 2 (AL)	45	3	3	25	50	75
	XXX1FM 105	Multi-Disciplinary Course 1 in B – (for batch B1 only)	45	3	3	25	50	75
		Total		**23 – 25**	**21**			**525**
2	XXX2CJ101	Core Course 2 in Major B –	75	5	4	30	70	100
	CSC2CJ101	Fundamentals of Programming (C Language)	75	5	4	30	70	100
	CSC2CJ 102 / CSC1CJ 102 / CSC4CJ 204*	Python Programming	75	5	4	30	70	100
	ENG2FA103(2)	Ability Enhancement Course – 3 (P) (E)	60	4	3	25	50	75
	XXX2FA108(2)	Ability Enhancement Course – 4 (AL)	45	3	3	25	50	75
	CSC2FM 106 / CSC3FM 106	Multi-Disciplinary Course -Digital Empowerment Through Ethical Standards	45	3	3	25	50	75
		Total		**24/ 25**	**21**			**525**
3	XXX3CJ203	Core Course 4 in Major B	60	4	4	30	70	100
	XXX3CJ202	Core Course 5 in Major B	75	5	4	30	70	100
	CSC3CJ203	Software Project Management	60/ 75	4/ 5	4	30	70	100

	CSC3CJ204	Data Structures and Algorithm	60/ 75	4/ 5	4	30	70	100
	XXX3FM 106 / XXX2FM 106	Multi-Disciplinary Course 2 in B –	45	3	3	25	50	75
	XXX3FV 108	Value-Added Course 1 in B – (for batch B1 only)	45	3	3	25	50	75
		Total		**23 – 25**	**22**			**550**
4	CSC4CJ203	Core Course 6 in Major A- Database management system	75	5	4	30	70	100
	XXX4CJXXX	Core Course 6 in Major B	60/ 75	4/ 5	4	30	70	100
	XXX4CJXXX	Core Course 7 in Major B – (for batch B1 only)	60/ 75	4/ 5	4	30	70	100
	CSC4FV 109(2)	Value-Added Course Introduction to Content management system	45	3	3	25	50	75
	XXX4FV 110	Value-Added Course 2 in B –	45	3	3	25	50	75
	CSC4FS 112	Skill Enhancement Course Introduction to Digital Marketing	45	3	3	25	50	75
		Total		**22 – 24**	**21**			**525**
5	CSC5CJ 302	Core Course 7 in Major A Object Oriented Programming	75	5	4	30	70	100
	XXX5CJXXX	Core Course 8 in Major B –	60/ 75	4/ 5	4	30	70	100
	XXX5CJXXX	Core Course 9 in Major B – (for batch B1 only)	60	4	4	30	70	100
		Elective Course 1 in Major A	60	4	4	30	70	100
	XXX5EJXXX	Elective Course 1 in Major B	60	4	4	30	70	100
	XXX5FS 112 / XXX4FS 112	Skill Enhancement Course 1 in B	45	3	3	25	50	75

		Total		**24/ 25**	**23**			**575**
6	CSC6CJ 305/ CSC8MN305	Core Course 8 in Major A Operating System	75	5	4	30	70	100
	XXX6CJXXX	Core Course 10 in Major B –	60/ 75	4/ 5	4	30	70	100
	CSC6CJ 306/ CSC8MN306	Core Course 9 in Major A (for batch A2 only) Introduction to AI and ML	60	4	4	30	70	100
		Elective Course 2 in Major A	60	4	4	30	70	100
	XXX6EJXXX	Elective Course 2 in Major B	60	4	4	30	70	100
	XXX6FS 113	Skill Enhancement Course 2 in B (for batch B1 only)	45	3	3	25	50	75
	XXX6CJ 349	Internship in Major B (Credit for internship to be awarded only at the end of Semester 6)	60		2	50	-	50
		Total		**24/ 25**	**25**			**625**
Total Credits for Three Years					**133**			**3325**
To continue to study Computer Science in semesters 7 and 8, batch B1(A2) needs to earn additional 15 credits in Computer Science to make the total credits of 68. Suppose this condition is achieved, and the student of batch B1(A2) proceeds to the next semesters to study Computer Science. The course structure in semesters 7 and 8 is the same as for pathways 1 – 4, except that the number of the core and elective courses is in continuation of the number of courses in the two categories completed at the end of semester 6, taking into account the number of courses in Computer Science taken online to earn the additional 15 credits.								

* The course code of the same course as used for the pathways 1 – 4

CREDIT DISTRIBUTION FOR BATCH B1(A2) IN PATHWAY 5: DOUBLE MAJOR

Semester	Major Courses in B	General Foundation Courses in B	Internship/ Project in B	Major Courses in Computer Science	General Foundation Courses in Computer Science	AEC	Total
1	4 + 4	3	-	4	-	3 + 3	21
2	4	-	-	4 + 4	3	3 + 3	21

3	4 + 4	3 + 3	-	4 + 4	-	-	22
4	4 + 4	3	-	4	3 + 3	-	21
5	4 + 4 + 4	3	-	4 + 4	-	-	23
6	4 + 4	3	2	4 + 4 + 4	-	-	25
Total for Three Years	**48**	**18**	**2**	**44**	**9**	**12**	**133**
	68			**53**		**12**	**133**
	Major Courses in B	Minor Courses					
7	4 + 4 + 4 + 4 + 4	-			-	-	20
8	4 + 4 + 4	4 + 4 + 4	12		-	-	24
Total for Four Years	**88 + 12 = 100**	**12**					**177**

EVALUATION SCHEME

1. The evaluation scheme for each course contains two parts: internal evaluation (about 30%) and external evaluation (about 70%). Each of the Major and Minor courses is of 4-credits. It is evaluated for 100 marks, out of which 30 marks is from internal evaluation and 70 marks, from external evaluation. Each of the General Foundation course is of 3-credits. It is evaluated for 75 marks, out of which 25 marks is from internal evaluation and 50 marks, from external evaluation.
2. The 4-credit courses (Major and Minor courses) are of two types: (i) courses with only theory and (ii) courses with 3-credit theory and 1-credit practical.
   - In 4-credit courses with only theory component, out of the total 5 modules of the syllabus, one open-ended module with 20% content is designed by the faculty member teaching that course, and it is internally evaluated for 10 marks. The internal evaluation of the remaining 4 theory modules is for 20 marks.
   - In 4-credit courses with 3-credit theory and 1-credit practical components, out of the total 5 modules of the syllabus, 4 modules are for theory and the fifth module is for practical. The practical component is internally evaluated for 20 marks. The internal evaluation of the 4 theory modules is for 10 marks.

3. All the 3-credit courses (General Foundational Courses) in Computer Science are with only theory component. Out of the total 5 modules of the syllabus, one open-ended module with 20% content is designed by the faculty member teaching that course, and it is internally evaluated for 5 marks. The internal evaluation of the remaining 4 theory modules is for 20 marks.

Sl. No.	Nature of the Course		Internal Evaluation in Marks (about 30% of the total)		External Exam on 4 modules (Marks)	Total Marks
			Open-ended module / Practical	On the other 4 modules		
1	4-credit course	only theory (5 modules)	10	20	70	100
2	4-credit course	Theory (4 modules) + Practical	20	10	70	100
3	3-credit course	only theory (5 modules)	5	20	50	75

1. MAJOR AND MINOR COURSES

1.1. INTERNAL EVALUATION OF THEORY COMPONENT

Sl. No.	Components of Internal Evaluation of Theory Part of a Major / Minor Course	Internal Marks for the Theory Part of a Major / Minor Course of 4-credits			
		Theory Only		Theory + Practical	
		4 Theory Modules	Open-ended Module	4 Theory Modules	Practical
1	Test paper/ Mid-semester Exam	10	4	5	-
2	Seminar/ Viva/ Quiz	6	4	3	-
3	Assignment	4	2	2	-
		20	10	10	20*
	Total	30		30	

* Refer the table in section 1.2 for the evaluation of practical component

1.2. EVALUATION OF PRACTICAL COMPONENT

The evaluation of practical component in Major and Minor courses is completely by internal evaluation.

- Continuous evaluation of practical by the teacher-in-charge shall carry a weightage of 50%.
- The end-semester practical examination and viva-voce, and the evaluation of practical records shall be conducted by the teacher in-charge and an internal examiner appointed by the Department Council.
- The process of continuous evaluation of practical courses shall be completed before 10 days from the commencement of the end-semester examination.
- Those who passed in continuous evaluation alone will be permitted to appear for the end-semester examination and viva-voce.

The scheme of continuous evaluation and the end-semester examination and viva-voce of practical component shall be as given below:

Sl. No.	Evaluation of Practical Component of Credit-1 in a Major / Minor Course	Marks for Practical	Weightage
1	Continuous evaluation of practical/ exercise performed in practical classes by the students	10	50%
2	End-semester examination and viva-voce to be conducted by teacher-in-charge along with an additional examiner arranged internally by the Department Council	7	35%
3	Evaluation of the Practical records submitted for the end semester viva–voce examination by the teacher-in-charge and additional examiner	3	15%
	Total Marks	20	

1.3. EXTERNAL EVALUATION OF THEORY COMPONENT

External evaluation carries 70% marks. Examinations will be conducted at the end of each semester. Individual questions are evaluated in marks and the total marks are converted into grades by the University based on 10-point grading system (refer section 5).

PATTERN OF QUESTION PAPER FOR MAJOR AND MINOR COURSES

Duration	Type	Total No. of Questions	No. of Questions to be	Marks for Each	Ceiling of

			Answered	Question	Marks
2 Hours	Short Answer	10	8 – 10	3	24
	Paragraph/ Problem	8	6 – 8	6	36
	Essay	2	1	10	10
				Total Marks	70

2. INTERNSHIP

- All students should undergo Internship of 2-credits during the first six semesters in a firm, industry or organization, or training in labs with faculty and researchers of their own institution or other Higher Educational Institutions (HEIs) or research institutions.
- Internship can be for enhancing the employability of the student or for developing the research aptitude.
- Internship can involve hands-on training on a particular skill/ equipment/ software. It can be a short project on a specific problem or area. Attending seminars or workshops related to an area of learning or skill can be a component of Internship.
- A faculty member/ scientist/ instructor of the respective institution, where the student does the Internship, should be the supervisor of the Internship.

2.1. GUIDELINES FOR INTERNSHIP

1. Internship can be in Computer Science or allied disciplines.
2. There should be minimum 60 hrs. of engagement from the student in the Internship.
3. Summer vacations and other holidays can be used for completing the Internship.
4. In BSc. Computer Science Honours programme, institute/ industry visit or study tour is a requirement for the completion of Internship. Visit to minimum one national research institute, research laboratory and place of scientific importance should be part of the study tour. A brief report of the study tour has to be submitted with photos and analysis.
5. The students should make regular and detailed entries in to a personal log book through the period of Internship. The log book will be a record of the progress of the Internship and the time spent on the work, and it will be useful in writing the final report. It may contain experimental conditions and results, ideas, mathematical expressions, rough work and calculation, computer file names etc. All entries should be dated. The Internship supervisor should periodically examine and countersign the log book.
6. The log book and the typed report must be submitted at the end of the Internship.

7. The institution at which the Internship will be carried out should be prior-approved by the Department Council of the college where the student has enrolled for the UG (Honours) programme.

2.2. EVALUATION OF INTERNSHIP

- The evaluation of Internship shall be done internally through continuous assessment mode by a committee internally constituted by the Department Council of the college where the student has enrolled for the UG (Honours) programme.
- The credits and marks for the Internship will be awarded only at the end of semester 6.
- The scheme of continuous evaluation and the end-semester viva-voce examination based on the submitted report shall be as given below:

<table>
<tr><th>Sl. No.</th><th colspan="2">Components of Evaluation of Internship</th><th>Marks for Internship 2 Credits</th><th>Weightage</th></tr>
<tr><td>1</td><td rowspan="3">Continuous evaluation of internship through interim presentations and reports by the committee internally constituted by the Department Council</td><td>Acquisition of skill set</td><td>10</td><td rowspan="3">40%</td></tr>
<tr><td>2</td><td>Interim Presentation and Viva-voce</td><td>5</td></tr>
<tr><td>3</td><td>Punctuality and Log Book</td><td>5</td></tr>
<tr><td>4</td><td colspan="2">Report of Institute Visit/ Study Tour</td><td>5</td><td>10%</td></tr>
<tr><td>5</td><td rowspan="3">End-semester viva-voce examination to be conducted by the committee internally constituted by the Department Council</td><td>Quality of the work</td><td>6</td><td rowspan="3">35%</td></tr>
<tr><td>6</td><td>Presentation of the work</td><td>5</td></tr>
<tr><td>7</td><td>Viva-voce</td><td>6</td></tr>
<tr><td>8</td><td colspan="2">Evaluation of the day-to-day records, the report of internship supervisor, and final report submitted for the end semester viva–voce examination before the committee internally constituted by the Department Council</td><td>8</td><td>15%</td></tr>
<tr><td></td><td colspan="2">Total Marks</td><td>50</td><td></td></tr>
</table>

3 MINI PROJECT WORK (Skill Enhancement Course 3 - CSC6FS307

A mandatory mini-project (SEC 3) is scheduled in the VI Semester of the BSc (Honours) Computer Science program. It is designed to cultivate students' research and software development skills. It will serve as a capstone experience, allowing students to bridge the gap between theoretical knowledge acquired in the classroom and its practical application to real-world problems.

3.1 Project Selection and Approval:

- Student groups (at most four members) can propose projects in computer science or related disciplines.
- Projects can be experimental (building a prototype), theoretical (a research paper), or computational (implementing an algorithm).
- Project proposals must be submitted for **prior approval** from the Department Council.
- Each project team will be assigned a project supervisor for guidance.

Project Duration:

- The mini-project duration is one semester.
- **Minimum engagement:** 90 hours per student.

Project Deliverables**:**

- Two hard copies and one softcopy of a well-structured typed report outlining:
    - Project objectives and requirements analysis
    - System design and architecture
    - Implementation details (including sample code snippets)
    - Test cases and results
    - Conclusion and future work
- A signed undertaking by the student declaring the originality of the work and the absence of plagiarism.
- A certificate from the project supervisor confirming the same.

3.2 Evaluation Criteria and Rubrics:

1. **Internal Evaluation (25 Marks)** - Conducted by the project supervisor throughout the semester. This could involve:
    - **Project Proposal and Planning**
        - Clarity of project goals and objectives.
        - Feasibility of the chosen approach.
        - Quality of system study/literature review and proposed methodology.
        - Clarity of project schedule and division of tasks within the team.
    - **Project Progress and Implementation**
        - Regular code reviews and adoption of feedback provided by the supervisor.
        - Attendance and active participation in project meetings.
        - Completion of project milestones as planned.
        - Quality of code documentation and adherence to coding standards.
    - **Interim Presentations**
        - Effectiveness of communication and presentation skills.
        - Clarity of technical details and progress made.
        - Ability to answer questions about the project effectively.

Sl. No	Components of Evaluation of Project	Marks for the Internal Evaluation of Mini project
1	Project Proposal and Planning	5
2	Project Progress and Implementation	10
3	Interim Presentations	10
	Total Marks	25

2. **External Evaluation (50 Marks)** - Conducted by an external examiner appointed by the University. This will take place at the end of the VIth semester:
   - **Project Report:**
     - **Content:** Completeness, organisation, clarity, and technical accuracy.
     - **Structure:** Introduction, System Design/literature review, methodology, implementation details, results, discussion, conclusion, future work, and references.
     - **Presentation:** Quality of writing, grammar, and formatting.
   - **Project Demonstration**
     - **Demonstration:** Ability to showcase the functionality of the project or present the research findings effectively.
   - **Viva-voce**
     - **Viva-voce:** Understanding of project concepts, ability to answer questions confidently, and critical thinking skills.

Sl. No	**Components of Evaluation of Project**	**Marks for the End Semester Evaluation of Mini project**
1	**Project Report**	**15**
2	**Project Demonstration**	**20**
3	**Viva-voce**	**15**
Total Marks		**50**

4. PROJECT

4.1. PROJECT IN HONOURS PROGRAMME

- In Honours programme, the student has the option to do a Project of 12-credits instead of three Core Courses in Major in semester 8.
- The Project can be done in the same institution or any other higher educational institution (HEI) or research center.

- A faculty member of the respective institution, where the student does the Project, should be the supervisor of the Project.

4.2. PROJECT IN HONOURS WITH RESEARCH PROGRAMME

- Students who secure 75% marks and above (equivalently, CGPA 7.5 and above) cumulatively in the first six semesters are eligible to get selected to Honours with Research stream in the fourth year.
- In Honours with Research programme, the student has to do a mandatory Research Project of 12-credits in semester 8.
- The approved research centres of University of Calicut or any other university/ HEI can offer the Honours with Research programme. The departments in the affiliated colleges under University of Calicut, which are not the approved research centres of the University, should get prior approval from the University to offer the Honours with Research programme. Such departments should have minimum one faculty member with Ph.D., and they should also have the necessary infrastructure to offer Honours with Research programme.
- A faculty member of the University/ College with a Ph.D. degree can supervise the research project of the students who have enrolled for Honours with Research. One such faculty member can supervise maximum four students in Honours with Research stream.

4.3. GUIDELINES FOR THE PROJECT IN HONOURS PROGRAMME AND HONOURS WITH RESEARCH PROGRAMME

1. Project can be in Computer Science or allied disciplines.
2. Project should be done individually.
3. Project work can be of experimental/ theoretical/ computational in nature.
4. There should be minimum 240 hrs. of engagement from the student in the Project work in Honours programme.
5. There should be minimum 360 hrs. of engagement from the student in the Project work in Honours with Research programme.
6. The various steps in project works are the following:
    - Wide review of a topic.
    - Investigation on a problem in systematic way using appropriate techniques.
    - Systematic recording of the work.
    - Reporting the results with interpretation in a standard documented form.
    - Presenting the results before the examiners.

7. During the Project the students should make regular and detailed entries in to a personal log book through the period of investigation. The log book will be a record of the progress of the Project and the time spent on the work, and it will be useful in writing the final report. It may contain experimental conditions and results, ideas, mathematical expressions, rough work and calculation, computer file names etc. All entries should be dated. The Project supervisor should periodically examine and countersign the log book.
8. The log book and the typed report must be submitted at the end of the Project. A copy of the report should be kept for reference at the department. A soft copy of the report too should be submitted, to be sent to the external examiner in advance.
9. It is desirable, but not mandatory, to publish the results of the Project in a peer reviewed journal.
10. The project report shall have an undertaking from the student and a certificate from the research supervisor for originality of the work, stating that there is no plagiarism, and that the work has not been submitted for the award of any other degree/ diploma in the same institution or any other institution.
11. The project proposal, institution at which the project is being carried out, and the project supervisor should be prior-approved by the Department Council of the college where the student has enrolled for the UG (Honours) programme.

4.4. EVALUATION OF PROJECT

- The evaluation of Project will be conducted at the end of the eighth semester by both internal and external modes.
- The Project in Honours programme/ Honours with Research programme will be evaluated for 300 marks. Out of this, 90 marks is from internal evaluation and 210 marks, from external evaluation.
- The internal evaluation of the Project work shall be done through continuous assessment mode by a committee internally constituted by the Department Council of the college where the student has enrolled for the UG (Honours) programme. 30% of the weightage shall be given through this mode.
- The remaining 70% shall be awarded by the external examiner appointed by the University.
- The scheme of continuous evaluation and the end-semester viva-voce of the Project shall be as given below:

Components of Evaluation of Project	Marks for the Research Project(Honours)/ (Honours with Research) 12 Credits	Weightage
Continuous evaluation of project work through interim presentations and reports by the committee internally constituted by the Department Council	90	30%
End-semester viva-voce examination to be conducted by the external examiner appointed by the university	150	50%
Evaluation of the day-to-day records and project report submitted for the end-semester viva–voce examination conducted by the external examiner	60	20%
Total Marks	300	

INTERNAL EVALUATION OF PROJECT

Sl. No	Components of Evaluation of Project	Marks for the Research Project (Honours programme) /(Honours with Research programme) 12 credits
1	Skill in doing project work	30
2	Interim Presentation and Viva-Voce	20
3	Punctuality and Log book	20
4	Scheme/ Organization of Project Report	20
	Total Marks	90

EXTERNAL EVALUATION OF PROJECT

Sl. No	Components of Evaluation of Project	Marks for the Research Project (Honours programme) / (Honours with Research programme) 12 credits
1	Content and relevance of the Project, Methodology, Quality of analysis, and Innovations of Research	50

2	Presentation of the Project	50
3	Project Report (typed copy), Log Book and References	60
4	Viva-Voce	50
	Total Marks	210

5. GENERAL FOUNDATION COURSES

- All the General Foundation Courses (3-credits) in Computer Science are with only theory component.

5.1. INTERNAL EVALUATION

Sl. No.	Components of Internal Evaluation of a General Foundation Course in Computer Science	Internal Marks of a General Foundation Course of 3-credits in Computer Science	
		4 Theory Modules	Open-ended Module
1	Test paper/ Mid-semester Exam	10	2
2	Seminar/ Viva/ Quiz	6	2
3	Assignment	4	1
		20	5
	Total	25	

5.2. EXTERNAL EVALUATION

External evaluation carries about 70% marks. Examinations will be conducted at the end of each semester. Individual questions are evaluated in marks and the total marks are converted into grades by the University based on 10-point grading system (refer section 5).

PATTERN OF QUESTION PAPER FOR GENERAL FOUNDATION COURSES

Duration	Type	Total No. of Questions	No. of Questions to be Answered	Marks for Each Question	Ceiling of Marks
1.5 Hours	Short Answer	10	8 – 10	2	16
	Paragraph/ Problem	5	4 – 5	6	24
	Essay	2	1	10	10
				Total Marks	50

6. LETTER GRADES AND GRADE POINTS

- Mark system is followed for evaluating each question.

- For each course in the semester letter grade and grade point are introduced in 10-point indirect grading system as per guidelines given below.
- The Semester Grade Point Average (SGPA) is computed from the grades as a measure of the student's performance in a given semester.
- The Cumulative GPA (CGPA) is based on the grades in all courses taken after joining the programme of study.
- Only the weighted grade point based on marks obtained shall be displayed on the grade card issued to the students.

LETTER GRADES AND GRADE POINTS

Sl. No.	Percentage of Marks (Internal & External Put Together)	Description	Letter Grade	Grade Point	Range of Grade Points	Class
1	95% and above	Outstanding	O	10	9.50 – 10	First Class with Distinction
2	Above 85% and below 95%	Excellent	A+	9	8.50 – 9. 49	
3	75% to below 85%	Very Good	A	8	7.50 – 8.49	
4	65% to below 75%	Good	B+	7	6.50 – 7.49	First Class
5	55% to below 65%	Above Average	B	6	5.50 – 6.49	
6	45% to below 55%	Average	C	5	4.50 – 5.49	Second Class
7	35% to below 45% aggregate (internal and external put together) with a minimum of 30% in external valuation	Pass	P	4	3.50 – 4.49	Third Class
8	Below an aggregate of 35% or below 30% in external evaluation	Fail	F	0	0 – 3.49	Fail
9	Not attending the examination	Absent	Ab	0	0	Fail

- When students take audit courses, they will be given Pass (P) or Fail (F) grade without any credits.
- The successful completion of all the courses and capstone components prescribed for the three-year or four-year programme with 'P' grade shall be the minimum requirement for the award of UG Degree or UG Degree Honours or UG Degree Honours with Research, as the case may be.

6.1. COMPUTATION OF SGPA AND CGPA

- The following method shall be used to compute the Semester Grade Point Average (SGPA):

The SGPA equals the product of the number of credits (Ci) with the grade points (Gi) scored by a student in each course in a semester, summed over all the courses taken by a student in the semester, and then divided by the total number of credits of all the courses taken by the student in the semester,

$$\text{i.e. SGPA (Si)} = \Sigma i\,(Ci \times Gi) / \Sigma i\,(Ci)$$

where Ci is the number of credits of the i^{th} course and Gi is the grade point scored by the student in the i^{th} course in the given semester. Credit Point of a course is the value obtained by multiplying the credit (Ci) of the course by the grade point (Gi) of the course.

$$\text{SGPA} = \frac{\text{Sum of the credit points of all the courses in a semester}}{\text{Total credits in that semester}}$$

ILLUSTRATION – COMPUTATION OF SGPA

Semester	Course	Credit	Letter Grade	Grade point	Credit Point (Credit x Grade)
I	Course 1	3	A	8	3 x 8 = 24
I	Course 2	4	B+	7	4 x 7 = 28
I	Course 3	3	B	6	3 x 6 = 18
I	Course 4	3	O	10	3 x 10 = 30
I	Course 5	3	C	5	3 x 5 = 15
I	Course 6	4	B	6	4 x 6 = 24
	Total	20			139
	SGPA				139/20 = 6.950

- The Cumulative Grade Point Average (CGPA) of the student shall be calculated at the end of a programme. The CGPA of a student determines the overall academic level of the student in a programme and is the criterion for ranking the students.

  CGPA for the three-year programme in CUFYUGP shall be calculated by the following formula.

$$\text{CGPA} = \frac{\text{Sum of the credit points of all the courses in six semesters}}{\text{Total credits in six semesters (133)}}$$

CGPA for the four-year programme in CUFYUGP shall be calculated by the following formula.

$$\text{CGPA} = \frac{\text{Sum of the credit points of all the courses in eight semesters}}{\text{Total credits in eight semesters (177)}}$$

- The SGPA and CGPA shall be rounded off to three decimal points and reported in the transcripts.
- Based on the above letter grades, grade points, SGPA and CGPA, the University shall issue the transcript for each semester and a consolidated transcript indicating the performance in all semesters.

Syllabus of Major Courses

Programme	B. Sc. Computer Science				
Course Code	**CSC1CJ101/CSC1MN100**				
Course Title	**Fundamentals of Computers and Computational Thinking**				
Type of Course	Major/Minor				
Semester	I				
Academic Level	100 - 199				
Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	4	3	-	2	75
Pre-requisites	1. Fundamentals of electronic components 2. Basic mathematical operations				
Course Summary	This course provides a comprehensive overview of computing, covering historical milestones, hardware components, software systems, and computational thinking principles. Students will explore the evolution of computing systems, from early pioneers to modern processors and quantum units. The curriculum delves into hardware intricacies, software distinctions, and essential concepts in computer science, emphasizing problem-solving skills and algorithmic thinking. Practical aspects include hands-on experiences with hardware assembling, operating system installation, algorithm and flowchart visualization.				

Course Outcomes (CO):

CO	CO Statement	Cognitive Level*	Knowledge Category#	Evaluation Tools used
CO1	Develop a foundational knowledge of computing systems, encompassing their historical development, evolutionary milestones, and the notable contributions of key figures in the field.	U	F	Instructor-created exams / Quiz
CO2	Acquire familiarity with diverse hardware components constituting a computer system.	U	C	Practical Assignment / Observation of Practical Skills
CO3	Gain practical expertise by engaging in hands-on activities focused on the installation and configuration of diverse hardware components within a computer system.	Ap	P	Practical Assignment / Observation of Practical Skills

CO4	Explore the spectrum of software types, and actively participate in the partitioning, installation, and configuration of operating systems to cultivate a comprehensive understanding of software systems.	Ap	P	Practical Assignment / Observation of Practical Skills
CO5	Develop a foundational understanding of computer science as a discipline, examining problems through the lens of computational thinking and cultivating analytical skills to address challenges in the field.	An	C	Instructor-created exams / Quiz
CO6	Represent complex problems using algorithmic approaches and enhance problem-solving skills by visualizing solutions through the utilization of various software tools.	Ap	P	Practical Assignment / Observation of Practical Skills
* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C) # - Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge (M)				

Detailed Syllabus:

Fundamentals of Computers and Computational Thinking

Module	Unit	Content	Hrs	Marks
I		**History and Evolution of Computing System**	9	15
	1	**Evolution of Computers** – History, Generations	1	
	2	**Overview of Computer System-** Von Neumann Model, Number Systems(Binary, Hexa, Octal, Decimal)	2	
	3	**Number Conversion and Digital Codes-** Conversion from one number system to another, Digital Codes (Gray, Excess-3, BCD)	2	
	4	**Pioneers and Contributors of Computing Systems** - First Mechanical computer - Charles Babbage, Stored-Program Architecture - John von Neumann, Turing machine - Alan Turing, First General-Purpose Electronic Digital Computer -	2	
		John Mauchly and J. Presper Eckert, Artificial Intelligence- John McCarthy (Contributions only).		
	5	**Computing Systems: Past to Present** - Single Core, Dual-Core and Multi-Core Processors, Graphics Processing Unit (GPU), Accelerated Processing Unit, Quantum Processing Units (QPU) (Concept only).	2	

II	**Hardware**		**11**	**20**
	6	**Electronic Components** – Active Components - Diode, Transistor, Integrated Circuits (Definition, Symbol and Function).	1	
	7	**Electronic Components** - Passive Components – Resistors, Capacitors, Inductors (Definition, Symbol and Function).	1	
	8	**Motherboard Components** – CPU and Cooling Fan, RAM, Expansion Slots (PCIe), Input/Output Ports, Chipset (Concept only).	2	
	9	**Motherboard Components** – BIOS/UEFI Chip, SATA/NVMe Slots, Network Interface, Ports- Ethernet, VGA, HDMI, USB (Concept only).	3	
	10	**Computer Components** – SMPS, Motherboard, Storage Devices (HDD, SSD, NVMe)(Concept only).	2	
	11	**Computer Components** – RAM (DRAM, SRAM, DDR SDRAM), ROM, Cache (Concept only).	2	
III	**Software**		**10**	**15**
	12	**Softwares** - Application Software, System Software, Examples	1	
	13	**Operating Systems** – Need of OS, Types – Proprietary and Open Source, Hardware Software Compatibility, POST, Booting.	4	
	14	**OS Installation** – Bootable Media, UEFI / Legacy BIOS, Disk Partitioning, Dual Booting, Boot Manager – BOOTMGR, Grub, File Systems- FAT, NTFS, ext4.	4	
	15	**Device Drivers** – Need of Device Drivers, Driver Interactions (Basic concept only).	1	
IV	**Computer Science and Computational Thinking**		**15**	**20**
	16	**Computer Science**- Introduction, Role of Computer Science in the Modern Era	1	
	17	**Problem Solving** - Defining the Problem, Systematic Approach.	2	
	18	**Computational Thinking** – Problem Decomposition, Pattern Identification, Abstraction, Generalization.	2	
	19	**Logical Thinking** – Inductive and Deductive Reasoning, Logical Expressions.	2	
	20	**Algorithmic Thinking** – Intuition vs Precision, Defining algorithms.	2	
	21	**Algorithm** – Need of Algorithm, Qualities of a Good Algorithm, Examples.	3	
	22	**Flowchart** - Flowchart Symbols, Examples. Raptor.	3	
V	**Lab Activities**		**30**	**30**

		Some of the suggested lab activities are given below. 1. Identify, categorize and list out specifications of given **electronic components**. 2. Identify and list out specifications of given **motherboard components**. 3. Identify and Describe various **ports and connectors on the motherboard**. 4. Installation of various **components on the motherboard** (Processor, Fan, Heat Sink, RAM etc.) 5. Hands-on experience in **assembling and disassembling** a computer system (SMPS, Motherboard, Storage Device etc.). 6. Accessing and configuring the **Basic Input/ Output System (BIOS) or Unified Extensible Firmware Interface (UEFI)** settings. 7. Preparation of **Bootable media** with software like ***Rufus***. 8. Check the hardware compatibility and **Install operating system** (single booting) on given computer. 9. Check the hardware compatibility and **Install operating systems** (dual booting – Windows and Linux) on a given computer. **Develop algorithms and implement** the solutions using ***RAPTOR*** flowchart execution tool for the following problems. 10. Read and print a number. 11. Read the price of three items and print the total bill amount. 12. Read the ages of two persons and print the elder one. 13. Read the number of units of electricity consumed and print the bill amount for various slabs. 14. Read a year and check whether it is a leap year. 15. Print first N numbers (using loop).		

References:

1. Gary B. Shelly, Thomas J. Cashman, and Misty E. Vermaat. “Introduction to Computers”, Cengage Learning, 2008.

2. Pradeep K. Sinha and Priti Sinha, Computer Fundamentals: Concepts, Systems & Applications. BPB Publications.

3. Kevin Wilson, Computer Hardware: The Illustrated Guide to Understanding Computer Hardware. Amazon Digital Services LLC – KDP, 2018.

4. John Hanna, OS Installation 101: A Step-by-Step Approach for Newbies.

5. David Riley and Kenny Hunt, Computational thinking for modern solver, Chapman & Hall/CRC, 2014

6. R.G. Dromey, How to solve it by Computer, PHI, 2008

Programme	B. Sc. Computer Science
Course Code	**CSC2CJ101/CSC2MN100**
Course Title	**Fundamentals Of Programming (C Language)**
Type Of Course	Major/Minor
Semester	II
Academic Level	100 – 199

Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	4	3	-	2	75

Pre-requisites	1. Fundamentals of Algorithms and Flowcharts 2. CSC1CJ101 – Fundamentals of Computers and Computational Thinking
Course Summary	The objectives of this course are to make the student understand programming language, programming, concepts of Loops, reading a set of Data, stepwise refinement, Functions, Control structure, Arrays, Structures, Unions, and Pointers. After completion of this course the student is expected to analyze the real life problem and write a program in 'C' language to solve the problem. The main emphasis of the course will be on problem solving aspect i.e. developing proper algorithms.

Course Outcomes (CO):

CO	CO Statement	Cognitive Level*	Knowledge Category#	Evaluation Tools used
CO1	Remember the program structure of C with its syntax and semantics	U	C	Instructor-created exams / Quiz
CO2	Use the various constructs of a programming language viz. conditional, iteration and recursion.	Ap	P	Practical Assignment / Observation of Practical Skills
CO3	Implement the algorithms in C language.	Ap	P	Practical Assignment / Observation of Practical Skills
CO4	Use simple data structure like array in solving problems.	Ap	C	Practical Assignment / Observation of Practical Skills
CO5	Handling pointers and memory management functions in C.	Ap	P	Practical Assignment / Observation of Practical Skills
CO6	Develop efficient programs for solving a problem.	Ap	P	Viva Voce

* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C)
# - Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge (M)

Detailed Syllabus:

Module	Unit	Content	Hrs
I		**Introduction to C Language**	**10**
	1	History of C, Importance of C, and sample programs	2
	2	Character set, Tokens, Constants, Variables, and Data types	2
	3	Operators - Arithmetic, Relational, logical, assignment, increment, decrement, conditional, bitwise and special operators. Arithmetic expressions, operator precedence, type conversions, mathematical functions	3
	4	Managing Input and Output Operators: Reading and writing a character, formatted input, formatted output.	3
II		**Decision Making Branching and Looping**	**10**
	5	Decision making with If - simple If, If else, nested If else, else If ladder	3
	6	Switch statement, conditional operator, Goto statement	2
	7	Loops: while, do while, for statements and nested loops	3
	8	Jumps in loops – break, continue	2
III		**Arrays and Functions**	**15**
	9	One dimensional array – declaration, initialization and accessing	2
	10	Two dimensional array – declaration, initialization and accessing	2
	11	Multi dimesnsional array, dynamic array	1
	12	Strings – Reading, Writing. Arithmetic operations on characters, Comparisons and string handling functions	2
	13	Functions – Need, Elements of user defined functions and definition	2
	14	Return values and their types, function call and declaration, call by value and call by reference	2
	15	Categories of functions, Nesting of functions	1
	16	Recursion and command line arguments	1
	17	Passing arrays to functions and passing strings to functions	2
IV		**Storage Classes, Structure and Union, Pointers**	**10**
	18	Storage classes – The scope, visibility and lifetime of variables. Auto, Extern, Static and Register storage classes. Storage classes in a single source file and multiple source files	2
	19	Structure and Union - Defining, giving values to members, initialization and comparison of structure variables, arrays of structure, arrays within structures, structures within structures, structures and functions, unions	2
	20	Pointers definition, declaring and initializing pointers, accessing a variable through address and through pointer, pointer expressions, pointer increments and scale factor	2
	21	Pointers and arrays, pointers and functions, pointers and structure	2
	22	Dynamic memory allocation and memory management functions	2
V		**Hands-on Problem Solving Using C** **Practical Applications, Case Study and Course Project**	**30**

	1	Implement the following: **1. Variables, Data types, Constants and Operators:** 1.Evaluation of expression ex: ((x+y) ^2 * (x+z))/w 2.Temperature conversion problem (Fahrenheit to Celsius) 3.Program to convert days to months and days (Ex: 364 days = 12 months and 4 days) 4. Salesman salary (Given: Basic Salary, Bonus for every item sold, commission on the total monthly sales) **2. Decision making (Branch / Loop) Statements:** 5. Solution of quadratic equation 6. Maximum of three numbers 7. Calculate Square root of five numbers (using goto statement) 8. Pay-Bill Calculation for different levels of employee (Switch statement) 9. Fibonacci series 10.Armstrong numbers 11.Pascal ‘s Triangle **3. Arrays, Functions and Strings:** 12. Prime numbers in an array 13. Sorting data (Ascending and Descending) 14. Matrix Addition and Subtraction 15. Matrix Multiplication 16.Transpose of a matrix 17Function with no arguments and no return value 18. Functions with argument and return value 19.Functions with argument and multiple return values 20.Function that convert lower case letters to upper case 21. Factorial using recursion. 22. Perform String Operations using Switch Case	30
		23. Largest among a set of numbers using command line argument **4. Structures and Union:** 24. Structure that describes a Hotel (name, address, grade, avg room rent, number of rooms) Perform some operations (list of hotels of a given grade etc.) 25. Using Pointers in Structures. 26. Cricket team details using Union. **5. Pointers:** 27.Evaluation of Pointer expressions 28.Function to exchange two pointer values 29. Reverse a string using pointers 30.Insertion, deletion, and searching in an array	

Mapping of COs with PSOs and POs:

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PO1	PO2	PO3	PO4	PO5	PO6
CO 1	-	-	1	1	-	-						
CO 2	-	1	2	2	-	-						
CO 3	-	1	3	3	-	-						
CO 4	1	1	2	2	-	-						
CO 5	-	2	2	2	-	-						

CO 6	-	1	3	3	1	1						

Correlation Levels:

Level	Correlation
-	Nil
1	Slightly / Low
2	Moderate / Medium
3	Substantial / High

Mapping of COs to Assessment Rubrics:

	Internal Exam	Assignment	Project Evaluation	End Semester Examinations
CO 1	✓	✓		✓
CO 2	✓	✓		✓
CO 3	✓	✓		✓
CO 4	✓	✓		✓
CO 5	✓	✓		✓
CO 6			✓	

Programme	B. Sc. Computer Science				
Course Code	**CSC3CJ201**				
Course Title	**Software Project Management**				
Type of Course	Major				
Semester	III				
Academic Level	200 - 299				
Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	4	4	-	-	60
Pre-requisites	1. Computer Science knowledge 2. Understanding fundamental computer science concepts, data structures, and algorithms 3. Basic knowledge of project planning and scheduling				
Course Summary	Students are introduced to the concepts, procedures, and resources of software project management in this course. Project scheduling, budgeting, quality assurance, risk management, and teamwork are among the subjects covered. The goal of the course is to equip students with the skills necessary for efficient project management in software development settings.				

Course Outcomes (CO):

CO	CO Statement	Cognitive Level*	Knowledge Category#	Evaluation Tools used
CO1	Define and explain the fundamental concepts, principles, and terminologies related to software project management. Differentiate between various software engineering process models. Understand the agile principle and methodologies and appreciate the need for iterative approaches to software Development	U	C	Instructor-created exams / Quiz

CO2	Master various design concepts used during project development life cycle.	U	P	Assignments/ Test papers/ Viva Voce
CO3	Master various SPM techniques	U	P	Seminar Presentation / Group Tutorial Work/ Viva Voce
CO4	Develop project plans, Create project schedules using tools like Gantt charts and network diagrams	Ap	C	Instructor-created exams / Home Assignments
CO5	Understand the importance of quality in software development by mastering quality assurance processes, methodologies, and testing strategies.	U	P	Writing assignments/ Exams
CO6	Prepare and deliver effective project presentations.	Ap	P	Case Study/ mini Project/ Seminar Presentation/ Group Presentations
* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C) # - Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge (M)				

Detailed Syllabus:

Module	Unit	Content	Hrs (48+12)	Marks (70)
I	**Introduction to Software Engineering and Process Models**		**10**	**12**
	1	Software and Software Engineering- nature of software, Software Engineering, Software Process	2	

	2	Software Development Life Cycle (SDLC)	2	
	3	Prescriptive Process Model- Water fall model, Incremental Model, Evolutionary Process Model	2	
	4	Agile Development- What is Agility, What is agile Process?	2	
	5	Extreme Programming	2	
II	**Software requirements and Design Concepts**		**16**	**22**
	6	Understanding requirements- requirement engineering process	3	
	7	Feasibility studies	1	
	8	Design Concepts- Design process, Design Concepts	2	
	9	Design Model Elements- Data design elements, Architectural design elements, Interface Design Elements, Component-Level Design Elements, Deployment-Level Design Elements	2	
	10	Architectural design using DFD	2	
	11	Component level design guidelines	2	
	12	Modelling with UML – Class diagram Use Case Diagram, State chart Diagram, Activity Diagram,	4	
III	**Software Project Management**		**11**	**18**
	13	Introduction to Software Project Management- Overview of software project management, Importance of project management in software engineering, Role of a project manager	2	
	14	Project Planning and Scope Management- Work breakdown structure (WBS) and project estimation techniques	2	
	15	Project Scheduling and Resource Allocation- Gantt charts and network diagrams,	2	
	16	Critical Path Method (CPM) and Program Evaluation and Review Technique (PERT)	2	
	17	Risk Management-reactive vs proactive risk strategies, Risk identification, risk projection, RMMMM plan	3	
IV		**Software Quality Assurance**	**11**	**18**
	18	Quality Concepts- Software quality, Achieving Software quality,	2	
	19	Testing Strategies	2	
	20	Software testing- levels of software testing	1	

	21	Types of software test- Unit testing, Integration testing, Black box testing, white box testing, System testing	4	
	22	Art of debugging	2	
V	**Open Ended Module- Trends in Software Engineering**		**12**	
	1	• Case study of CASE tools • Prepare a project report • Analysis of real-world software project management case studies • Group project presentations		

References

- Roger S, "Software Engineering – A Practitioner's Approach", seventh edition, Pressman, 2010.
- Pearson Education, "Software Engineering by Ian Sommerville", 9th edition, 2010.
- Pankaj Jalote, An Integrated Approach to Software Engineering, 3rd Edition, Narosa Publishing House.

Mapping of COs with PSOs and POs :

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PO1	PO2	PO3	PO4	PO5	PO6
CO 1	1	1	-	-	3	-						
CO 2	1	1	2	-	3	-						
CO 3	1	1	-	-	3	-						
CO 4	1	1	-	-	3	-						
CO 5	1	1	-	-	3	-						
CO 6	1	1	-	-	3	-						

Correlation Levels:

Level	Correlation
-	Nil
1	Slightly / Low
2	Moderate / Medium
3	Substantial / High

Assessment Rubrics:

- Quiz / Assignment/ Quiz/ Discussion / Seminar
- Midterm Exam
- Programming Assignments (20%)

- Final Exam (70%)

Mapping of COs to Assessment Rubrics :

	Internal Exam	Assignment	End Semester Examinations
CO 1	✓		✓
CO 2	✓		✓
CO 3	✓		✓
CO 4		✓	✓
CO 5		✓	✓
CO 6	✓	✓	

.

Programme	B. Sc. Computer Science				
Course Code	**CSC3CJ202/CSC3MN200**				
Course Title	**Data Structures and Algorithm**				
Type of Course	**Major/Minor**				
Semester	III				
Academic Level	200 - 299				
Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	4	3	-	2	75
Pre-requisites	1. Fundamental Mathematics Concepts: Set, Functions, Logic 2. CSC2CJ101 – Fundamentals of Programming				
Course Summary	This course explores implementations of linked list and array-based data structures, delving into the inner workings of basic data structures including lists, stacks, queues, trees, and graphs.				

Course Outcomes (CO):

CO	CO Statement	Cognitive Level*	Knowledge Category#	Evaluation Tools used
CO1	Differentiate basic data structures (arrays, linked lists, stacks, queues) based on their characteristics, operations, and real-world applications.	U	C	Instructor-created exams / Quiz
CO2	Perform basic operations (e.g., insertion, deletion, search) on fundamental data structures using a chosen programming language.	Ap	P	Practical Assignment / Observation of Practical Skills
CO3	Identify the properties and applications of advanced data structures (trees, graphs).	Ap	P	Seminar Presentation / Group Tutorial Work
CO4	Investigate the properties of various searching and sorting Techniques	U	C	Practical Assignment / Seminar
CO5	Demonstrate critical thinking and problem-solving skills by applying data structures and algorithms to address complex computational challenges.	Ap	P	Viva Voce/ Observation of Practical Skills
CO6	Implement and analyse different data structure algorithms (to solve practical problems.	Ap	P	Case study/ Project
* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C) # - Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge (M)				

Detailed Syllabus:

Module	Unit	Content	Hrs (45+30)	Marks (70)
I	**Introduction to Data Structures and Basic Algorithms**		**9**	**15**
	1	Overview of Data Structures: Data type Vs. Data structure, ADT, Definition of Data structure, Data structure Classification – Linear, Non- Linear (Array, Linked List, Stack, Queue, Tree, Graph)	1	
	2	Introduction to Arrays: Definition, Types (1 Dimensional, 2 Dimensional, Multi-Dimensional, Sparse matrix), Different Array Operations with Algorithm (insertion, deletion, traversal)	3	
	3	Structures and Self-referential structures	1	
	4	Introduction to Linked list: Definition, Types (Single linked list, Doublelinked list, Circular linked list- concept only).	2	
	5	Singly Linked List Operations with Algorithm (insertion, deletion, traversal)	2	
II	**Stack and Queue**		**10**	**20**
	6	Introduction to Stack: Definition, stack operations with Algorithm, Applications: recursion, infix to postfix - example and Algorithm	3	
	7	Implementation of Stack: using array (overflow & underflow) and Linkedlist (with algorithm)	2	
	8	Introduction to Queue: Definition, queue operations with Algorithm, Types: Double ended queue (Input Restricted and Output restricted), Circular queue, Applications	2	
	9	Implementation of Queue: using array and Linked list (with algorithm)	3	
III	**Non- Linear Data Structures**		**16**	**20**
	10	Introduction to Trees: Basic terminology, Types (Binary tree-complete,full, skewed etc., Expression Tree)	2	
	11	Properties of Binary tree, Applications.	2	
	12	Binary tree representations- using array and linked list	2	
	13	Operations on Binary tree- Insertion, Deletion, Traversal- inorder, preorder, postorder - (concepts with examples)	3	
	14	Algorithm of non-recursive Binary tree traversal	3	
	15	Introduction to Graph: Definition, Basic terminology, Types (Directed, Undirected, Weighted).	2	
	16	Graph representation –Adjacency list and Adjacency Matrix, Applications.	2	
IV	**Sorting and Searching**		**10**	**15**
	17	Introduction to Sorting: Definition, Classification (Internal, External)	1	
	18	Internal Sorting Algorithms: Selection sort- Selection sort algorithm, Exchange sort- Bubble sort algorithm	2	
	19	External Sorting Algorithms: Merge sort- Demonstrate with example.(NoAlgorithm needed)	1	
	20	Advanced sorting Algorithm-: Quick sort- Demonstrate with example. (NoAlgorithm needed)	1	

	21	Introduction to Searching: Linear search and Binary search(Algorithm needed) with example.	2	
	22	Hashing: Hash Tables, Hash Functions, Different Hash Functions – Division method, Multiplication method, Mid square method, Folding Method, Collision and Collision resolution Techniques: Open hashing- Chaining, Closed hashing- Probing	2	
V	**Hands-on Programming in Data Structures: Practical Applications, Case Study and Course Project**		**30**	
	1	**Implement the following:** **1.** Basic Operations in a single linked list (Menu driven) **2.** Sort the elements in given singly linked list **3.** Stack using array. **4.** Stack using Linked list **5.** Queue using Array **6.** Queue using Linked list **7.** Sorting algorithms- Selection, Bubble Sort **8.** Searching Algorithms- Linear and Binary search	25	
	2	Project/ Case study	5	

REFERENCES

1. Seymour Lipschutz, "Data Structures with C", McGraw Hill Education (Schaum's Outline Series)
2. Reema Thareja, " Data Structures Using C", Oxford University Press

Mapping of COs with PSOs and POs :

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PO1	PO2	PO3	PO4	PO5	PO6
CO 1	2	1	1	-	-	-						
CO 2	2	1	2	3	-	-						
CO 3	2	1	2	3	-	-						
CO 4	2	-	2	3	-	-						
CO 5	1	1	2	3	1	-						
CO 6	1	1	3	3	1	-						

Correlation Levels:

Level	Correlation
-	Nil

1	Slightly / Low
2	Moderate / Medium
3	Substantial / High

Assessment Rubrics:

- Assignment/ Quiz/ Discussion / Seminar
- Midterm Exam
- Programming Assignments (20%)
- Final Exam (70%)

Mapping of COs to Assessment Rubrics :

	InternalExam	Assignment	Practical Evaluation	End Semester Examinations
CO 1	✓			✓
CO 2	✓	✓		✓
CO 3	✓	✓		✓
CO 4	✓	✓		✓
CO 5		✓		✓
CO 6			✓	

Course Code & Title	**CSC4CJ203**	**Database Management System**			
Type of Course	**Major**	Academic Level	200 - 299		
Pre-requisites	Discrete Mathematics, Data structures and Programming Basics				
Semester	IV				
Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	4	3	-	2	75
Course Summary	This course provides an introduction to database management systems. The topics covered include the concept of Database Management System, ER Model, Relational model, SQL, Database design, Transactions, concepts of other data model-NoSQL and practical session to implement Database Concepts.				

Course Outcomes (CO):

CO	CO Statement	Cognitive Level*	Knowledge Category#	Evaluation Tools used
CO1	A comprehensive understanding of fundamental concepts in database management systems and its application	U	C	Instructor-created exams / Quiz
CO2	Understand concepts of Relational Data Model and Normalization Techniques	U	C	Instructor-created exams / Quiz
CO3	Apply principles of entity-relationship modeling and normalization techniques to design efficient and well-structured databases that meet specified requirements.	Ap	P	Practical Assignment / Observation of Practical Skills
CO4	Acquire expertise in crafting and executing SQL queries for the retrieval, updating, and manipulation	Ap	p	Practical Assignment /
	of data, showcasing adept skills in database querying and data manipulation			Observation of Practical Skills
CO5	Comprehend and apply strategies for managing transactions and implementing mechanisms for controlling concurrency, ensuring the database's consistency and reliability in environments with multiple users.	Ap	P	Practical Assignment / Observation of Practical Skills
CO6	Explore and analyze recent trends in database management systems, with a focus on unstructured databases, NoSQL technologies	An	P	Practical Assignment / Observation of Practical Skills

Detailed Syllabus:

Module	Unit	Content	Hrs	Mark
I	**Database System- Concept**		**10**	**15**
	1	Introduction, Characteristics of the Database Approach	2	

	2	Actors on the Scene, Workers behind the Scene, Advantages of Using the DBMS Approach, File system vs Database	2	
	3	Data Models, Schemas, and Instances , Three-Schema Architecture and Data Independence	3	
	4	Database Languages and Interfaces	2	
	5	Structured, Semi Structured and Unstructured Database	1	
II		**Database Design**	**14**	**20**
	6	ER Model- Basic concepts, entity set & attributes, notations	2	
	7	Relationships and constraints, cardinality, participation, notations, weak entities	2	
	8	Relational Model Concepts-Domains, Attributes, Tuples, and Relations, Values and NULLs in the Tuple	2	
	9	Relational Model Constraints and Relational Database Schemas	2	
	10	Relational Database Design- Atomic Domain and Normalization-INF, 2NF,3NF,BCNF	4	
	11	4NF,5NF	2	
III		**Query Languages**	**11**	**20**
	12	SQL-, introduction to Structured Query Language (SQL)	1	
	13	Data Definition Language (DDL), Table definitions and operations	2	
	14	SQL DML (Data Manipulation Language) - SQL queries on single and multiple tables	4	
	15	Nested queries (correlated and non-correlated), Aggregation and grouping, Views, assertions, Triggers, SQL data types.		
	16	Introduction to NoSQL Databases	2	
	17	Main characteristics of Key-value DB (examples from: Redis), Document DB (examples from: MongoDB)	2	
IV	**Transaction Processing,Concurrency Control**		**10**	**15**
	18	Transaction Processing: Introduction, Transaction and System Concepts	3	
	19	Desirable Properties of Transactions	1	
	20	Characterizing Schedules Based on Recoverability & Serializability	2	
	21	Transaction Support in SQL.	1	
	22	Introduction to Concurrency Control: Two-Phase Locking Techniques	3	
V	**DBMS LAB**		**30**	
	1	Students should decide on a case study and formulate the problem statement.	3	
	2	Based on Identified problem Statement, Design ER Diagram (Identifying entities, attributes, keys and relationships between entities, cardinalities, generalization, specialization etc.) Note: Student is required to submit a document by drawing ER Diagram to the Lab teacher.	3	
	3	Converting ER Model to Relational Model (Represent entities and relationships in Tabular form, Represent attributes as columns, identifying keys) Note: Student is required to submit a document showing the database tables created from ER Model.	2	

	4	Normalization -To remove the redundancies and anomalies in the above relational tables, Normalize up to Third Normal Form	3	
	5	Creation of Tables using SQL- Overview of using SQL tool, Data types in SQL, Creating Tables (along with Primary and Foreign keys), Altering Tables and Dropping Tables	3	
	6	Practicing DML commands-Insert, Select, Update, Delete	2	
	7	Experiment 7:Practicing Queries using ANY, ALL, IN, EXISTS, NOT EXISTS, UNION, INTERSECT, CONSTRAINTS etc.	2	
	8	Practicing Sub queries (Nested, Correlated) and Joins (Inner, Outer and Equi).	2	
	9	Practice Queries using COUNT, SUM, AVG, MAX, MIN, GROUP BY, HAVING, VIEWS Creation and Dropping.	4	
	10	Install and Configure MongoDB to execute NoSQL Commands.	6	

Mapping of COs with PSOs and POs :

	PSO1	PSO2	PSO3	PSO4	PS O5	PSO6	PO1	PO2	PO3	PO4	PO5	PO6
CO 1	3	2	-	-	-	-						
CO 2	3	2	1	-	-	-						
CO 3	1	-	2	3	-	-						
CO 4	-	-	-	3	3	-						
CO 5	-	-	-	3	3	-						
CO 6	-	-	-	-	2	3						

Assessment Rubrics:

- Quiz / Assignment/ Quiz/ Discussion / Seminar
- Midterm Exam
- Programming Assignments (20%)
- Final Exam (70%)

Mapping of COs to Assessment Rubrics :

	Internal Exam	Assignment	Project Evaluation	End Semester Examinations
CO 1	✓			✓
CO 2	✓			✓
CO 3		✓	✓	✓
CO 4		✓	✓	✓
CO 5	✓	✓		✓
CO 6		✓	✓	✓

Text books

1. Database System Concepts (Sixth Edition) Avi Silberschatz, Henry F. Korth, S. Sudarshan McGraw-Hill 2011 ISBN 978-0071325226/ 0-07-352332-1
2. Database Management Systems, Third Edition Raghu Ramakrishnan and Johannes Gehrke McGraw-Hill ©2003 ISBN: 978-0072465631/ 0-07-246563-8

Programme	B. Sc. Computer Science				
Course Code	**CSC3CJ204**				
Course Title	**Python Programming**				
Type of Course	Major				
Semester	IV				
Academic Level	200 - 299				
Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	4	3	-	2	75
Pre-requisites	1. CSC2CJ101 – Fundamentals of Programming				
Course Summary	This course explores the versatility of Python language in programming and teaches the application of various data structures using Python. The course also gives an introduction to scientific computing using popular Python packages.				

Course Outcomes (CO):

CO	CO Statement	Cognitive Level*	Knowledge Category#	Evaluation Tools used
CO1	Understand the basic concepts of Python programming language.	U	C	Instructor-created exams / Quiz
CO2	Apply problem-solving skills using the basic constructs in Python programming	Ap	P	Coding Assignments/ Code reading and review
CO3	Apply modular programming using functions in Python	Ap	P	Coding Assignments/ Code reading and review
CO4	Analyse the various data structures and operations on it using Python	An	C	Instructor-created exams / Case studies
CO5	Apply various packages available in Python	Ap	P	Coding Assignments/ Case studies
CO5	Apply visualization tools in Python	Ap	P	Coding Assignments/ Case studies

* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C)
# - Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge (M)

Detailed Syllabus:

Module	Unit	Content	Hrs
I	**Fundamentals of Python**		**12**
	1	Features of Python, Identifiers, Keywords, Variables, Operators, Operands, Expressions and Data types	3
	2	Precedence and Associativity, Indentation, Comments	1
	3	Input, Output and Import functions, Mathematical functions, range function, Type Conversions	1
	4	Decision-making Structures	3
	5	Looping Structures	3
	6	Control Statements	1
II	**Functions & Modules**		**8**
	7	Function Definition, Function Calling, Flow of Execution, Parameters and Arguments	2
	8	Types of Function Arguments – Required, Keyword, Positional and Variable length arguments	2
	9	Scope and lifetime of variables	1
	10	Types of Functions – Recursive, Anonymous, Functions with more than one return value, Void Functions	2
	11	Built in modules, User defined modules and packages	1
III	**Data Structures in Python**		**15**
	13	Strings - Indexing, Traversal, Slicing, Joining, and Splitting of Strings, Formatting Strings, Operation and Methods of Strings	5
	14	Lists- Indexing and Traversal, Slicing, Joining, and Splitting of Lists, Operations and Methods of Lists	4
	15	Tuples – Indexing and Traversal, Operations and Methods of Tuples	2
	16	Dictionaries – Accessing and Modifying *key-value* pairs in Dictionary, Operations and Methods	3
	17	Sets - Creation and Operations of Sets	1
IV	**Introduction to Scientific Computing in Python**		**10**
	18	Introduction to NumPy Arrays – Advantage of NumPy Arrays, Creation of NumPy Arrays	2
	19	Computation on NumPy Arrays - Universal Functions, Broadcasting, Fancy Indexing	3
	20	Introduction to Pandas - Pandas Series and Pandas Data Frames. Series - Construction from arrays, explicit indices, and dictionaries. **Data Frames-** Construction from arrays and dictionaries.	3
	21	Introduction to Matplotlib Basic plotting - Line plots, Scatter plots, Bar plots ,Histograms and Pie charts.	2
V	**Hands-on Data Structures: Practical Applications, Case Study and Course Project**		**30**

	1	**Basics of Python** 1. Demonstrate basic data types in python using interactive Interpreter. 2. Write a Python script that reads two integers and perform all arithmetic operations on these two numbers. 3. Write a program to compute distance between two points. 4. Write a program to calculate the area of a circle. **Control Structures** 5. Write a program to check whether a number is odd or even. 6. Write a program that reads a positive integer, *n*, from the user and then displays the sum of the first n natural numbers. 7. Write a Python program to check whether a given year is a leap year or not. 8. Develop a program that reads a four-digit integer from the user and displays the sum of the digits in the number. For example, if the user enters 2151 then your program should display 2+1+5+1=9. **Function** 9. Write a program to find the largest of three numbers using functions. The program should pass three numbers as arguments and should return the result. 10. Write a function to check whether a given number is prime or not. 11. Write a recursive function to find the factorial of a number. **Python Data Structures: Strings, Sets, Lists , Tuples and Dictionaries** 12. Create a program that checks whether a given string is a palindrome or not. 13. Write a program to check whether an item exists in a tuple. 14. Write a program to create intersection, union, set difference, and symmetric difference of sets. 15. Write a program to create a telephone directory using a dictionary and display its contents. Also check for a specific phone number in the dictionary. **NumPy, Pandas and Matplotlib** 16. Write a program to implement matrix multiplication using NumPy. 17. Create a pandas series from a dictionary of values, and an ndarray. 18. Write a program to draw a line plot for the given heights and weights of a group of people. height=[145,155,165,175,185,195] weight=[43, 56, 60,69, 78,95]	20
	2	**Case Study**	3
	3	**Capstone (/Course) Project:** Build a practical application using any one package and demonstrate using visualization tools.	7

Mapping of COs with PSOs and POs :

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PO1	PO2	PO3	PO4	PO5	PO6

CO 1	3	-	-	-	-	-						
CO 2	2	-	2	-	1	-						
CO 3	2	-	2	1	-	-						
CO 4	1	-	1	-	-	-						
CO 5	-	2	2	2	2	2						
CO 6	-	2	2	-	2	2						

Mapping of COs to Assessment Rubrics :

	Internal Exam	Assignment	Project Evaluation	End Semester Examinations
CO 1	✓			✓
CO 2	✓			✓
CO 3	✓	✓		✓
CO 4		✓		✓
CO 5		✓		✓
CO 6			✓	

Reference Books:

1. Jose, Jeeva. Taming Python By Programming. Khanna Book Publishing, 2017. Print.
2. S, Gowrishankar, and A, Veena. Introduction to Python Programming. Chapman & Hall/CRC Press, 2018.
3. Downey, Allen. Think Python. Green Tea Press, 2nd ed. 2009
4. VanderPlas, Jake. Python Data Science Handbook: Essential Tools for Working with Data. United States, O'Reilly Media, 2016.
5. Stephenson, Ben. *The Python Workbook*. SPRINGER INTERNATIONAL PU, 2016.

Programme	B. Sc. Computer Science				
Course Code	**CSC4CJ205**				
Course Title	**Computer Networks**				
Type of Course	Major				
Semester	IV				
Academic Level	200 - 299				
Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	4	3	-	2	75
Pre-requisites	1. Knowledge in Computer Organization and Architecture. 2. Knowledge in Operating System.				
Course Summary	This course covers the concepts of data communication and computer networks. It comprises of the study of the standard models for the layered protocol architecture to communicate between autonomous computers in a network and also the main features and issues of communication protocols for different layers. Topics covered comprise of introduction to OSI and TCP/IP models also.				

Sl. NO:	Course Outcome	Cognitive level *	Knowledge category #	Evaluation Tools used
CO1	To understand the fundamentals of computer networks including concepts like data communication ,network topologies and the reference models	U	C	Instructor-Create Exams or Quiz
CO2	Proficiency in Transmission Media and Multiplexing Techniques:	A	P	Discussions and Quizzes
CO3	To familiarise with the common networking protocols and standards	U	F	Instructor created exams or Home assignments
CO4	Describe ,analyse and compare different data link, network and transport layer protocols	A, E	P	Discussions, Quizzes
CO5	Design/implement data link and network layer protocols in simulated networking environment	Ap	P	Viva Voce Observation of practical skills

CO6	To understand the need of various Application layer protocols	U	M	Instructor Created -Exams, Assignments
* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C) # - Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge				
(M)				

Module	Unit	Content	Hrs	Marks
I		**Introduction to Computer networks and Network models**	**12**	**17**
	1	*Types of computer networks, Internet, Intranet, Network topologies, Network classifications.*	2	
	2	*Network Architecture Models: Layered architecture approach, OSI Reference Model, TCP/IP*	2	
	3	*Physical Layer: Analog signal, digital signal, Analog to Digital, Digital to Analog, maximum data rate of a channel transmission*	4	
	4	*Transmission media (guided - unguided transmission media)*	2	
	5	*multiplexing (frequency division multiplexing, time division multiplexing, wavelength division multiplexing)*	2	
II		**Data Link Layer**	**11**	**18**
	6	*Data link layer services, error-detection* Types of errors, Single bit error and Burst error, Vertical redundancy check (VRC), longitudinal redundancy Check (LRC), Cyclic Redundancy Check (CRC),Check sum Error correction - Single bit error correction, Hamming code	2	
	7	*Error correction techniques,* *error recovery protocols (stop and wait, go back n, selective repeat),*	3	
	8	*multiple access protocols, (TDMA/FDP, CDMA/FDD/CSMA/CD, CSMA/CA),*	2	
	9	*Datalink and MAC addressing, Ethernet, Polling*	1	
	10	IEEE Standards- Wireless LANS,Ethernet,Bluetooth	3	
III		**Network layer**	**11**	**18**
	11	*Networking and Internetworking devices - Repeaters, Bridges, Routers, Gateways, Firewall*	2	
	12	*Logical addressing - IPv4 & IPv6 addresses, Network Address Translation (NAT), Internet protocols, internetworking, Datagram,*	2	

	13	*Transition from IPv4 to IPv6*	1	
	14	*Address Mapping-Error reporting and multicasting - Delivery,*	2	
	15	*Forwarding and Routing algorithms, Distance Vector Routing,*	2	
	16	*Link State Routing. Dijkstra*	2	
IV		**Transport Layer and Application layer**	**11**	**17**
	17	*Transport layer, Process-to-process Delivery: UDP, TCP*	2	
	18	*Congestion control and Quality of Service,*	2	
	19	*Domain Name Systems-Remote Login, Email*	2	
	20	*FTP, WWW, HTTP*	2	
	21	*Introductory concepts on Network management & Mail transfer: SNMP,*	2	
	22	*SMTP*	1	
V		**Hands-on Computer Networks:** **Practical Applications,**	**30**	
	1	Lab 1: identifying Networking Hardware components(Jacks, Cables, Tools) Lab 2 IP address - configuring. Lab3. crimping Lab 4: Configuring network host - setting hostname - assigning IP address Lab 5: configuring the Network Interface card – Lab 6: Setup a Wired LAN with more than two systems Lab 7:Setup a Wireless LAN with more than two systems Lab 8: Setting up Internet services File Transfer Protocol(FTP), Lab 9: Simple Mail Transfer Protocol(SMTP) and Post Office Protocol(POP) Lab 10: Setting up Intranet Services - Network File System(NFS),	20	
	2	Case study	3	

	3	**Capstone (/Course) Project:** Build a practical application using Wired Network	7	

References:

1. *Behurouz A Forozan, Introduction to Data Communications & Networking, TMH*
2. *Andrew S. Tanenbaum, Computer Networks, PHI*
3. *William Stallings, Data and Computer Communications, VIIth Edition, Pearson Education*

Programme	B. Sc. Computer Science				
Course Code	CSC5CJ301				
Course Title	Data Mining				
Type of Course	Major				
Semester	V				
Academic Level	300 – 399				
Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	4	4	-	-	60
Pre-requisites	1. Basics of statistics				
Course Summary	This course provides an introduction to the principles, techniques, and applications of data mining.				

Course Outcomes (CO):

CO	CO Statement	Cognitive Level*	Knowledge Category#	Evaluation Tools used
CO1	Understand the fundamental concepts and principles of data mining.	U	C	Instructor-created exams / Quiz
CO2	Demonstrate proficiency in preprocessing techniques such as cleaning, transformation, and reduction of data.	U	P	Assignment / Seminar presentations/ Exams
CO3	Understand popular data mining algorithms and models, such as decision trees, k-means clustering, and association rule algorithms.	U	P	Seminar Presentation / Group Tutorial Work/ Viva Voce
CO4	Explore various methods to Evaluate and interpret the results of data mining models using appropriate performance metrics.	U	C	Instructor-created exams / Home Assignments
CO5	Understand the role of data mining in extracting patterns and knowledge from large datasets.	U	P	Writing assignments/ exams/ Seminar
CO6	Apply data mining techniques to real-world problems and datasets, emphasizing practical applications in various domains	Ap	P	Case Study
* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C) # - Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge (M)				

Detailed Syllabus:

Module	Unit	Content	Hrs (48+12)	Marks (70)
I		**Introduction to Data Mining**	**10**	**15**
	1	Introduction- Data mining defining, KDD vs Data mining, DBMS vs data mining	2	
	2	What kind of data can be mined? - database data, data warehouse, transactional data, other types	2	
	3	What kind of patterns can be mined? - Class/Concept Description: Characterization and Discrimination, Mining Frequent Patterns, Associations, and Correlations, Classification and Regression for Predictive Analysis, cluster analysis, outlier analysis	3	
	4	Technologies used- statistics, machine learning, data base systems and ware house, information retrieval (Introduction only)	3	
II		**Data Preprocessing**	**14**	**20**
	5	Data Preprocessing: An Overview	2	
	6	Data Cleaning- missing value, noisy data, Data Cleaning as a Process	2	
	7	Data Integration- Entity Identification Problem, Redundancy and Correlation Analysis, Tuple Duplication, Data Value Conflict Detection and Resolution	3	
	8	Data Reduction - Wavelet Transforms, Principal Components Analysis, Attribute Subset Selection, Regression and Log-Linear Models: Parametric Data Reduction, Histograms,	4	
	9	Data Transformation and and Data Discretization- Data Transformation by Normalization, Discretization by Binning	3	
III		**Association Rule Mining & Classification**	**10**	**15**
	10	Introduction to Association Rule Mining Frequent Itemset, Closed Itemset, and Association Rules	1	
	11	Frequent Itemset Mining Apriori Algorithm, Generating Association Rules from Frequent Itemsets	1	
	12	Introductio to classification: Decision tree	2	
	13	Attribute Selection measures in decision tree	2	
	14	Bayes Classification methods	2	
	15	Techniques to Improve Classification Accuracy	2	
IV		**Clustering, Outlier detection**	**14**	**20**
	16	Introduction to unsupervised techniques: challenges	2	
	17	Clustering- K Means	2	
	18	Variants of k- Means	2	

	19	Hierarchical clustering	2	
	20	Density Based clustering- DBScan	2	
	21	Introduction to outliers and novelty detection	2	
	22	Recommender system	2	
V		**Open Ended Module: Case Studies**	**12**	
	1	• **Real-world applications of data mining** • **Case studies and projects** • **Ethical considerations in data mining**		

References

- "Han, J., Kamber, M., & Pei, J. (2011). Data mining: Concepts and techniques. Morgan Kaufmann."
- Data Mining Techniques - Arun K. Pujari
- Jiawei Han and Micheline Kamber, Data Mining Concepts & Techniques, Second Edition, Elsevier.
- Pang Ning Tan, Michael Steinbach and Vipin Kumar, Introduction To Data Mining, Pearson Education, 2007.

Mapping of COs with PSOs and POs :

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PO1	PO2	PO3	PO4	PO5	PO6
CO 1	1	-	-	-	-	1						
CO 2	1	-	-	-	1	1						
CO 3	1	-	2	-	2	2						
CO 4	1	-	1	-	1	1						
CO 5	1	-	1	-	1	1						
CO 6	-	-	1	1	2	2						

Correlation Levels:

Level	Correlation
-	Nil
1	Slightly / Low
2	Moderate / Medium
3	Substantial / High

Assessment Rubrics:

- Quiz / Assignment/ Quiz/ Discussion / Seminar

- Midterm Exam
- Programming Assignments (20%)
- Final Exam (70%)

Mapping of COs to Assessment Rubrics :

	Internal Exam	Assignment	End Semester Examinations
CO 1	✓		✓
CO 2	✓		✓
CO 3	✓		✓
CO 4	✓	✓	✓
CO 5	✓	✓	✓
CO 6	✓	✓	

Programme	B. Sc. Computer Science				
Course Code	**CSC5CJ302**				
Course Title	**Object Oriented Programming (Java)**				
Type of Course	**Major**				
Semester	V				
Academic Level	300-399				
Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	4	3	-	2	75
Pre-requisites	1. Knowledge in basic programming 2. Knowledge in OOP Concepts				
Course Summary	The aim of this course is to provide students with an understanding of the basic concepts in Java programming. This course will help students create GUI applications in Java and establish database connectivity.				

Course Outcomes (CO):

CO	CO Statement	Cognitive Level*	Knowledge Category#	Evaluation Tools used
CO1	To understand the concepts and features of Object Oriented Programming(OOPs)	U	C	
CO2	To practice programming in Java	Ap	P	
CO3	To learn java's exception handling mechanism, I/O operations and multithreading.	Ap	P	
CO4	To learn java's O operations and multithreading.	Ap	P	
CO5	Implement programs using Java Database Connectivity	Ap	P	
CO6	Students will be capable of developing Graphical User Interface (GUI) applications using Swing, understanding layout management, and implementing basic event handling.	Ap	P	
* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C) # - Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge (M)				

Detailed Syllabus:

Module	Unit	Content	Hrs	Marks (70)
I		**Review of OOPs and Introduction to Java**	**17**	**20**
	1	Overview of OOPs Concept	1	
	2	History of Java and Java Virtual Machine	1	
	3	Basic Structure of Java Programming : Data Types, Operators, Expression and Control Statement	2	
	5	Arrays and String: One Dimensional Array, Multidimensional Array, String Operations	2	
	6	Scanner, Type Conversion and Casting	2	
	7	Introduction to Class and Objects: Definition of Class and Objects, Access Modifier	2	
	8	Constructor and Inheritance: Types of Constructors, Types of Inheritance, use of extends, super, final, this keyword	3	
	9	Method Overriding, Method Overloading and Dynamic Method Dispatch : Programming implementation of Method Overriding and Overloading	2	
	10	Interface, Abstract Class and Packages ; Programming implementation of Interface, Abstract class and Packages	2	
II		**Exception and I/O Operations**	**8**	**15**
	11	Exception: Baic Concept of exception and Exception Hierarchy	2	
	12	Managing Exception: Use of try….catch ….. finally blocks, throw and throws keyword	2	
	13	Managing Input/Output files in Java : Importance of I/O Operations, BufferedInputStream, BufferedOutputStream	2	
	14	File Operations : Programming implementation of FileInputStream, FileOutputStream, FileReader, FileWriter	2	
III		**Multithreading and Database Connectivity**	**9**	**20**
	15	Thread : Concept of Thread and Thread state	2	
	16	Programming Implementation of Thread : Using extending thread class and Runnable interface, Thread Priorities	2	

	17	Database Programming : Basic Concept of Database and JDBC Driver, Connecting with Database	2	
	18	Querying Database: Programming implementation of creating table, insert and update values to the table using	3	
		preparedStatement, Statement object and querying the values using ResultSet and ResultSetMetadata		
IV		**GUI Programming**	**11**	**15**
	19	Introduction to GUI Application : AWT Basics, Introduction to IDE	2	
	20	Swing Programming : Introduction of Model-View-Controller Pattern	2	
	21	Introduction to layout Management : Fundamental controls used in SWING	4	
	22	Event Handling : Basic Knowledge of Event Handling(Event Class and Event Listener)	3	
V		**Hands-on Programming in Java(Using IDE NetBeans, Eclipse, VSCode):**	**30**	**30**
		Practical Applications, Case Study and Course Project		
	1	Implement the following:		
		1. String and Arrays:	20	
		Write a program to perform various String operations in Java(Hint: charAt, substring, concat, equals,, isEmpty..)		
		Write a program to implement Multi-Dimensional Array(Hint : Matrix multiplication)		
		2. Object Oriented Programming Concept:		
		Write a program to implement the concept of class and object.(Hint: Complex Number addition)		
		Write a program to demonstrate the order in which constructors are invoked in multilevel inheritance.		
		Write a program to implement method overloading		
		Write a program to implement method overriding.		
		3. Exception Handling and Multithreading:		
		Write a program to implement try…cath, finally block (Hint: Arithmetic and ArrayOutOfBoundException)		

		Write a multi thread java program for displaying odd numbers and even numbers up to a limit (Hint :Create thread by inheriting Thread class).		
		Write a multi thread java program for displaying odd numbers and even numbers up to a limit (Hint : Implement thread using Runnable interface).		
		4. GUI Application with Database:		
		Write a swing program to track mouse & key events		
		Write a swing program to fetch data from TextFiled and display it in Label		
		Write a swing program to perform form validation		
		Write a swing program to display data in tabular form		
		Write a simple login program without database connectivity		
		Write a swing program to create a registration form (Hint : Create table student in any database and link the registration form with database using JDBC)		
	2	Case Study	2	
	3	**Project:** Build a application for shop management system (Eg: Admin Login, Product registration, stock management, product selling, employee salary)	8	

Text Book :

1. Herbert Scheldt, Java: The Complete Reference, 12th Edition, Tata McGraw-Hill Edition, ISBN: 9781260463415.

References :

1. C. Thomas Wu, An introduction to Object-oriented programming with Java, 5e, McGraw-Hill, 2009.
2. Y. Daniel Liang, Introduction to Java programming, Comprehensive Version, 10e, Prentice Hall India, 2013.
3. K. Arnold, J. Gosling, David Holmes, The JAVA programming language, 4e, Addision- Wesley, 2005.

Mapping of COs with PSOs and POs :

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PO1	PO2	PO3	PO4	PO5	PO6

CO 1	1	-	3	3	-	-						
CO 2	1	-	3	3	-	-						
CO 3	-	-	3	3	2	3						
CO 4	-	-	2	3	-	-						
CO 5	-	-	3	3	2	3						
CO 6			3	3	3							

Correlation Levels:

Level	**Correlation**
-	Nil
1	Slightly / Low
2	Moderate / Medium
3	Substantial / High

Assessment Rubrics:

- Quiz / Assignment/ Quiz/ Discussion / Seminar
- Midterm Exam
- Programming Assignments (20%)
- Final Exam (70%)

Mapping of COs to Assessment Rubrics :

	Internal Exam	Assignment	Project Evaluation	End Semester Examinations
CO 1	✓			✓
CO 2	✓			✓
CO 3	✓	✓		✓
CO 4		✓		✓
CO 5		✓		✓

Programme	B. Sc. Computer Science
Course Code	**CSC5CJ303**
Course Title	**Full Stack Web Development**
Type of Course	**Major**
Semester	V
Academic Level	300-399

Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	4	3	-	2	75

Pre-requisites	1. Fundamental of Web Pages and web servers 2. Basics of HTML
Course Summary	This course provides the ideas, techniques, and applications for efficient Web Development. The advanced industry demand and emerging trends are covered in this syllabus.

Course Outcomes (CO):

CO	CO Statement	Cognitive Level*	Knowledge Category#	Evaluation Tools used
CO1	Understand the concepts to create responsive web pages using HTML and CSS	U	C	Instructor-created exams / Quiz
CO2	Familiarization with Client-side Scripting using JavaScript	U	C	Practical Assignment / Observation of Practical Skills
CO3	Understand Node.JS and equip learners with a comprehensive understanding of NodeJS and its functionalities.	U	F	Seminar Presentation / Group Tutorial Work/ Viva Voce

CO4	Understanding and building interactive web pages using React JS.	U	P	Instructor-created exams / Home Assignments
CO5	Familiarization with SQL and NoSQL	Ap	P	Writing assignments/ Instructor-created exams/ practicals
CO6	Explore MongoDB and Develop real-world web applications using various technologies learned in the course	Ap	P	Case Study/ mini Project/ practicals
* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C) # - Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge (M)				

Detailed Syllabus:

Module	**Unit**	**Content**	**Hrs (45+30)**	**Marks (70)**
I	**HTML & CSS**		**9**	**12**
	1	Introduction to HTML5 Tags, Attribute and Elements Doctype Element, Comments	2	2
	2	Semantic tags Headings, Paragraphs, and Formatting Text Lists, Links, Images-	1	2
	3	Forms and Tables Introduction CSS Applying CSS to HTML.	2	2
	4	Selectors, Properties and Values CSS Colors and Backgrounds CSS Box Model	3	5

	5	CSS Margins, Padding, and Borders CSS Text and Font Properties Webpage Layout Responsive web design	1	1
II		**JavaScript & Node.JS**	**11**	**15**
	6	Introduction to JavaScript Applying JavaScript (internal and external) Understanding JS Syntax	1	2
	7	Introduction to Document and Window Object Variables and Operators Data Types and Num Type Conversion	1	2
	8	Math and String Manipulation Objects and Arrays Date and Time Conditional Statements	2	3
	9	Switch Case Looping in JS Functions	2	2
	10	Node.JS Overview Node.JS - Basics and Setup Node.JS Console Node.JS Command Utilities Node.JS Modules	3	3
	11	Node.JS Concepts Node.JS Events Node.JS with Express js Node.JS Database Access	2	3
III		**React.JS**	**12**	**15**
	12	Introduction Templating using JSX	2	3
	13	Components, State and Props Lifecycle of Components Rendering List and Portals	3	3
	14	Redux and Redux Saga Immutable.js Service Side Rendering	2	3
	15	Unit Testing	2	3
	16	Webpack	3	3
I V		**MongoDB**	**13**	**20**
	17	SQL and NoSQL Concepts	3	4
	18	Create and Manage MongoDB	2	3
	19	Migration of Data into MongoDB	1	3
	20	MongoDB with PHP	1	3
	21	MongoDB with NodeJS.	2	4
	22	Services Offered by MongoDB	3	3
V		**Practical Implementations of Full Stack Web Development**	**30**	**20**
	1	● Webpage Development using HTML And CSS	25	

		• Webpage Development using Javascript & Node.JS • Webpage Development using React.JS With Backend MongoDB		
	2	Case Study/ Project	**5**	

References Books

1. Hawramani, Ikram. HTML, CSS and JavaScript for Complete Beginners: A Step by Step Guide to Learning HTML5, CSS3 and the JavaScript Programming Language. United States, Amazon Digital Services LLC - KDP Print US, 2018.

2. Soni, Ravi Kant. Full Stack AngularJS for Java Developers: Build a Full-Featured Web Application from Scratch Using AngularJS with Spring RESTful. United States, Apress, 2017.

3. Northwood, Chris. The Full Stack Developer: Your Essential Guide to the Everyday Skills Expected of a Modern Full Stack Web Developer. Germany, Apress, 2018.

4. Sharma, Aneeta. Full-Stack Web Development with Vue. Js and Node: Build Scalable and Powerful Web Apps with Modern Web Stack: MongoDB, Vue, Node. Js, and Express. United Kingdom, Packt Publishing, Limited, 2018.

5. Sharma, Manu. Mongodb Complete Guide: Develop a Strong Understanding of Administering Mongodb, Crud Operations, and Mongodb Commands. India, Bpb Publications, 2021.

Mapping of COs with PSOs and POs :

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6
CO 1	-	1	3	2	2	3						
CO 2	-	1	3	3	3	2						
CO 3	-	1	3	3	3	2						
CO 4	-	1	3	3	3	2						
CO 5	-	1	3	3	3	2						
CO 6	-	1	3	3	3	2						

Correlation Levels:

Level	Correlation
-	Nil
1	Slightly / Low
2	Moderate / Medium
3	Substantial / High

Assessment Rubrics:

- Quiz / Assignment/ Quiz/ Discussion / Seminar
- Midterm Exam
- Programming Assignments (20%)
- Final Exam (70%)

Mapping of COs to Assessment Rubrics :

Mapping of COs to Assessment Rubrics :

	Internal Exam	Assignment	End Semester Examinations	Practical
CO 1	✓		✓	
CO 2	✓	✓	✓	✓
CO 3	✓	✓	✓	✓
CO 4	✓	✓	✓	
CO 5	✓	✓	✓	✓
CO 6	✓		✓	✓

Programme	B. Sc. Computer Science				
Course Code	**CSC6CJ304/** CSC8MN304				
Course Title	**Digital Electronics and Computer Architecture**				
Type of Course	**Major/Minor**				
Semester	VI				
Academic Level	300-399				
Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	4	4	-	-	60
Pre-requisites	Basic understanding of mathematical concepts, especially areas like algebra				
Course Summary	This course provides a comprehensive introduction to the fundamentals of digital systems, covering topics related to binary arithmetic, basic computer logic, combinational and sequential logic circuits, as well as basic computer organization and design. Throughout the course, students will gain a solid understanding of digital systems, from the basic building blocks of logic circuits to the design and organization of processors and memory				

Course Outcomes (CO):

CO	CO Statement	Cognitive Level*	Knowledge Category#	Evaluation Tools used
CO1	Understand Basic Binary arithmetic Techniques	U	C	Instructor-created exams / Quiz
CO2	Implement logic operations using basic gates and Boolean algebra, design and optimise logic expressions using Karnaugh maps and design combinational logic circuits	Ap	P	Instructor-created exams/ Home Assignments
CO3	Understand the operation of latches and flip flops and the design of sequential logic circuits	U	C	Instructor-created exams

CO4	Learn the basic computer organization by understanding the role of registers, buses, ALU and control unit and the concepts like parallel processing and pipelining	U	C	Instructor-created exams
CO5	Understand how instructions represented, addressed and executed and how a microprogrammed control unit work	U	C	Instructor-created exams
CO6	Understand the concepts of memory and IO organization	U	C	Instructor-created exams
* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C) # - Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge (M)				

Detailed Syllabus:

Module	**Unit**	**Content**	**Hrs (48+12)**	**Marks (70)**
I		**Number systems and Boolean Algebra**	**10**	**15**
	1	Binary arithmetic: Addition, Subtraction, Concepts of 1's and 2's complement, 1's and 2's complement addition	2	
	2	Logic Gates: AND, OR, NOT, NOR, NAND, XOR, XNOR , Universal Property of NAND and NOR gates	3	
	3	Boolean algebra: Boolean operations, laws and rules, Demorgan's theorem	2	
	4	Boolean Expression Simplification using K Map up to 4 variable	3	
II		**Combinational and Sequential Logic Circuits**	**12**	**20**
	5	Combinational Circuits: Half Adder, Full Adder, Ripple Carry Adder	1	
	6	Combinational Circuits: Encoder and Decoder (Basic Circuit Only)	1	
	7	Combinational Circuits: Multiplexer and Demultiplexer (Basic Circuit Only)	1	
	8	Concepts of Latches and Flip Flops, Types of Flip Flops (SR, D, JK, T): Truth Table and Circuit	3	
	9	Sequential Circuits: Synchronous and Asynchronous Counters	4	

	10	Johnsons and Ring counter, Shift Registers	2	
III	**Basic Computer Organization and Micro Programmed Control**		**10**	**15**
	11	Instruction codes, Registers and Common Bus system	2	
	12	Computer Instructions	1	
	13	Timing and Control: Concepts of hardwired and microprogrammed control	1	
	14	Instruction Cycle	1	
	15	Microprogrammed Control: Control memory & Address Sequencing	3	
	16	Micro Instruction Format and Symbolic Micro Instruction	2	
IV	**Processor ,Memory and I/O Organization**		**16**	**20**
	17	Processor Organisation: General Register organization and stack organization, Instruction formats and addressing modes	4	
	18	Processor Organisation: RISC vs CISC , Parallel Processing	2	
	19	Pipelining: General Considerations , Arithmetic Pipeline, Instruction Pipeline	3	
	20	Memory Organisation: Memory Hierarchy, Main Memory	1	
	21	Associative Memory, Cache Memory Mapping	4	
	22	IO Organisation: Modes of transfer: programmed IO, Interrupt initiated IO, DMA (Concepts Only)	2	
V	**Open Ended Module: Computer Arithmetic & Types of Instruction**		**12**	
	1	Computer Arithmetic: Addition and Subtraction, Multiplication Algorithms, Division Algorithms	7	
	2	Examples for Memory Reference, Register Reference, Input Output Instructions, Data Transfer Instructions, Data Manipulation Instructions , Arithmetic Instructions , Logical and Bit Manipulation Instructions , Shift Instructions, Program Control Instruction, Conditional Branch Instructions Subroutine Call and Return	5	

References

1. “Digital Fundamentals”, Thomas L. Floyd
2. “Computer System Architecture”, M. Morris Mano
3. “Computer Organization” , Carl Hamacher, Zvonko Vranesic

Mapping of COs with PSOs and POs :

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PO1	PO2	PO3	PO4	PO5	PO6
CO 1	3	-	-	-	-	-						
CO 2	-	3	-	-	-	-						
CO 3	2	3	-	-	-	-						
CO 4	2	2	-	-	-	-						
CO 5	-	3	-	-	-	-						
CO 6	-	3	-	-	-	-						

Correlation Levels:

Level	Correlation
-	Nil
1	Slightly / Low
2	Moderate / Medium
3	Substantial / High

Assessment Rubrics:

- Quiz / Assignment/ Quiz/ Discussion / Seminar
- Midterm Exam
- Programming Assignments (20%)
- Final Exam (70%)

Mapping of COs to Assessment Rubrics :

	Internal Exam	Assignment	Project Evaluation	End Semester Examinations
CO 1	✓			✓
CO 2	✓			✓
CO 3	✓			✓
CO 4	✓	✓		✓
CO 5	✓	✓		✓
CO 6	✓			✓

Programme	B. Sc. Computer Science				
Course Code	**CSC6CJ305/** CSC8MN305				
Course Title	**Principles of Operating System**				
Type of Course	Major/Minor				
Semester	VI				
Academic Level	300-399				
Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	4	3	-	2	75
Pre-requisites	Knowledge in Basic System Architecture				
Course Summary	This course provides students with a comprehensive understanding of the fundamental principles, design concepts, and practical implementation aspects of operating systems. The course covers key topics such as Process Management, CPU Scheduling, Memory Management and Linux Shell Programming concepts.				

Course Outcomes (CO):

CO	CO Statement	Cognitive Level*	Knowledge Category#	Evaluation Tools used
CO1	Summarize the History, Objectives and Functions of an operating system	U	C	Instructor-created exams / Quiz
CO2	Understand process management concepts: Process Control Block, States, Scheduling, Operations, Inter process Communication	U	C	Instructor-created exams
CO3	Evaluate various processor scheduling strategies, algorithms	E	P	Seminar Presentation / Group Tutorial Work
CO4	Apply process synchronisation concepts for effective process management	Ap	P	Viva Voce
CO5	Analyse conditions for deadlock occurrence and methods of resolving.	An	C	Instructor-created exams/Assignments

CO6	Describe various memory management techniques, including paging , segmentation and virtual memory	U	C	Instructor-created exams / Home Assignments
CO7	Develop Shell Scripts using Linux	C	P	Practical Assignment / Observation of Practical Skills
* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C) # - Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge (M)				

Detailed Syllabus:

Module	**Unit**	**Content**	**Hrs (45+30)**	**Marks (70)**
I		**Introduction to Operating Systems & Process Management**	**10**	**15**
	1	Operating System: History, Types, Objectives and Functions	2	
	2	Process Concepts: Process States, Process Control Block	2	
	3	Types of Process Schedulers and Operations on Process	2	
	4	Co operating Processes	2	
	5	Inter Process Communication	2	
II		**CPU Scheduling, Process Synchronisation and Deadlocks**	**15**	**20**
	6	Basic Scheduling Concepts, Scheduling Criteria	1	
	7	CPU Scheduling Algorithms	2	
	8	Process Synchronisation: Critical Section	2	
	9	Semaphores	2	
	10	Classical Problems of Synchronisation: Reader Writer, Dining Philosopher	2	
	11	Introduction to Deadlock: Necessary Conditions, Resource Allocation Graph	2	
	12	Handling Deadlocks: Prevention, Avoidance, Detection & Recovery	4	
III		**Memory Management Techniques**	**10**	**20**
	13	Basic Concepts: Physical VS Logical Address, Continuous Memory Allocation	2	
	14	Fragmentation Problem and Solutions	1	
	15	Non contiguous Memory Allocation: Paging	2	
	16	Non contiguous Memory Allocation: Segmentation, Segmentation with Paging	2	

<table>
<tr><td></td><td>17</td><td>Virtual Memory Concepts: Demand Paging and Page Replacement Algorithms, Thrashing</td><td>3</td><td></td></tr>
<tr><td>IV</td><td colspan="2">Linux Shell Programming</td><td>10</td><td>15</td></tr>
<tr><td></td><td>18</td><td>Introduction: Types of Linux Shells, File
Directory & File Management Commands:ls, cd,pwd,mkdir,rm,cp,mv, chmod,touch
Input/Output Commands: read, echo,
Text Processing Commands: grep , cat</td><td>2</td><td></td></tr>
<tr><td></td><td>19</td><td>Piping and Redirection operators: |,>,<,>>,<<
Arithmetic, Logical and Relational Operator</td><td>2</td><td></td></tr>
<tr><td></td><td>20</td><td>Iterative and Conditional Commands : if, while, for, break, continue, case</td><td>2</td><td></td></tr>
<tr><td></td><td>21</td><td>Arrays and functions</td><td>2</td><td></td></tr>
<tr><td></td><td>22</td><td>Command line arguments, Network commands: ipconfig, ping, date and time commands, Informative commands: random, w, ps, free, uptime</td><td>2</td><td></td></tr>
<tr><td>V</td><td colspan="2">Practical Applications using Linux Shell Programming</td><td>30</td><td></td></tr>
<tr><td></td><td></td><td>Implement the following:
1. Write a Shell Script to find the roots of a quadratic equation.
2. Write a shell script for a menu driven program to perform file management (File creation, display content, remove, write content to a file).
3. Write a shell script to count no of line, words and characters of an input file.
4. Write a shell script to find the average of the number entered as command line arguments.
5. Write a shell script to copy the contents of file to another. Input file names through command line. The copy should not be allowed if second file exists.
6. Write a shell script to check network connectivity.
7. Write a shell script that analyzes a log file, extracting and summarizing relevant information such as error counts ,warning messages, info and debug messages using grep command.
8. Write a shell script to display current date and time, list all user account names, count of logged in user accounts, list all logged in user accounts with login time.
9. Write a simple game script using random function to implement number guessing game.
10. Write a shell script to display your system details (number of users, current processes, memory usage , system running time).</td><td>30</td><td></td></tr>
<tr><td></td><td></td><td>11. Write a shell script to implement and examine the effectiveness of the First Come First Serve CPU Scheduling algorithm. Find the average waiting time and turnaround time.
12. Write a shell script program to implement Inter Process Communication.</td><td></td><td></td></tr>
</table>

References

1. Silberschatz, Galvin and Gagne, Operating System Concepts, John Willey & Sons
2. William Stallings, Operating Systems, Internals and Design Principles, PHI

Mapping of COs with PSOs and POs :

	PSO1	PSO2	PSO3	PSO4	PS O5	PSO6	PO1	PO2	PO3	PO4	PO5	PO6
CO 1	2	-	-	-	-	-						
CO 2	-	2	-	-	-	-						
CO 3	-	3	-	1	-	-						
CO 4	-	2	2	-	-	-						
CO 5	-	3	-	-	-	-						
CO 6	-	3	-	-	-	-						
CO7	-	-	2	2	-	-						

Correlation Levels:

Level	Correlation
-	Nil
1	Slightly / Low
2	Moderate / Medium
3	Substantial / High

Assessment Rubrics:

- Quiz / Assignment/ Quiz/ Discussion / Seminar
- Midterm Exam
- Programming Assignments (20%)
- Final Exam (70%)

Mapping of COs to Assessment Rubrics :

	Internal Exam	Assignment	Practical Evaluation	End Semester Examinations
CO 1	✓			✓
CO 2	✓			✓
CO 3	✓	✓		✓
CO 4		✓		✓
CO 5	✓			✓
CO 6	✓			✓

CO7			✓	

Programme	B. Sc. Computer Science				
Course Code	**CSC6CJ306/** CSC8MN306				
Course Title	**Introduction to Artificial Intelligence & Machine Learning**				
Type of Course	Major/Minor				
Semester	VI				
Academic Level	300 - 399				
Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	4	3	-	2	75
Pre-requisites	1. Fundamental Mathematics Concepts: Sets 2. Fundamentals of Python Programming				
Course Summary	This course provides an introduction to the ideas, techniques, and applications of artificial intelligence (AI) is given in this course. The fundamentals of knowledge representation, machine learning, and problem solving will be taught to the students.				

Course Outcomes (CO):

CO	CO Statement	Cognitive Level*	Knowledge Category#	Evaluation Tools used
CO1	Differentiate various knowledge representation methods, AI operations, Machine learning approaches and real-world applications.	U	C	Instructor-created exams / Quiz
CO2	Master Problem-Solving Techniques (search algorithms, heuristic approaches, and informed search strategies). Analyse and evaluate its efficiency.	Ap	P	Practical Assignment / Observation of Practical Skills
CO3	Investigate the properties and applications of various machine learning techniques	Ap	C	Seminar Presentation /
				Group Tutorial Work/ Viva Voce

CO4	Evaluate Artificial Intelligence Search algorithms and Machine learning approaches' efficiency.	U	C	Instructor-created exams / Home Assignments
CO5	Implement and analyse Machine learning algorithms to solve practical problems.	Ap	P	Writing assignments/ Exams
CO6	Apply Concepts in Real-World Projects	Ap	P	Case Study/ mini Project
* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C) # - Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge (M)				

Detailed Syllabus:

Module	Unit	Content	Hrs (45+30)	Marks (70)
I		**Introduction to Artificial Intelligence & Problem Solving and Searching**	**15**	**20**
	1	Introduction to AI – AI problems, AI Techniques	2	
	2	Various AI Domains (Introduction only)	1	
	3	Problem Solving Techniques - Search Algorithms, Knowledge representation and reasoning, constraint satisfaction problems, Game playing, Machine learning, Simulated Annealing (Concepts only)	3	
	4	Uninformed search algorithms (breadth-first, depth-first)	3	
	5	Informed search algorithms (A*, heuristic search- Generate and Test, Hill Climbing, Best First Search)	6	
II		**Knowledge Representation & Reasoning**	**10**	**15**
	6	Knowledge representation using Propositional & Predicate Logic	3	
	7	Semantic Networks & Frames	3	
	8	Rule based system & Introduction to Expert System (Concepts only)	2	
	9	Reasoning- Forward Vs Backward reasoning & logics for non-monotonic Reasoning	2	
III		**Introduction to Neural Networks**	**8**	**15**
	10	Introduction to Artificial Neural Network	1	
	11	Understanding Brain & Perceptron Model	1	
	12	Single Layer Perceptron Model & Learning in Single layer Perceptron Model	2	
	13	Multi-Layer Perceptron Model & Learning in Multi-layer Perceptron Model	2	

	14	Introduction to python packages- keras & sklearn	2	
IV		**Machine Learning Fundamentals**	**12**	**20**
	15	Introduction to Machine learning- Applications of Machine Learning	1	
	16	Supervised Machine learning- Classification & regression algorithms (Introduction: Linear Regression, Decision tree)	2	
	17	Unsupervised Machine Learning-Clustering & Dimensionality Reduction (Introduction: K means Clustering, PCA)	2	
	18	Reinforcement Learning: Elements of Reinforcement Learning	2	
	19	Feature Engineering & Feature Selection	2	
	20	Building a classification model by training with data	1	
	21	Classification model evaluation- Introduction to confusion matrix	1	
	22	Practical implementation to set up a machine learning model	1	
V	**Hands-on Artificial Intelligence & Machine Learning using Python: Practical Applications, Case Study and Course Project**		**30**	
	1	Implement the following: **1. Search algorithms** BFS DFS **2. Neural Network** Building a single layer perceptron using Keras **3. Multi-layer Neural Network** Setting up a multi-layer perceptron model **4. Supervised machine learning** Linear regression Decision tree **5. Unsupervised machine learning** K means clustering PCA **6. Feature Engineering** Feature selection from a dataset	20	
	2	Case study – AI tools / Use of AI in any movie	3	
	3	**Implementation of Comparison of any two machine learning algorithms on a dataset**	7	

References

- Elaine Rich, Kevin Knight, Shivsankar B Nair, “Artificial Intelligence”, Third Edition, Tata McGraw Hill Publisher
- Tom M. Mitchell, Machine Learning, McGraw-Hill, 1st Ed.
- Ethem Alpaydin, Introduction to Machine Learning- 3rd Edition, PHI.

Mapping of COs with PSOs and POs :

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PO1	PO2	PO3	PO4	PO5	PO6
CO 1	2	1	1	1	2	1						

CO 2	2	1	2	3	2	2						
CO 3	2	1	2	3	2	3						
CO 4	3	-	1	2	-	-						
CO 5	1	-	2	3	3	3						
CO 6	2	-	3	3	3	3						

Correlation Levels:

Level	Correlation
-	Nil
1	Slightly / Low
2	Moderate / Medium
3	Substantial / High

Assessment Rubrics:

- Quiz / Assignment/ Quiz/ Discussion / Seminar
- Midterm Exam
- Programming Assignments (20%)
- Final Exam (70%)

Mapping of COs to Assessment Rubrics :

	Internal Exam	Assignment	Practical Evaluation	End Semester Examinations
CO 1	✓	✓		✓
CO 2	✓	✓		✓
CO 3	✓	✓		✓
CO 4	✓	✓		✓
CO 5	✓	✓	✓	✓
CO 6	✓	✓	✓	

Programme	B.Sc Computer Science				
Course Code	**CSC7CJ401**				
Course Title	**Theory of Computation**				
Type of Course	**Major**				
Semester	VII				
Academic Level	400-499				
Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	4	4	-	-	60
Pre-requisites	1. Understanding of basic mathematical concepts such as sets, functions, relations, logic and discrete structures. 2. Understanding of fundamental programming constructs such as loops, conditionals, functions, and recursion.				
Course Summary	This course covers a comprehensive exploration of fundamental concepts in computer science, delving into computational models, formal language theory, and computational complexity. Students learn about various computational models such as finite automata, pushdown automata, and Turing machines, gaining insights into their capabilities and limitations. Through the study of formal languages and grammars, students understand the structure and properties of regular and context-free languages.				

Course Outcomes (CO):

CO	CO Statement	Cognitive Level*	Knowledge Category#	Evaluation Tools used
CO1	To learn and understand fundamental concepts in computational theory, including computational models such as finite automata, pushdown automata, and Turing machines.	U	P	Practical Assignment / Instructor-created exams / Quiz
CO2	To be able to classify formal language into regular, context-free, context sensitive and unrestricted languages.	Ap	P	Practical Assignment / Instructor-created exams / Quiz
CO3	To design and analyse Turing machines, their capabilities and limitations	Ap	P	Practical Assignment / Instructor-created exams / Quiz
CO4	Construct the abstract machines including finite automata, pushdown automata, and Turing machines from their associated languages and grammar	Ap	P	Practical Assignment / Instructor-created exams / Quiz
CO5	Gain insights into decidability and undecidability, and understand the limitations of computation through the study of the halting	Ap	P	Practical Assignment / Instructor-

	problem and other undecidable problems.			created exams / Quiz
CO6	Solve computational problems regarding their computability and complexity and prove the basic results of the theory of computation	E	P	Practical Assignment / Instructor-created exams / Quiz
* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C) # - Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge (M)				

Detailed Syllabus:

Theory of Computation

Module	Unit	Content	Hrs	Max Marks
I	**FINITE AUTOMATA**		**10**	**16**
	1	Formal Language: Definition, Chomsky classification of Grammar, Language and Relation, Language and Automata	2	
	2	Finite Automata: DFA, NFA with and without €- moves	2	
	3	Equivalence of DFA and NFA	2	
	4	Equivalence of NFA and €-NFA	2	
	5	Mealy and Moore Models - Conversions	2	
II	**REGULAR LANGUAGE, REGULAR EXPRESSION**		**10**	**18**
	6	Regular Languages: Regular Expressions, Ardens Theorm	2	
	7	Conversion of Regular Expression to Finite Automata	2	
	8	Closure properties of RLs	2	
	9	Pumping lemma for RLs	2	
	10	Myhill-Nerode theorem	2	
III	**PUSH DOWN AUTOMATA, CONTEXT FREE LANGUAGE**		**14**	**18**
	11	Pushdown Automata - Instantaneous Description - Transition Diagram	3	
	12	Deterministic and Non Deterministic PDA	3	
	13	Equivalence of PDAs and CFGs, Pumping lemma for CFLs	2	
	14	Closure properties of CFLs, Simplification of CFLs	2	

	15	Chomsky Normal form (CNF) and Greibach Normal form (GNF)	2	
	16	CYK algorithm for CFL membership	2	
IV	**TURING MACHINE, UNDECIDABILITY**		**14**	**18**
	17	Closure properties of CFLs, Simplification of CFLs	3	
	18	Chomsky Normal form (CNF) and Greibach Normal form (GNF)	3	
	19	CYK algorithm for CFL membership	2	
	20	Church Turing hypothesis - Rices theorem	2	
	21	Undecidability of Posts correspondence problem	2	
	22	The Class P and NP	2	
V	**Open Ended Module- Application Level**		**12**	
		1. Application of regular expressions in pattern matching and text processing. 2. Analysis of context-free languages using pumping lemma and closure properties. 3. Investigation of undecidability and un-solvability using the halting problem and Rice's theorem. 4. Notion of tractability: The Class P and NP , NP completeness of propositional satisfiability, other variants of satisfiability. NP-complete problems from other domains:		
		graphs (clique, vertex cover, independent sets, Hamiltonian cycle), number problem (partition), set cover. 5. Discussion of practical implications and applications of complexity theory.		

Textbooks

1. 1. J.E.Hopcroft, R.Motwani and J.D Ullman, ―Introduction to Automata Theory, Languages and Computations, Third Edition, Pearson Education, 2016.
2. Theory of Computer Science- Automata, Languages and Computation- K.L.P. Mishra, N Chandrasekaran, PHI

Reference books :

1. Theory of Computation, Sachin Agrawal, Vikas Publishing House

2. Micheal Sipser, ―Introduction of the Theory and Computation, Thomson Brokecole, 3rd Edition, 2013.

3. J.Martin, ―Introduction to Languages and the Theory of Computation, Third Edition, TMH, 2007. .

4. An Introduction to the Theory of Computer Science, Languages and Machines-Thomas A. Sudkamp, Third Edition, Pearson Education.

5. An Introduction to Formal languages and Automata- Peter Linz

Mapping of COs with PSOs and POs:

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PO1	PO2	PO3	PO4	PO5	PO6
CO 1	3	3	2	1	-	-						
CO 2	2	1	2	1	-	-						
CO 3	1	3	1	1	-	-						
CO 4	-	3	3	2	-	-						
CO 5	-	1	3	3	1	-						
CO 6	-	1	3	3	1	-						

Correlation Levels:

Level	**Correlation**
-	Nil
1	Slightly / Low
2	Moderate / Medium
3	Substantial / High

Assessment Rubrics:

- Quiz / Assignment/ Quiz/ Discussion / Seminar
- Midterm Exam
- Programming Assignments (20%)
- Final Exam (70%)

Mapping of COs to Assessment Rubrics:

	Internal Exam	Assignment	Practical Evaluation	End Semester Examinations
CO 1	✓	✓		✓
CO 2	✓	✓		✓
CO 3	✓	✓		✓
CO 4	✓	✓		✓
CO 5	✓	✓		✓
CO 6	✓	✓		✓

Programme	B. Sc. Computer Science				
Course Code	**CSC7CJ402**				
Course Title	**System Security**				
Type of Course	Major				
Semester	VII				
Academic Level	400 - 499				
Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	4	4	-	-	60
Pre-requisites	Knowledge in Fundamentals of Network and Operating System and COA				
Course Summary	The syllabus is prepared with the view of preparing the BSc Computer Science Graduates to build effective an understanding of the differences between various forms of computer system security, where they arise, and appropriate tools to achieve them				

Course Outcomes (CO):

CO	**CO Statement**	**Cognitive Level***	**Knowledge Category#**	**Evaluation Tools used**
CO1	Understand the different types of securities in information and computer systems, security goals and confidentiality, integrity, availability	U	C	Instructor-created exams / Quiz
CO2	Outline computer system threats and various types of system attacks	U	C	Instructor-created exams / Quiz
CO3	Identify different issues associated with system attacks and how attacking occurs and various types of attackers	U	P	Instructor-created exams / Quiz
CO4	Provide knowledge in operating system security, file protections, security assurance	U	C	Instructor-created exams / Case studies
CO5	Understand important elements of Database security	U	P	Instructor-created exams / Quiz Case studies
CO6	Define security planning, various types of security policies and risk analysis	U	P	Instructor-created exams / Quiz / Case studies
* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C) # - Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge (M)				

Module	Un it	Content	Hrs	Marks
I	**Notion of different types of securities**		**12**	**15**
	1	Information security - computer security - security goals, relation between security, confidentiality, integrity, availability and authorization, vulnerabilities - principles of adequate protection.	3	
	2	Notions of operating security, database security, program security, network security attacks - threats, vulnerabilities and controls.	3	
	3	The kind of problems - interception, interruption, modification, fabrication.	2	
	4	Computer criminals - amateurs, crackers, career criminals.	2	
	5	Methods of defense control, hardware controls, software controls, effectiveness of controls.	2	
II	**Program security**		**12**	**15**
	6	Secure programs - fixing faults, unexpected behaviour, types of flaws.	2	
	7	Non-malicious program errors - buffer overflows, incomplete mediation.	1	
	8	Viruses and other malicious code - kinds of malicious code, how viruses attach, how viruses gain control, prevention,	3	
	9	Control example - the brain virus, the internet worm, web bugs..	3	
	10	Targeted malicious code - trapdoors, Salami attack	1	
	11	Controls against program threats - development controls, peer reviews, hazard analysis	2	
III	**Operating system security**		**12**	**20**
	12	Protected objects and methods of protection - memory address protection - fence, relocation, base/bounds registers, tagged architecture, segmentation, paging.	2	
	13	Control of access to general objects - directory, access control list	2	
	14	File protection mechanism - basics forms of protection, single permissions.	2	

	15	Authentication - authentication basics, password, authentication process challenge - response, biometrics	2	
	16	Trusted operating systems - security policies for operating systems	2	
	17	Models of security - requirement of security systems, multilevel security, access security, limitations of security systems	2	
IV	**Database Security**		**12**	**20**
	18	Security requirements - integrity, confidentiality and availability of database	2	
	19	Reliability and integrity of database	2	
	20	Sensitive data, interface	3	
	21	Multilevel database	2	
	22	Proposals for multilevel database security	3	
V	**Open Ended Module**		**12**	
	CASE STUDY: Administrating security Security planning – Contents of a security planning, team members, commitment to a security plan, business continuity plans. Risk analysis – the nature of risk, steps of risk analysis.			

Reference Books:

1. C. P. Pfleeger and S. L. Pfleeger, Security in Computing, 4th Edition, Pearson India, ISBN: 9788131727256.
2. Matt Bishop, Computer Security: Art & Science, 1st Edition, Pearson, ISBN: 0201440997.
3. William Stallings, Cryptography and Network Security: Principles and Practice, 6th Edition, Pearson India, ISBN: 9332518777.
4. Michael E. Whitman and Herbert J. Mattord, Principles of Information Security, 4th Edition, Ceneage Learning India Pvt Ltd, ISBN: 8131516458.

Mapping of COs with PSOs and POs :

	PSO1	PSO 2	PSO 3	PSO4	PSO 5	PSO6	PO1	PO2	PO3	PO4	PO5	PO6

CO 1	-	2	-	-	1	1						
CO 2	-	3	1	-	1	1						
CO 3	-	2	1	-	1	1						
CO 4	-	2	1	-	1	1						
CO 5	-	3	1	-	1	2						
CO 6	-	2	1	1	1	2						

Correlation Levels:

Level	**Correlation**
-	Nil
1	Slightly / Low
2	Moderate / Medium
3	Substantial / High

Assessment Rubrics:

- Quiz / Assignment/ Quiz/ Discussion / Seminar
- Midterm Exam
- Assignments (20%)
- Final Exam (70%)

Mapping of COs to Assessment Rubrics :

	Internal Exam	Assignment	Project Evaluation	End Semester Examinations
CO 1	✓			✓
CO 2	✓			✓
CO 3	✓	✓		✓
CO 4		✓		✓
CO 5		✓		✓
CO 6			✓	

Programme	B. Sc. Computer Science				
Course Code	**CSC7CJ403**				
Course Title	**Advanced Data Structures and algorithms**				
Type of Course	**Major**				
Semester	VII				
Academic Level	400-499				
Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	4	3	-	2	75
Pre-requisites	1. Fundamental Mathematics Concepts: Sets, matrices 2. Awareness of Data structures and operations like array, stack, queue 3. Fundamentals of Java, C Programming				
Course Summary	This course provides an introduction to the ideas, techniques, and applications of advanced data structures) is given in this course. The advanced data structures and its variants like tree, graph, heaps are covered in this syllabus.				

Course Outcomes (CO):

CO	CO Statement	Cognitive Level*	Knowledge Category#	Evaluation Tools used
CO1	Understand the concepts of advanced data structures like tree, graphs, heaps.	U	C	Instructor-created exams / Quiz
CO2	Understand familiarity with algorithmic techniques such as brute force, greedy, and divide and conquer.	U	C	Practical Assignment / Observation of Practical Skills
CO3	Understand Asymptotic analysis (big-O notation, time and space complexity).	U	F	Seminar Presentation / Group Tutorial Work/ Viva Voce
CO4	Application of advanced abstract data type (ADT) and data structures in solving real world problems.	AP	P	Instructor-created exams / Home Assignments

CO5	Effectively combine fundamental data structures and algorithmic techniques in building a complete algorithmic solution to a given problem	Ap	P	Writing assignments/ Instructor-created exams/ practicals
CO6	Apply Concepts of data structures in real world problem solving	Ap	P	Case Study/ mini Project/ practicals
* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C) # - Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge (M)				

Detailed Syllabus:

Module	Unit	Content	Hrs (45+30)	Marks (70)
I	**Introduction to data structures and analysis of Quality of an algorithm**		**9**	**12**
	1	Data structure - definition - types & operations, characteristics of data structures	2	
	2	Abstract Data Type (ADT) – algorithms - concepts - definition - objectives of algorithms -	1	
	3	Quality of an algorithm - space complexity and time complexity of an algorithm.	2	
	4	Growth of Functions: Asymptotic notations, Cost estimation based on key operations- Big Oh, Big Omega, Little Oh, Little Omega and Theta notations	3	
	5	Algorithm Design: Introduction, Steps in developing algorithm, Methods of specifying an algorithm	1	
II	**Basic Technique for Design of Efficient Algorithm**		**11**	**15**
	6	Brute Force approach (String pattern matching)	1	
	7	Divide-and-Conquer approach (Merge sort)	1	
	8	Branch-and-Bound technique (Knapsack problem)	2	
	9	Greedy approach (Kruskal's algorithm and Prim's Algorithm)	3	

	10	Dynamic Programming (Longest Common Subsequence)	2	
	11	Backtracking (Sum of subsets problem)	2	
III	**Linked lists - operations and implementations**		**12**	**15**
	12	Introduction to Singly Linked list and its operations	2	
	13	Circular Linked list and its operations	3	
	14	Doubly Linked list and its operations	2	
	15	Circular Doubly Linked list and its operations	2	
	16	Recursive lists, heterogeneous lists, deterministic skip lists- Creation & Searching	3	
IV		**Non-linear Data Structures**	**13**	**20**
	17	Binary search trees - traversals and operations on BST	3	
	18	AVL tree, Red Black Tree (concept only)	2	
	19	Balanced trees - M-way trees - B Tree (Concepts only)	1	
	20	Graphs - representation of graphs	1	
	21	Graphs- operations - traversals and their implementation.	2	
	22	Heap structures- Min-Max heaps - Deaps - leftist heaps - binomial heaps (concepts only) - applications	3	
V	**Practical Implementations of Data structures and its operations in Java or C programming Language**		**30**	**20**
	1	• Implementation of linear linked list • Implementation of circular linked list • Implementation of doubly linked list • Implementation of BST operations • Implementation of Depth First Search using graph • Implementation of Breadth First Search using graph • Implementation of max heap and delete a node from it. • Sort a set of data using Heap tree	25	
	2	Case Study/ Project	5	

References

1. Alfred V. Aho, John E. Hopcroft and Jeffrey D. Ullman, Data Structures and Algorithms, Addison-Wesley, ISBN: 978-0201000238.

2. Horowitz E and Sahni S, Fundamentals of Data Structures, Computer Science Press, ISBN: 9780716780427.
3. Ellis Horowitz, Sartaj Sahni and Susan Anderson-Freed, Fundamentals of Data Structures in C, Silicon Press, ISBN: 0929306406.
4. Thomas H Cormen, Charles E Leiserson, and Ronald L Rivest, Introduction to Algorithms, 3rd Edition, Prentice Hall of India Private Limited, New Delhi, ISBN: 9780262033848
5. Alfred V. Aho, John E. Hopcroft and Jeffrey D. Ullman, The Design and Analysis of Computer Algorithms, 1st Edition. Addison Wesley, ISBN: 0534915728

Mapping of COs with PSOs and POs :

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PO1	PO2	PO3	PO4	PO5	PO6
CO 1	2	1	3	3	2	1						
CO 2	2	1	3	3	2	1						
CO 3	2	1	3	3	2	1						
CO 4	3	1	3	3	-	-						
CO 5	1	1	3	3	3	1						
CO 6	2	1	3	3	3	1						

Correlation Levels:

Level	**Correlation**
-	Nil
1	Slightly / Low
2	Moderate / Medium
3	Substantial / High

Assessment Rubrics:

- Quiz / Assignment/ Quiz/ Discussion / Seminar
- Midterm Exam
- Programming Assignments (20%)
- Final Exam (70%)

Mapping of COs to Assessment Rubrics :

	Internal Exam	Assignment	Practical Evaluation	End Semester Examinations

CO 1	✓	✓	✓	✓
CO 2	✓	✓	✓	✓
CO 3	✓	✓	✓	✓
CO 4	✓	✓	✓	✓
CO 5	✓	✓	✓	✓
CO 6	✓	✓	✓	

Programme	B. Sc. Computer Science				
Course Code	**CSC7CJ404**				
Course Title	**Blockchain Technology**				
Type of Course	Major				
Semester	VII				
Academic Level	400 - 499				
Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	4	4	-	-	60
Pre-requisites	Strong programming skills in at least one popular language, such as Java or Python. Knowledge of cryptography and data structures (like linked lists and arrays). Good understanding of networking concepts				
Course Summary	The syllabus is prepared with the view of preparing the BSc Computer Science Graduates to create awareness and understanding among students on the foundation of blockchain technology. The course introduces the cryptographic principles behind blockchain and helps the students understand concepts like consensus, crypto-currency, smart contracts, use cases etc.				

Course Outcomes (CO):

CO	CO Statement	Cognitive Level*	Knowledge Category#	Evaluation Tools used
CO1	Understand the basics of cryptographic building blocks in blockchain technology.	U	C	Instructor-created exams / Quiz
CO2	Explain the fundamental concepts of blockchain technology.	U	C	Instructor-created exams / Quiz
CO3	Summarize the classification of consensus algorithms	U	P	Instructor-created exams / Quiz
CO4	Explain the concepts of first decentralized cryptocurrency bitcoin	U	C	Instructor-created exams / Case studies

CO5	Describe the use of smart contracts and its use cases	U	P	Instructor-created exams / Quiz Case studies
CO6	Develop simple block chain applications	U	P	Instructor-created exams / Quiz / Case studies
* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C) # - Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge (M)				

Detailed Syllabus:

Module	Un it	Content	Hrs	Marks
I	**Fundamentals of Cryptography**		**12**	**15**
	1	Introduction to Cryptography, Symmetric cryptography – AES. Asymmetric cryptography –RSA. Elliptic curve cryptography,	3	
	2	Digital signatures – RSA digital signature algorithms.	2	
	3	Secure Hash Algorithms – SHA-256.	2	
	4	Applications of cryptographic hash functions – Merkle trees	3	
	5	Distributed hash tables	2	
II	**Fundamentals of Blockchain Technology**		**12**	**15**
	6	Blockchain – Definition, architecture, elements of blockchain, benefits and limitations.	2	
	7	Types of blockchain	1	
	8	Consensus – definition, types, consensus in blockchain,	3	
	9	Decentralization – Decentralization using blockchain	3	
	10	Methods of decentralization, Routes to decentralization,	1	
	11	Blockchain and full ecosystem decentralization	2	
III	**Consensus Algorithms and Bitcoin**		**12**	**20**
	12	Consensus Algorithms, Crash fault-tolerance (CFT) algorithms – Paxos, Raft. Byzantine fault tolerance(BFT) algorithms – Practical Byzantine Fault Tolerance (PBFT),.	2	
	13	Proof of work (PoW),Proof of stake (PoS), Types of PoS	2	

	14	Bitcoin – Definition, Cryptographic keys – Private keys, public keys, addresses	2	
	15	Transactions –Lifecycle, Coinbase transactions, transaction validation Blockchain – The genesis block.	2	
	16	Mining – Tasks of miners, mining algorithm, hash rate	2	
	17	Wallets – Types of wallets..	2	
IV	**Smart Contracts and Use cases**		**12**	**20**
	18	Smart Contracts – Definition, Smart contract templates, Deploying smart contracts	2	
	19	Oracles, Types of oracles.	2	
	20	Decentralization terminology – Decentralized applications, Decentralized Autonomous Organizations	3	
	21	Use cases of Blockchain technology – Government, Health care, Finance, Supply chain management.	2	
	22	Blockchain and allied technologies – Blockchain and Cloud Computing, Blockchain and Artificial Intelligence	3	
V	**Open Ended Module**		**12**	
	CASE STUDY: BLOCKCHAIN TECHNOLOGY Solidity language Ethereum platform			

Reference Books:

1. Imran Bashir, Mastering Blockchain: A deep dive into distributed ledgers, consensus protocols, smart contracts, DApps, cryptocurrencies, Ethereum, and more, Packt Publishing, Third edition, 2020.

2. Ritesh Modi, Solidity Programming Essentials: A beginner's guide to build smart contracts for Ethereum and blockchain, Packt Publishing, First edition, 2018.

3. Kumar Saurabh, Ashutosh Saxena, Blockchain Technology: Concepts and Applications, First Edition, Wiley Publications, First edition, 2020.

4. Chandramouli Subramanian, Asha A George, et al, Blockchain Technology, Universities Press (India) Pvt. Ltd, First edition, August 2020

5. Lorne Lantz, Daniel Cawrey, Mastering Blockchain: Unlocking the Power of Cryptocurrencies, Smart Contracts, and Decentralized Applications, O'Reilly Media, First edition, 2020.
6. Andreas M. Antonopoulos, Gavin Wood, Mastering Ethereum: Building Smart Contracts and DApps, O'Reilly Media, First edition, 2018.

Mapping of COs with PSOs and POs :

	PSO1	PSO 2	PSO 3	PSO4	PSO 5	PSO6	PO1	PO2	PO3	PO4	PO5	PO6
CO 1	1	2	-	-	-	-						
CO 2	-	2	-	-	-	-						
CO 3	-	2	3	3	-	-						
CO 4	-	2	3	3	1	1						
CO 5	-	1	1	-	2	3						
CO 6	-	1	1	-	2	3						

Correlation Levels:

Level	**Correlation**
-	Nil
1	Slightly / Low
2	Moderate / Medium
3	Substantial / High

Assessment Rubrics:

- Quiz / Assignment/ Quiz/ Discussion / Seminar
- Midterm Exam
- Assignments (20%)
- Final Exam (70%)
-

Mapping of COs to Assessment Rubrics :

	Internal Exam	Assignment	Project Evaluation	End Semester Examinations
CO 1	✓			✓
CO 2	✓			✓
CO 3	✓	✓		✓
CO 4		✓		✓
CO 5		✓		✓
CO 6			✓	

Programme	B. Sc. Computer Science				
Course Code	**CSC7CJ405**				
Course Title	**Internet of Things**				
Type of Course	**Major**				
Semester	VII				
Academic Level	400 - 499				
Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	4	3	-	2	75
Pre-requisites	1. Understanding of Computer Networks, Hardware and Sensors. 2. Basic Programming Skills. 3. Understanding of basic Internet Terminologies				
Course Summary	The course covers a comprehensive overview of Internet of Things (IoT) concepts, technologies, and architectures. It begins by explaining the foundational framework and architectural views of IoT systems, highlighting machine-to-machine (M2M) communication and various IoT sources. Design principles for connected devices are emphasized, focusing on IoT system layers and standardization. Communication technologies, data enrichment, and device management at gateways are explored in detail. The importance of internet connectivity principles, web connectivity design, and protocols are discussed. Additionally, the syllabus delves into message communication protocols and networking using gateways. Sensor technology, participatory sensing, and industrial and automotive IoT applications are also covered. Furthermore, it includes embedded computing basics, platforms like Arduino and Raspberry Pi. Finally, cloud Platform as a Service (PaaS) as well as essential networking technologies are introduced.				

Course Outcomes (CO):

CO	CO Statement	Cognitive Level*	Knowledge Category#	Evaluation Tools used
CO1	Understanding the concepts and architecture of IoT involves grasping the fundamental principles and interconnected structures of its diverse components.	U	F	Instructor-created exams / Quiz
CO2	Understanding the hardware components of IoT involves recognizing sensors, actuators, communication modules, and processing units, crucial for data collection, transmission, and analysis.	U	C	Practical Assignment / Observation of Practical Skills
CO3	Explain the design principles for connected devices, focusing on IoT system layers and standardization. Communication technologies, data enrichment, and device management at gateways	U	C	Practical Assignment / Observation of Practical Skills

CO4	Perceive the basic protocols in IoT, that enable efficient communication between devices, facilitating data exchange and interoperability within IoT networks.	Ap	C	Instructor-created exams / Quiz
CO5	Demonstrating IoT utilizes popular hardware and software platforms showcasing practical implementations of interconnected devices.	Ap	P	Practical Assignment / Observation of Practical Skills
CO6	Implementing IoT in real-time situations by deploying interconnected devices to collect, process, act upon data and visualize them..	Ap	P	Practical Assignment / Observation of Practical Skills
* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C) # - Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge (M)				

Detailed Syllabus:

Module	Unit	Content	Hrs	Marks
I	**Internet of Things: An Overview**		**9**	**15**
	1	Internet of Things (IoT) - Conceptual Framework, Architectural View	2	
	2	IoT Technology, IoT Sources, M2M Communication, Examples of IoT	2	
	3	Design Principles for Connected Devices - Introduction , IoT/M2M System Layers and Design Standardisation	2	
	4	Communication Technologies, Data Enrichment	1	
	5	Data Consolidation and Device Management at Gateway	2	
II	**Internet Connectivity Principles**		**9**	**18**
	6	Internet Connectivity, Internet-based Communications, IP Addressing in the IOT	2	
	7	Media Access Control, Application Layer Protocols, HTTP, HTTPS, FTP	2	
	8	Web Communication protocols for Connected Devices	1	
	9	Message Communication Protocols for Connected Devices	1	
	10	Network using Gateway	1	

	11	SOAP, REST, HTTP RESTful and Web Sockets	2	
III	**Sensors, Participatory Sensing, RFIDs, and Wireless Sensor Networks**		**12**	**17**
	12	Sensor Technology, Participatory Sensing	3	
	13	Industrial IoT, Automotive IoT	3	
	14	Actuators, Sensor Data Communication Protocols	3	
	15	RF Identification Technology, Wireless Sensor Networks Basics	3	
IV	**Basics of Embedded Computing**		**15**	**20**
	16	Introduction, Embedded Hardware Unit	2	
	17	Basics of Embedded Platforms for Prototyping - Arduino, Raspberry Pi	2	
	18	Basics of Embedded Platforms for Prototyping - Intel Galileo, Intel Edison, BeagleBone, mBed	2	
	19	Prototyping Embedded Device Software for IoT using Arduino	3	
	20	Understanding Cloud PaaS - Xively, Nimbits, IBM Bluemix, CISCO IoT, AWS IoT, TCS Connected AWS Platform	2	
	21	Basics of popular technologies - IEEE 802.15.4, Zigbee Specification, WiFi, 6LowPAN, IPv6, LoRaWAN	2	
	22	IoT - Vulnerabilities, Security and Threats	2	
V	**Practical Activities**		**30**	

		Following are some of the suggested practical activities. 1.Temperature and Humidity Monitoring System: Use sensors to measure temperature and humidity. Connect the sensor to Arduino or Raspberry Pi and send the data to a cloud platform . Visualize the data in real-time on a web dashboard. 2. Smart Home Automation: Control home appliances such as lights, fans etc. using Arduino or Raspberry Pi. Interface relay modules or solid-state relays with the microcontroller to control the appliances remotely. Use MQTT protocol for communication and control the devices via a mobile app or web interface. 3. Weather Station: Build a weather station using sensors like BMP180 or BME280 for measuring temperature, pressure, and altitude. Interface the sensors with Arduino or Raspberry Pi and log the data to a cloud platform. Display weather data on an LCD screen or visualize it on a web dashboard. 4. Smart Plant Monitoring System: Monitor the soil moisture level, light intensity, and temperature to create a smart plant monitoring system. Use sensors like soil moisture sensors, LDRs, and temperature sensors with Arduino or Raspberry Pi. Send notifications to users when plants need watering or when light conditions are inadequate. 5. Home Security Camera System: Build a home security camera system using Raspberry Pi and a USB webcam or Raspberry Pi Camera Module. Stream live video footage over the network and access it remotely using a web browser or mobile app. Implement motion detection algorithms to trigger recording or notifications.	30	

Mapping of COs with PSOs and POs:

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PO1	PO2	PO3	PO4	PO5	PO6
CO 1	1	3	-	-	-	-						
CO 2	1	3	-	-	-	-						
CO 3	1	3	-	-	-	-						
CO 4	1	2	-	-	-	-						
CO 5	1	2	3	3	-	3						
CO 6	1	2	3	3	-	3						

Correlation Levels:

Level	**Correlation**
-	Nil
1	Slightly / Low
2	Moderate / Medium
3	Substantial / High

Assessment Rubrics:

- Quiz / Assignment/ Quiz/ Discussion / Seminar
- Midterm Exam
- Programming Assignments (20%)
- Final Exam (70%)

Mapping of COs to Assessment Rubrics :

	Internal Exam	Assignment	Practical Evaluation	End Semester Examinations
CO 1	✓	✓		✓
CO 2	✓	✓	✓	✓
CO 3	✓	✓	✓	✓
CO 4		✓	✓	✓
CO 5		✓		✓
CO 6	✓	✓	✓	✓

References:

1. Rajkamal, “Internet of Things : Architecture and Design Principles”, McGraw Hill (India) Private Limited.

2. Arshadeep Bahga, Vijay Madisetti, “Internet of Things: A hands-on approach”, University Press, 2015 (First edition)

3. Dieter Uckelmann, Mark Harrison, Michahelles Florian (Ed.), Architecting the internet of things, Springer, 2011

4. Dr. Ovidiu Vermesan, Dr. Peter Friess, Internet of Things: Converging Technologies for Smart Environments and Integrated Ecosystems, River Publishers, 2013

5. C. Doukas, "Building Internet of Things with the Arduino," Packt Publishing, 2016.

6. Dr. Ovidiu Vermesan, Dr. Peter Friess, Internet of Things: Converging Technologies for Smart Environments and Integrated Ecosystems, River Publishers, 2013

7. Simon Monk, “Programming Arduino: Getting Started with Sketches”, McGraw Hill Publications

Programme	B.Sc Computer Science				
Course Code	**CSC8CJ406**				
Course Title	**Compiler Design**				
Type of Course	Major				
Semester	VIII				
Academic Level	400-499				
Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	4	4	-	-	60
Pre-requisites	1. Formal Languages & Automata Theory. 2. Data Structure and Algorithms				
Course Summary	This course covers the fundamental concepts of different phases of compilation such as lexical analysis, syntax analysis, semantic analysis, intermediate code generation, code optimization and code generation. Students can apply this knowledge in design and development of compilers.				

Course Outcomes (CO):

CO	CO Statement	Cognitive Level*	Knowledge Category#	Evaluation Tools used
CO1	To identify different phases in compilation process and model a lexical analyser.	Ap	P	Practical Assignment / Instructor-created exams / Quiz
CO2	To model language syntax using Context Free Grammar and develop parse tree representation using leftmost and rightmost derivations.	Ap	P	Practical Assignment / Instructor-created exams / Quiz
CO3	To compare different types of parsers and construct parser for a given grammar.	Ap	P	Practical Assignment / Instructor-created exams / Quiz
CO4	To build Syntax Directed Translation for a context free grammar, compare various storage allocation strategies and classify intermediate representations.	Ap	P	Practical Assignment / Instructor-created exams / Quiz
CO5	Students will demonstrate the ability to design and implement lexical analyzers to recognize tokens in source programs.	Ap	P	Practical Assignment / Instructor-created exams / Quiz

CO6	Illustrate code optimization and code generation techniques in compilation	Ap	P	Practical Assignment / Instructor-created exams / Quiz
* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C) # - Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge (M)				

Detailed Syllabus:

Module	Unit	Content	Hrs	Max Marks
I		**COMPILERS AND LEXICAL ANALYSIS**	**10**	**15**
	1	Analysis of the source program - Analysis and synthesis phases	2	
	2	Phases of a compiler, The grouping of Phases	2	
	3	Compiler writing tools. Bootstrapping.	2	
	4	Lexical Analysis : Parsing, Abstract stack machine, Role of Lexical Analyser	2	
	5	Input Buffering, Specification of Tokens, Recognition of Tokens.	2	
II		**SYNTAX ANALYSIS**	**18**	**25**
	6	Role of the Syntax Analyser, Role of the Parser	2	
	7	Context-free grammars, Parse Tree and Derivations, Eliminating Ambiguity	2	
	8	Basic Parsing Approaches – Eliminating left recursion, left factoring	3	
	9	Top Down parsing - Recursive Descent Parsing	2	
	10	Predictive Parsing - LL(1) Grammars	3	
	11	Bottom-up parsing -Handle Pruning - Shift Reduce Parsing - Operator Precedent Parsing	3	
	12	LR Parsers - SLR Parser- Canonical LR Parser - LALR Parser	3	
III		**SEMANTIC ANALYSIS AND INTERMEDIATE CODE GENERATION**	**10**	
	13	Syntax directed translation - Syntax directed definitions	2	**15**
	14	S-attributed definitions, L-attributed definitions, Bottom-up evaluation of S-attributed definitions. Run-Time Environments	2	
	15	Source Language issues, Storage organization, Storage-allocation strategies.	2	
	16	Intermediate Code Generation - Intermediate languages, Graphical representations,	2	
	17	Three-Address code, Quadruples, Triples.	2	
IV		**CODE OPTIMIZATION AND CODE GENERATION**	**10**	**15**
	18	Code Optimization - Principal sources of optimization	2	
	19	Machine dependent and machine independent optimizations,	2	

	20	Local and global optimizations.	2	
	21	Code generation - Issues in the design of a code generator,	2	
	22	Target Language, A simple code generator.	2	
V		**Open Ended Module – Application Level**	**12**	
		1. Learn the fundamentals of lexical analysis and parsing using Lex and Yacc, essential tools in compiler construction. 2. Apply the concepts learned to develop a small compiler, progressively enhancing its functionality while implementing error handling and optimization strategies. 3. Apply the concept of Bootstrapping and its significance in compiler construction.		
		4. Understanding of run-time environments and storage allocation strategies. 5. Development of a simple code generator for translating intermediate code into target code.		

Textbooks :

1. Aho A.V., Ravi Sethi and D. Ullman. Compilers – Principles Techniques and Tools, Addison Wesley, 2006.

Reference books :

1. D.M.Dhamdhere, System Programming and Operating Systems, Tata McGraw Hill &Company, 1996.
2. Kenneth C. Louden, Compiler Construction – Principles and Practice, Cengage Learning Indian Edition, 2006.
3. Tremblay and Sorenson, The Theory and Practice of Compiler Writing, Tata McGraw Hill & Company,1984.
4. Compiler Design in C, Allen I. Holub, Prentice Hall (Software Series).
6. Crafting a Compiler with C, C. N. Fischer and R. J. LeBlanc, Pearson Education.
7. Allen I Holub, Compiler Design in C, 1st Edition, PHI Learning Pvt Ltd.

Mapping of COs with PSOs and POs:

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PO1	PO2	PO3	PO4	PO5	PO6
CO 1	2	1	2	2	2	-						
CO 2	2	1	2	2	2	-						
CO 3	2	1	2	3	3	-						
CO 4	2	-	2	3	3	-						

CO 5	2	-	2	2	2	-					
CO 6	-	-	2	1	2	-					

Correlation Levels:

Level	Correlation
-	Nil
1	Slightly / Low
2	Moderate / Medium
3	Substantial / High

Assessment Rubrics:

- Quiz / Assignment/ Quiz/ Discussion / Seminar
- Midterm Exam
- Programming Assignments (20%)
- Final Exam (70%)

Mapping of COs to Assessment Rubrics:

	Internal Exam	Assignment	Practical Evaluation	End Semester Examinations
CO 1	✓	✓		✓
CO 2	✓	✓		✓
CO 3	✓	✓		✓
CO 4	✓	✓		✓
CO 5	✓	✓		✓
CO 6	✓	✓		✓

Programme	B. Sc. Computer Science				
Course Code	**CSC8CJ407**				
Course Title	**Client Server Architecture**				
Type of Course	Major				
Semester	VIII				
Academic Level	400 - 499				
Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	4	4	-	-	60
Pre-requisites	Knowledge in Fundamentals of Network and Operating System				
Course Summary	The syllabus is prepared with the view of preparing the BSc Computer Science Graduates to build effective Client/Server applications. This course aims at providing a foundation in decentralized computer systems, using the client/server model. The course content is decided to cover the essential fundamentals which can be taught within the given slots in the curriculum.				

Course Outcomes (CO):

CO	CO Statement	Cognitive Level*	Knowledge Category#	Evaluation Toolsused
CO1	Understand the basics of client/server systems and the driving force behind the development of client/server systems.	U	C	Instructor-created exams / Quiz
CO2	Outline the architecture and classifications of client/server systems	U	C	Instructor-created exams / Quiz
CO3	Choose the appropriate client/server network services for a typical application	U	P	Instructor-created exams / Quiz
CO4	Describe management services	U	C	Instructor-created exams / Case studies
CO5	Describe issues in network	U	P	Instructor-created exams / Quiz Case studies
CO6	Apply various services and support	U	P	Instructor-created exams / Quiz /Case studies
* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C)# - Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge (M)				

Detailed Syllabus:

Module	Unit	Content	Hrs	Marks
I		**Introduction to Client/Server computing**	**12**	**15**
	1	Introduction to Client/Server computing - Driving forces behind Client/ Server, Client/ Server development tools	2	
	2	Development of client/server systems, Client/Server security	2	
	3	Organizational Expectations, Improving performance of client/server applications	2	
	4	Single system image, Downsizing and Rightsizing	3	
	5	Advantages of client server computing, Applications of Client/Server	3	
II		**Client/Server Application Components**	**12**	**15**
	6	Classification of Client/Server Systems- Two-Tier Computing, Middleware, Three-Tier Computing	2	
	7	Model View Controller (MVC)	1	
	8	Principles behind Client/Server Systems	3	
	9	Client/Server Topologies	3	
	10	Existing Client/Server Architecture.	1	
	11	Architecture for Business Information System	2	
III		**Client/ Server Systems Development**	**12**	**20**
	12	Client- Services, Request for services, RPC, Windows services, Print services, Remote boot services, other remote services, Utility Services.	2	
	13	Dynamic Data Exchange (DDE).	2	
	14	Object Linking and Embedding (OLE).	2	
	15	Common Object Request Broker Architecture (CORBA).	2	
	16	Server- Detailed server functionality	2	
	17	Network operating system, Available platforms, Server operating system.	2	
IV		**Client/ Server Systems Development**	**12**	**20**
	18	Services and Support- System administration, Availability, Reliability, Scalability, Observability, Agility, Serviceability.	2	
	19	Software Distribution, Performance, Network management.	2	
	20	Remote Systems Management- RDP, Telnet, SSH, Security.	3	
	21	LAN and Network Management issues, Training, Connectivity,	2	
	22	Communication interface technology, Inter process communication,	3	
V		**Open Ended Module**	**12**	
		CASE STUDY: Client Server Architecture Generic Client/Server Classes Client/Server Communication via Sockets The Server Protocol The Client Protocol A Two-Way Stream Connection		

1. Patrick Smith & Steave Guengerich, “Client / Server Computing”, PHI
2. Dawna Travis Dewire, “Client/Server Computing”, TMH
3. Jeffrey D.Schank, “Novell’s Guide to Client-Server Application & Architecture” Novell Press
4. Robert Orfali, Dan Harkey, Jeri Edwards, Client/Server Survival Guide, Wiley-India Edition, Third Edition

Mapping of COs with PSOs and POs :

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PO1	PO2	PO3	PO4	PO5	PO6
CO1	-	2	-	-	-	-						
CO2	-	3	-	-	-	-						
CO3	-	2	1	-	-	-						
CO4	-	2	1	1	1	1						
CO5	-	2	1	-	-	-						
CO6	-	2	1	-	1	1						

Correlation Levels:

Level	Correlation
-	Nil
1	Slightly / Low
2	Moderate / Medium
3	Substantial / High

Assessment Rubrics:

- Quiz / Assignment/ Quiz/ Discussion / Seminar
- Midterm Exam
- Assignments (20%)
- Final Exam (70%)

Mapping of COs to Assessment Rubrics :

	Internal Exam	Assignment	Project Evaluation	End Semester Examinations
CO 1	✓			✓
CO 2	✓			✓
CO 3	✓	✓		✓
CO 4		✓		✓
CO 5		✓		✓
CO 6			✓	

Programme	B. Sc. Computer Science
Course Code	**CSC8CJ408**
Course Title	**Parallel Computing**
Type of Course	Major
Semester	VIII
Academic Level	400 - 499

Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	4	4	-	-	60

Pre-requisites	Knowledge in Fundamentals of COA and Operating System
Course Summary	The syllabus is prepared with the view of preparing the BSc Computer Science Graduates to understand basic and advanced concepts of parallel computing. It covers Principles of Parallel Algorithm Design, Communication operations, Programming Using the Message Passing Paradigm, Programming Shared Address Space Platforms, Thread Basics,

Course Outcomes (CO):

CO	CO Statement	**Cognitive Level***	**Knowledge Category#**	**Evaluation Toolsused**
CO1	Summarize the key parallel computational models	U	C	Instructor-created exams / Quiz
CO2	Appreciate and apply parallel and distributed algorithms in problem Solving	U	C	Instructor-created exams / Quiz
CO3	Appreciate the communication models for parallel algorithm development	U	P	Instructor-created exams / Quiz
CO4	Develop parallel algorithms using message passing paradigm	U	C	Instructor-created exams / Case studies
CO5	Formulate parallel algorithms for shared memory architectures	U	P	Instructor-created exams / Quiz Case studies
CO6	Understand thread management	U	P	Instructor-created exams / Quiz /Case studies
* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C)# - Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge (M)				

Detailed Syllabus:

Module	**Unit**	**Content**	**Hrs**	**Marks**

I		**Principles of Parallel Algorithm Design**	**12**	**15**
	1	Parallel Processing platforms. Preliminaries, Decomposition Techniques,	2	
	2	Characteristics of Tasks and Interactions	2	
	3	Mapping Techniques for Load Balancing	2	
	4	Methods for Containing Interaction Overheads	3	
	5	Parallel Algorithm Models.	3	
II		**Communication Operations**	**12**	**15**
	6	Basic Communication Operations - One-to-All Broadcast and All-to-One Reduction	2	
	7	All-to-All Broadcast and Reduction	1	
	8	All-Reduce and Prefix-Sum Operations	3	
	9	Scatter and Gather	3	
	10	All-to-All Personalized Communication, Circular Shift	1	
	11	Improving the Speed of Some Communication Operation	2	
III		**Programming Using the Message Passing Paradigm**	**12**	**20**
	12	Principles of Message-Passing Programming, The Building Blocks: Send Operations	2	
	13	Receive Operations	2	
	14	MPI: The Message Passing Interface	2	
	15	Overlapping Communication with Computation	2	
	16	Collective Communication and Computation Operations	2	
	17	Groups and Communicators	2	
IV		**Programming Shared Address Space Platforms Thread Basics**	**12**	**20**
	18	Thread Basics, Why Threads? The POSIX Thread Application Programme Interface, Synchronization Primitives in POSIX, Controlling Thread and Synchronization Attributes	2	
	19	Thread Cancellation, Composite Synchronization Constructs	2	
	20	OpenMP: a Standard for Directive Based Parallel Programming, Specifying Concurrent Tasks in OpenMP	3	
	21	Synchronization Constructs in OpenMP	2	
	22	OpenMP Applications: Parallel algorithm development for Matrix multiplication	3	
V		**Open Ended Module**	**12**	
		CASE STUDY: PARALLEL COMPUTING Heterogeneous Parallel Computing Data parallel computing Device Global Memory and Data Transfer Kernel Functions and Threading		

Reference Books:

1. Ananth Grama, Anshul Gupta, George Karypis, Vipin Kumar, Introduction to Parallel Computing, 2nd Ed, Addison-Wesley, 2003

2. David B. Kirk, Wen-mei W. Hwu, Programming Massively Parallel Processors: A Hands-on Approach, 3rd Ed., Morgan Kaufman, 2016. References

3. Steven Brawer, Introduction to Parallel Computing, Academic Press, (1989)

4. Barbara Chapman, Gabriele Jost, Ruud van der Pas, Using OpenMP: Portable Shared Memory Parallel Programming , MIT Press, 2008.

5. William Gropp, Ewing Lusk, Anthony Skjellum Using MPI: Portable Parallel Programming with the Message-Passing Interface, 3rd Ed, MIT Press, 2014.

Mapping of COs with PSOs and POs :

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PO1	PO2	PO3	PO4	PO5	PO6
CO1	-	2	-	-	-	-						
CO2	-	2	1	-	-	-						
CO3	-	2	1	-	1	1						
CO4	-	2	1	1	1	2						
CO5	-	3	1	1	-	2						
CO6	-	2	-	-	-	2						

Correlation Levels:

Level	Correlation
-	Nil
1	Slightly / Low
2	Moderate / Medium
3	Substantial / High

Assessment Rubrics:

- Quiz / Assignment/ Quiz/ Discussion / Seminar
- Midterm Exam
- Assignments (20%)
- Final Exam (70%)

Mapping of COs to Assessment Rubrics :

	Internal Exam	Assignment	Project Evaluation	End Semester Examinations
CO 1	✓			✓

CO 2	✓			✓
CO 3	✓	✓		✓
CO 4		✓		✓
CO 5		✓		✓
CO 6			✓	

Programme	B.Sc. Computer Science				
Course Code	CSC8CJ489				
Course Title	**Research Methodology**				
Type of Course	Major				
Semester	VIII				
Academic Level	400 - 499				
Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	4	4	-	-	60
Pre-requisites	Knowledge of Planning a research project, problem formulation, framing objectives				
Course Summary	This course introduces and discusses approaches, strategies, and data collection methods relating to research. Students will consider how to select the appropriate methodology for use in a study to be performed. Additionally, these students will learn how to collect data based on different data collection methods, construct these tools, and pilot them before they become ready for use. To culminate this final stage, students will learn to write a comprehensive research proposal that may be conducted in the future				

Course Outcomes (CO):

CO	CO Statement	Cognitive Level*	Knowledge Category#	Evaluation Toolsused
CO1	Understand the psychology of research which includes different perspectives and necessity of research.	U	C	Instructor-createdexams / Quiz
CO2	Apply the research knowledge to formulate a suitable problem statement by adopting different research methods and models	U	C	Instructor-createdexams / Quiz
CO3	Understand different methods of Collection, Validation and Testing of Data	U	P	Instructor-createdexams / Quiz
CO4	To understand the data processing and analysis techniques	U	C	Instructor-created exams / Case studies
CO5	Analyze the research outcome by using suitable statistical tool.	U	P	Instructor-created exams / Quiz Case studies
CO6	To write or present a scientific report and research proposal	U	P	Instructor-createdexams / Quiz /Case studies

* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C)# - Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge (M)

Detailed Syllabus:

Module	Unit	Content	Hrs	Marks
I		**Introduction to Research Methodology**	**12**	**15**
	1	Research Methodology: An Introduction to the Meaning of Research and Objectives of Research	2	
	2	Motivation in Research ,Types of Research	2	
	3	Research Approaches	2	
	4	Significance of Research	3	
	5	Research Methods versus Methodology .	3	
II		**Identifying, Defining and Designing Research Problem**	**12**	**15**
	6	Defining the Research Problem What is a Research Problem? Selecting the Problem, Necessity of Defining the Problem	2	
	7	Technique Involved in Defining a Problem	1	
	8	Research Design: Meaning of Research Design, Need for Research Design	3	
	9	Research Methodology,Features of a Good Design	3	
	10	Important Concepts Relating to Research Design	1	
	11	Different Research Designs	2	
III		**Collection, Validation and Testing of Data**	**12**	**20**
	12	Sources of Data: Primary and Secondary, Validation of Data , Data Collection Methods: Questionnaire Designing	2	
	13	Construction Sampling Design & Techniques – Probability Sampling and Non Probability Sampling Scaling Techniques:	2	
	14	Meaning & Types Reliability: Test – Retest Reliability,	2	
	15	Alternative Form Reliability	2	
	16	Internal Comparison Reliability and Scorer Reliability	2	
	17	Validity: Content Validity, Criterion Related Validity and Construct Validity	2	
IV		**Data Processing and Analysis**	**12**	**20**
	18	Processing and Analysis of Data, Processing Operations, Some Problems in Processing, Elements/Types of Analysis	2	
	19	Statistics in Research Measures of Central Tendency	2	
	20	Measures of Dispersion Interpretation and Report Writing	3	
	21	Meaning of Interpretation Why Interpretation? Technique of Interpretation:Precaution in Interpretation	2	
	22	Significance of Report Writing Different Steps in Writing Report Layout of the Research Report	3	
V		**Open Ended Module**	**12**	
		CASE STUDY: RESEARCH METHODOLOGY Methods of Research Applications of Statistical tools & Methods Structure and components of scientific reports		

Reference Books:

1. C.R .Kothari, 'Research Methodology Methods & Techniques', Revised 2 nd Edn., New Age International Publishers.Research Methodology and Scientific Writing by C George Thomas, Ane Books Pvt. Ltd.

2. An Introduction to Research Methodology; Garg B.L., Karadia, R., Agarwal, F. and Agarwal, U.K., 2002., RBSA Publishers.

3. Research Methodology ; Panneerselvam R., PHI, Learning Pvt. Ltd., New Delhi - 2009

4. Research Methodology: Concepts and cases, Chawala D. and N. Sondhi ; Vikas Publishing House Pvt. Ltd.

Mapping of COs with PSOs and POs :

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PO1	PO2	PO3	PO4	PO5	PO6
CO1	-	-	-	-	-	-						
CO2	-	-	1	-	-	-						
CO3	2	2	1	-	1	1						
CO4	2	2	2	2	1	2						
CO5	2	1	2	2	1	2						
CO6	-	-	-	-	-	1						

Correlation Levels:

Level	Correlation
-	Nil
1	Slightly / Low
2	Moderate / Medium
3	Substantial / High

Assessment Rubrics:

- Quiz / Assignment/ Quiz/ Discussion / Seminar
- Midterm Exam
- Assignments (20%)
- Final Exam (70%)

Mapping of COs to Assessment Rubrics :

	Internal Exam	Assignment	Project Evaluation	End Semester Examinations
CO 1	✓			✓
CO 2	✓			✓
CO 3	✓	✓		✓
CO 4		✓		✓
CO 5		✓		✓
CO 6			✓	

ELECTIVES

Basket of Electives

Data Science										
No	**Course Code**	**Course Name**	**C**	**Marks**			**Hrs/wk**		
				I	**E**	**T**	**L**	**P**	**T**
29	CSC5EJ305a	Mathematical and Statistical Foundation for Data Science	4	30	70	100	4	0	4
30	CSC5EJ306a	Exploratory Data Analysis	4	30	70	100	4	0	4
35	CSC6EJ311a	Introduction to Data Warehousing and Big Data	4	30	70	100	4	0	4
36	CSC6EJ312a	Advanced Python for Data Science	4	30	70	100	4	0	4

Programme	B. Sc. Computer Science				
Course Code	**CSC5EJ305a**				
Course Title	**Mathematical and Statistical Foundations for Data Science**				
Type of Course	Elective				
Semester	V				
Academic Level	300-399				
Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	4	4	-	-	60
Pre-requisites	1. Basic Mathematics and Statistics 2. Python basics (If Python implementation is preferred in module V by the course tutor)				
Course Summary	This undergraduate course provides the fundamental mathematical and statistical tools necessary for understanding and analyzing data in the context of data science. The course covers topics ranging from basic algebraic operations to advanced statistical techniques.				

Course Outcomes (CO):

CO	CO Statement	Cognitive Level*	Knowledge Category#	Evaluation Tools used
CO1	Apply Vector and Matrix operations to solve computational problems.	Ap	P	Instructor-created exams / Assignment

CO2	Students will evaluate eigenvalues and eigenvectors to decompose matrices, enabling them to analyze and interpret data transformations effectively.	An	P	Instructor-created exams / Assignment
CO3	Students will apply fundamental probability concepts to solve real-world problems.	Ap	P	Assignment / Quiz
CO4	Students will utilize statistical techniques for data interpretation and decision-making	Ap	P	Instructor-created exams / Assignment
CO5	Students will apply sampling techniques and hypothesis tests to make inferences about populations from sample data, using one-tailed, two-tailed tests, and ANOVA for analysis	Ap	Q	Assignment / Case Studies
CO6	Students will apply PCA to reduce data dimensionality, identify principal components, and interpret results in data science application.	Ap	R	Assignment / Case Studies

* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C)
# - Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge (M)

Detailed Syllabus:

Module	**Unit**	**Content**	**Hrs (48+12)**	**Marks 70**
I	**Linear Algebra**		**14**	**20**

	1	Scalars, Vectors, Matrices and Tensors Vectors:- Vector Arithmetic - Vector Addition, Vector Subtraction, Vector Multiplication, Vector Division; Vector Dot Product; Vector Scalar Multiplication	2	
	2	Matrix Multiplication, Identity and Inverse Matrices, Linear dependence and Span , Norms, Diagonal and Orthogonal Matrices	4	
	3	Eigenvectors and Eigenvalues , Eigen Decomposition	2	
	4	Singular Value Decomposition	2	
	5	The Trace Operator , The Determinant	2	
	6	Principal Component Analysis	2	
II	**Probability**		**14**	**20**
	7	Random Variables , Probability Distributions	3	
	8	Marginal Probability , Conditional Probability,	1	
	9	The Chain Rule of Conditional Probabilities	3	
	10	Independence and Conditional Independence	2	
	11	Expectation, Variance and Covariance	1	
	12	Common Probability Distributions - Bernoulli Distribution	3	
		Binomial, Normal and Poisson Distribution		
	13	Bayes' Rule	1	
III	**Basic Statistics**		**8**	**15**
	14	Measures of Central Tendency	3	
	15	Measures of Dispersion	2	

	16	Skewness, Kurtosis	1	
	17	Correlation and Regression	2	
IV	**Sampling and Hypothesis Testing**		**12**	**15**
	18	Sampling distributions of the sample mean and the sample variance for a normal population	2	
	19	Point and interval estimation	1	
	20	Sampling distributions (Chi-square, t, F, Z)	3	
	21	Hypothesis testing	1	
	22	One tailed and two-tailed tests, Analysis of variance, ANOVA, One way and two way classifications.	5	
V	**Application oriented module**		**12**	
	Solve the following problems mathematically. Or Try to implement these problems using Python.			
	1	**Linear Algebra** Concepts to be learned:- • Vector arithmetic (a) Define a vector *a* and *b* with the length of 3 and the integer values 1, 2 and 3. (b) Perform addition, subtraction, multiplication, division and dot product of the two vectors *a* and	1	

<table>
<tr><td></td><td>2</td><td>Linear Algebra
Concepts to be learned: -
• Matrix arithmetic
(a) Create two 2 row, 3 column matrices, say A and B. Perform matrix addition, subtraction, division and multiplication (element-wise matrix multiplication or the Hadamard product).
(b) Create a matrix A with 3 rows and 2 columns, and a matrix B with 2 rows and 2 columns. Perform matrix dot product of matrices A and B.</td><td>1</td><td></td></tr>
<tr><td></td><td>3</td><td>Linear Algebra
Concepts to be learned: -
• Singular Value Decomposition
• Orthogonal Matrices
• Diagonal Matrix
• Singular Value
• Eigen values and Eigen Vectors
• Matrix Multiplication
Find the singular value decomposition of the matrix
[2 2
-1 1]</td><td>2</td><td></td></tr>
<tr><td></td><td>4</td><td>Basic StatisticsConcepts to be learned: -
Measures of Central Tendency - Mean, Median, ModeFind the mean, the median, and the mode for the number of vehicles owned in a s
survey of 52 households.
<table><tr><td>x</td><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td></tr><tr><td>f</td><td>2</td><td>12</td><td>15</td><td>11</td><td>6</td><td>3</td><td>1</td><td>2</td></tr></table></td><td>1</td><td></td></tr>
</table>

<table>
<tr><td></td><td>5</td><td>Basic Statistics
Concepts to be learned: -
• Measures of Dispersion - Range, Variance, Standard Deviation
Find the range, the variance and the standard deviation for the sample of ten IQ scores randomly selected from a school for academically gifted students.
142 152 138 145 148 139 147 155 150 153</td><td>1</td><td></td></tr>
<tr><td></td><td>6</td><td>Application of Probability
Concepts to be learned: -
• Probability basics
• Combinations
• Mutually exclusive events
• Complementary events
Of 10 girls in a class, 3 have blue eyes. If two of the girls are chosen at random, what is the probability that (i) both have blue eyes, (ii) neither has blue eyes, (iii) at least one has blue eyes?</td><td>1</td><td></td></tr>
<tr><td></td><td>7</td><td>Application of Probability
Concepts to be learned: -
• Probability Basics
• Contingency Tables
• Marginal and Joint Probabilities
• Conditional Probability
The following two-way contingency table gives the breakdown of the population in a particular locale according to age and tobacco usage.
<table>
<tr><td rowspan="2">Age</td><td colspan="2">Tobacco Use</td></tr>
<tr><td>Smoker</td><td>Non-smoker</td></tr>
<tr><td>Under 30</td><td>0.05</td><td>0.20</td></tr>
<tr><td>Over 30</td><td>0.20</td><td>0.55</td></tr>
</table>
</td><td>1</td><td></td></tr>
</table>

		A person is selected at random. Find the probability of each of the following events. (a) The person is a smoker. (b) The person is under 30. (c) The person is a smoker who is under 30.		
	8	**Application of Probability** Concepts to be learned: - • Understand the characteristics of a normal distribution. • Calculating and interpreting z-scores. Suppose the heights H of 800 students are normally distributed with mean 66 inches and standard deviation 5 inches. Find the number N of students with heights (a) between 65 and 70 inches, (b) greater than or equal to 6 feet(72inches).	1	
	9	**Application of Probability** Concepts to be learned: - • Bayes' Theorem A patient goes to see a doctor. The doctor performs a test with 99 percent reliability--that is, 99 percent of people who are sick test positive and 99 percent of the healthy people test negative. The doctor knows that only 1 percent of the people in the country are sick. If the patient tests positive, what are the chances the patient is sick?	1	
	10	**Sampling and Hypothesis Testing** Concepts to be learned: - • Hypothesis testing Contingency tables, and • Chi-square analysis	1	

		A die is suspected of being biased. It is rolled 25 times with the following result: Outcome Frequency 1 9 2 4 3 1 4 8 5 3 6 0 Conduct a significance test to see if the die is biased. (a) What Chi Square value do you get and how many degrees of freedom does it have? (b) What is the p value?		
	11	**Sampling and Hypothesis Testing** Concepts to be learned: - • Central Limit Theorem • Sampling distribution of the Sample Mean • Standard Error of the Mean • Z-score • Normal Distribution Properties • Probability Calculation Suppose scores on an IQ test are normally distributed, with a mean of 100. Suppose 20 people are randomly selected and tested. The standard deviation in the sample group is 15. What is the probability that the average test score in the sample group will be at most 110?	1	

Mapping of COs with PSOs and POs:

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6
CO 1	3	-	2	2	2	2
CO 2	3	-	2	3	2	2
CO 3	3	-	3	3	2	2
CO 4	3	-	3	3	2	2
CO 5	3	-	3	3	2	2
CO 6	3	-	3	3	2	2

Correlation Levels:

Level	**Correlation**
-	Nil
1	Slightly / Low
2	Moderate / Medium
3	Substantial / High

Mapping of COs to Assessment Rubrics:

	Internal Exam	Assignment	Project Evaluation	End Semester Examinations
CO 1	✓	✓		✓
CO 2	✓	✓		✓
CO 3	✓			✓
CO 4	✓	✓		✓
CO 5	✓	✓	✓	✓
CO 6	c✓	✓	✓	

References

1. Ian Goodfellow, Yoshua Bengio, and Aaron Courville. Deep Learning. MIT Press, 2017.
2. Gilbert Strang. Introduction to Linear Algebra. 5th ed. Wellesley-Cambridge Press, 2016.
3. S. Ross, Introduction to Probability and Statistics for and Engineers and Scientists, Third Edition, Elsevier, 2004.

Programme	B. Sc. Computer Science				
Course Code	**CSC5EJ306a**				
Course Title	**Exploratory Data Analysis**				
Type of Course	**Elective**				
Semester	V				
Academic Level	300-399				
Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	4	4	-	-	60
Pre-requisites	1. Basic Statistical Knowledge 2. Python Programming including knowledge in Pandas library				
Course Summary	This course explores the different visualization tools and techniques and teaches the application of these techniques using Python packages.				

Course Outcomes (CO):

CO	CO Statement	Cognitive Level*	Knowledge Category#	Evaluation Tools used
CO1	Understand the importance of data visualization for business intelligence and decision making.	U	C	Instructor-created exams / Quiz
CO2	Understand different types of charts and plots such as line, area, histograms, bar, pie, box, scatter, and bubble.	U	C	Instructor-created exams / Quiz
CO3	Learn about categories of visualization and application areas.	R	C	Instructor-created exams / Quiz
CO4	Familiarize with the data visualization tools and techniques.	Ap	P	Assignments/ Case Studies
CO5	Familiarise with the Python libraries, such as Matplotlib, Seaborn, Folium, Bokeh and learn how to to tell a stimulating story.	Ap	P	Assignments/ Case Studies
CO6	Create advanced visualizations for geo spatial data.	Ap	P	Assignments/ Case Studies

* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C) # - Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge (M)

Detailed Syllabus:

Module	Unit	Content	Hrs 48+12	Marks 70
I	**Introduction to Data Visualization**		**15**	**20**
	1	Data:- Types of Data-Structured and Unstructured Data, Qualitative and Quantitative Data, Continuous and Discrete Data, Primary and Secondary Data, Data Attributes - Types of Data Attributes - Nominal, Ordinal, Interval, Ratio	3	
	2	Introduction to Data Visualization:- Data Visualization, The Importance of Data Visualization, Overview of popular data visualization libraries in Python - Matplotlib, Seaborn, Folium, Bokeh	1	
	3	Plots:- Comparison Plots: Line Chart, Bar Chart and Radar Chart	2	
	4	Relation Plots: Scatter Plot, Bubble Plot, Correlogram and Heatmap	2	
	5	Composition Plots: Pie Chart, Stacked Bar Chart, Stacked Area Chart, Venn Diagram	3	
	6	Distribution Plots: Histogram, Density Plot, Box Plot, Violin Plot	2	
	7	Geo Plots: Dot Map, Choropleth Map, Connection Map	2	
II	**Data Visualization with Matplotlib**		**10**	**20**
	8	Introduction, Overview of Plots in Matplotlib Pyplot Basics: Creating Figures, Closing Figures, Format Strings, Plotting, Plotting Using pandas DataFrames, Displaying Figures, Saving Figures	3	
	9	Basic Text and Legend Functions: Labels, Titles, Text, Annotations, Legends	1	

	10	Basic Plots: Bar Chart, Pie Chart, Stacked Bar Chart, Stacked Area Chart, Histogram, Box Plot, Scatter Plot, Bubble Plot	3	
	11	Layouts: Subplots, Tight Layout, Radar Charts, GridSpec	2	
	12	Images: Basic Image Operations, Writing Mathematical Expressions	1	
III	**Simplifying Visualizations using Seaborn**		**12**	**15**
	13	Introduction, Advantages of Seaborn, Plot a Relation Plot, Line Plot, Box Plot and, a Heat Map	2	
	14	Controlling Figure Aesthetics: Seaborn Figure Styles, Removing Axes Spines, Contexts	2	
	15	Color Palettes: Categorical Color Palettes, Sequential Color Palettes, Diverging Color Palettes	4	
	16	Interesting Plots in Seaborn: Bar Plots, Kernel Density Estimation, Plotting Bivariate Distributions, Visualizing Pairwise Relationships, Violin Plots	4	
IV	**Plotting Geospatial Data**		**11**	**15**
	17	Introduction to Geoplotlib, The Design Principles of Geoplotlib	1	
	18	Geospatial Visualizations - Choropleth Plot, GeoJSON File	2	
	19	Introduction to Folium	1	
	20	Visualizing Data: Building a Google map from geocoded data	2	
	21	Making Things Interactive with Bokeh : Introduction to Bokeh, Concepts of Bokeh, Interfaces in Bokeh, Output	3	
	22	Bokeh Server, Presentation, Integrating, Adding Widgets	2	
V	**Hands-on Data Visualization:** Practical Applications - Implement any 10 programs		**12**	

		Comparison Plots: Line Chart, Bar Chart, and Radar Chart 1. Write a Python script to create a line chart comparing the sales performance of two products over different months using Matplotlib. 2. Create a bar chart using Seaborn to visualize the average scores of students in different subjects. 3. Implement a radar chart using Matplotlib to compare the performance of multiple candidates in different skills.	2	
		Relation Plots: Scatter Plot, Bubble Plot, Correlogram, and Heatmap 4. Generate a scatter plot using Seaborn to analyze the relationship between the height and weight of individuals in a dataset. 5. Create a line graph with bokeh using Annotations and Legends. 6. Plot a correlogram heatmap using Seaborn to visualize the correlation matrix of variables in a dataset.	2	
		Composition Plots: Pie Chart, Stacked Bar Chart, Stacked Area Chart, Venn Diagram 7. Implement a pie chart using Matplotlib to represent the distribution of expenses in a budget. 8. Create a stacked bar chart using Seaborn to visualize the sales performance of different product categories over multiple quarters. 9. Generate a stacked area chart using Matplotlib to display the cumulative distribution of COVID-19 cases over time in different regions. 10. Use the matplotlib-venn library to create a Venn diagram illustrating the intersection of sets in a survey dataset.	2	

		Distribution Plots: Histogram, Density Plot, Box Plot, Violin Plot 11. Write a Python function to generate a histogram using Matplotlib for analyzing the distribution of exam scores in a class. 12. Create a density plot using Seaborn to visualize the distribution of income levels in a population. 13. Implement a box plot using Matplotlib to compare the distribution of salaries across different job roles. 14. Generate a violin plot using Seaborn to compare the distribution of ages between male and female participants in a study.	3	
		Geo Plots: Dot Map, Choropleth Map, Connection Map 15. Use Folium to create a dot map representing the locations of earthquake occurrences around the world. 16. Generate a choropleth map using Folium to visualize the population density of different countries. 17. Create a connection map using Matplotlib to illustrate flight routes between various cities.	3	

Mapping of COs with PSOs and POs :

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6
CO 1	-	-	2	2	2	2
CO 2	-	-	2	2	2	2
CO 3	-	-	-	2	2	2
CO 4	-	-	2	2	2	2
CO 5	-	-	2	2	2	2
CO 6	-	-	2	2	2	2

Correlation Levels:

Level	**Correlation**
-	Nil
1	Slightly / Low
2	Moderate / Medium
3	Substantial / High

Assessment Rubrics:

- Quiz / Assignment/ Quiz/ Discussion / Seminar
- Midterm Exam
- Programming Assignments (20%)
- Final Exam (70%

Mapping of COs to Assessment Rubrics :

	Internal Exam	Assignment	Project Evaluation	End Semester Examinations
CO 1	✓	✓		✓
CO 2	✓		✓	✓
CO 3	✓			✓
CO 4	✓	✓		✓
CO 5	✓	✓	✓	✓
CO 6	✓	✓		✓

References

1. Data Visualization workshop, Tim Grobmann and Mario Dobler, Packt Publishing
2. Kristen Sosulski, “Data Visualization Made Simple”, Taylor & Francis, 2019.
3. Pooja, Dr. Data Visualization with Python: Exploring Matplotlib, Seaborn, and Bokeh for Interactive Visualizations. BPB Online, 2023.
4. Wilke, Claus O. Fundamentals of data visualization: a primer on making informative and compelling figures. O'Reilly Media, 2019.
5. VanderPlas, Jake. Python data science handbook: Essential tools for working with data. " O'Reilly Media, Inc.", 2016.

Online Learning Resources

1. https://www.coursera.org/courses?query=data%20visualization
2. https://www.simplilearn.com/free-data-visualization-course-online-skillup

Programme	B.Sc. Computer Science
Course Code	**CSC6EJ311a**
Course Title	**Introduction to Data Warehousing and Big Data**
Type of Course	**Elective**
Semester	VI
Academic Level	300 - 399

Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	4	4	-	0	75

Pre-requisites	1. Data Science Concepts 2. RDBMS
Course Summary	This course provides insight into the basic concepts of data warehousing and its architecture. The various OLAP operations are also discussed in this syllabus to understand the summarisation and retrieval of the data. The fundamentals of big data technology are also introduced in this syllabus following the data warehousing concepts. An overview of the storage, retrieval and processing of big data is also provided here.

Course Outcomes (CO):

CO	CO Statement	Cognitive Level*	Knowledge Category#	Evaluation Tools used
CO1	Understand the concepts of data warehouse and its architecture	U	C	Instructor-created exams / Quiz
CO2	Analyse the differences between OLTP and OLAP operations	An	C	Instructor-created exams / Quiz
CO3	Understand the various operations performed in the data warehouse to process the data	U	C	Modelling Assignments/ Case Studies
CO4	Understand Big Data and the importance of cloud and distributed computing in the real world	U	C	Instructor-created exams / Quiz
CO5	Understand the Map Reduce concepts of the jobs	U	C	Modelling Assignments/ / Case studies
CO5	Understand the Hadoop ecosystem	U	C	Instructor-created exams / Quiz

* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C)

\# - Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge (M)

Detailed Syllabus:

Module	Unit	Content	Hrs	Marks
I	**Introduction to Data Warehousing**		**8**	**10**
	1	Overview of databases and need for normalisation. Databases vs data warehouse	2	
	2	Introduction to Data warehousing, Need for data warehousing	2	
	3	Architecture of data warehousing	3	
	4	Data Marts vs Data Lakes	1	
II	**Concepts and techniques in Data Warehousing**		**13**	**20**
	6	Data warehouse Schema - Stars, snowflakes and fact constellations	3	
	7	OLAP (Online analytical processing) definitions// Difference between OLAP and OLTP	2	
	8	Dimensional analysis - What are cubes?	2	
	9	Drill-down and roll-up - slice and dice or rotation	2	
	10	OLAP models, ROLAP versus MOLAP	4	
III	**Big Data Technology**		**16**	**25**
	11	Fundamentals of Big Data, 3V's of big data. Structured Data and its sources; Unstructured data and its sources; integrating data types to big data	2	
	12	Big Data Stack: Layers 1 to 4; Big data analytics and applications	4	
	13	Role of Distributed computing and virtualizations in big data	3	
	14	Hypervisor and implementing virtualizations in big data	1	
	15	Cloud in big data; cloud deployment models	2	
	16	Cloud delivery models; advantages of using cloud	2	
	17	Cloud Providers for Big Data	2	
IV	**Big Data Management**		**11**	**15**
	18	Fundamentals of Map Reduce: Map and reduce functions	2	
	19	Putting Map and Reduce together	2	
	20	Hadoop: Name nodes, Data Nodes, Hadoop MapReduce	3	
	21	Hadoop ecosystem: Yarn, HBase and Hive Interactive tools: Pig, Pig Latin, SQOOP, ZooKeeper	4	
	22	Big Data Analytics: Basic, Advanced, Operationalized	1	
V	**Hands-on data and Data warehousing: Practical Applications, Case Study and Course Project**		**12**	
	1	Data warehousing case studies	4	
	2	Case studies on Big Data Analytics and Big Data Solutions in the Real World	5	
	3	Assignments on Security in Big Data Environments	3	

Mapping of COs with PSOs and POs :

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PO1	PO2	PO3	PO4	PO5	PO6
CO 1	3	-	-	-	-	1						
CO 2	1	-	2	-	-	-						
CO 3	-	-	2	-	-	-						
CO 4	-	2	3	3	-	1						
CO 5	-	2	3	3	-	1						
CO 6	-	-	-	-	-	2						

Correlation Levels:

Level	**Correlation**
-	Nil
1	Slightly / Low
2	Moderate / Medium
3	Substantial / High

Assessment Rubrics:

- Quiz / Assignment/ Quiz/ Discussion / Seminar
- Midterm Exam
- Programming Assignments (20%)
- Final Exam (70%)

Mapping of COs to Assessment Rubrics :

	Internal Exam	Assignment	Project Evaluation	End Semester Examinations
CO 1		✓		✓
CO 2	✓	✓		✓
CO 3		✓		✓
CO 4	✓			✓
CO 5	✓		✓	✓
CO 6	✓		✓	✓

References

1. O'Neil, Cathy, and Rachel Schutt. *Doing data science: Straight talk from the frontline*. " O'Reilly Media, Inc.", 2013.
2. Han, Jiawei, et al. Data Mining: Concepts and Techniques. Netherlands, Elsevier Science, 2011.
3. Shah, Chirag. A Hands-On Introduction to Data Science. United Kingdom, Cambridge University Press, 2020.
4. Chopra, Rohan, et al. Data Science with Python: Combine Python with Machine Learning Principles to Discover Hidden Patterns in Raw Data. United Kingdom, Packt Publishing, 2019.

Programme	B. Sc. Computer Science				
Course Code	**CSC6EJ312a**				
Course Title	**Advanced Python for Data Science**				
Type of Course	**Elective**				
Semester	VI				
Academic Level	300 ▫ 399				
Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	4	4	-	-	60
Pre-requisites	1. Data Science Concepts 2. Python basics				
Course Summary	This course provides insight into the basic concepts of Python required for Data Science. It includes array fundamentals, array transformations, and matrices fundamentals. The analysis of data using Pandas will help the students to understand the basics of data analysis				

Course Outcomes (CO):

CO	CO Statement	Cognitive Level*	Knowledge Category#	Evaluation Tools used
CO1	Understand the concepts of arrays, matrices and their transformations	U	C	Instructor-created exams / Quiz
CO2	Create informative plots using Python packages	Ap	P	Modelling Assignments/ Case Studies
CO3	Understand the loading mechanism of different types of data and manipulate them	U	C	Instructor-created exams / Quiz
CO4	Analyse the data using Pandas and Data Frames	An	P	Modelling Assignments/ Case Studies
CO5	Understand the concepts of random tensors and generate tensors from various distributions	U	C	Instructor-created exams / Quiz
CO6	Familiarize with various TensorFlow operations needed for Data Science	U	C	Instructor-created exams / Quiz

* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C)

- Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge (M)

Detailed Syllabus:

Module	Unit	Content	Hrs	Marks
I	**Arrays, Matrix manipulation using NumPy**		**10**	**12**
	1	Array creation, sorting, concatenating	2	
	2	Shape and size of an array, basic arithmetic operations on an array, broadcasting	2	
	3	Aggregate functions on arrays, Unique and count operations	2	
	4	Matrices using NumPy	2	
	5	Transpose, reverse, flatten and ravel	2	
II	**Data Analysis and Manipulation using Pandas**		**12**	**18**
	6	Series - constructing from an array, using explicitly defined indices, using a dictionary.	2	
	7	Data Frame - constructing from arrays, dictionaries, structured arrays, and series, Indexing of data frames	3	
	8	Arithmetic and Binary operations on Data frame	3	
	9	Broadcasting operations	2	
	10	Universal functions, melt() and pivot()	2	
III	**Other Python packages for data science**		**10**	**14**
	11	Scipy, Scikit-learn, PyTorch, Seaborn, Scrapy, and Beautiful Soup.	3	
	12	Python Data Operations: Importing and Exporting Data, Data Cleansing	3	
	13	Processing CSV Data, Processing JSON Data, Processing XLS Data.	2	
	14	Data Analysis: Measuring Central Tendency, Measuring Variance, and Correlation in Python	2	
IV	**TensorFlow Fundamentals**		**16**	**26**
	15	Tensors, creation of tensors and random tensors, Tensors from the Normal distribution, Poisson distribution, set_seed()	2	
	16	Tensor attributes, size, rank and reshaping of a tensor	2	
	17	Tensor arithmetic, relational, logical operations. Shuffle()	2	
	18	Reduce operations on tensor Dimension-wise	2	
	19	Ragged tensors, TensorArray, dynamic arrays,	2	

	20	unique(), fill(), concat(), gather(), ones(), ones_like(), zeros(),	2	
	21	eye(), range(), repeat, reverse(), roll(), slice(), sort(),	2	
	22	split(),squeeze(), tile(), stack(), unstack(), tensordot()	2	
V	**Hands-on Data Structures:** **Practical Applications, Case Study and Course Project**		**12**	
	1	Use Pandas and NumPy to efficiently process and analyze CSV, Excel, or JSON data	4	
	2	Create compelling visual insights using Matplotlib, Seaborn, or Plotly	3	
	3	Case studies with Tensor flow	5	

Mapping of COs with PSOs and POs :

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PO1	PO2	PO3	PO4	PO5	PO6
CO 1	1	-	-	2	2	2						
CO 2	-	1	-	-	2	2						
CO 3	-	-	2	-	2	2						
CO 4	-	1	1	2	2	2						
CO 5	1	-	-	-	2	2						
CO 6	-	-	2	2	2	2						

Correlation Levels:

Level	Correlation
-	Nil
1	Slightly / Low
2	Moderate / Medium
3	Substantial / High

Assessment Rubrics:

- Quiz / Assignment/ Quiz/ Discussion / Seminar
- Midterm Exam
- Programming Assignments (20%)
- Final Exam (70%)

Mapping of COs to Assessment Rubrics :

	Internal Exam	Assignment	Project Evaluation	End Semester Examinations
CO 1	√			√
CO 2	√	√	√	√
CO 3	√		√	√
CO 4	√	√	√	√
CO 5	√			√
CO 6	√			√

References

1. VanderPlas, Jake. Python Data Science Handbook: Essential Tools for Working with Data. United States, O'Reilly Media, 2016.
2. Rogel-Salazar, Jesus. Data Science and Analytics with Python. United Kingdom, CRC Press, 2018.
3. https://numpy.org/doc/
4. https://pandas.pydata.org/docs/
5. https://www.tensorflow.org/guide

AI and ML										
No	**Course Code**	**Course Name**	**C**	**Marks**			**Hrs/wk**		
				I	**E**	**T**	**L**	**P**	**T**
29	CSC5EJ305b	Machine Learning Algorithms	4	30	70	100	4	0	4
30	CSC5EJ306b	Knowledge Engineering	4	30	70	100	4	0	4
35	CSC6EJ311b	Soft Computing	4	30	70	100	4	0	4
36	CSC6EJ312b	Deep Learning	4	30	70	100	4	0	4

Programme	B. Sc. Computer Science				
Course Code	**CSC5EJ305b**				
Course Title	**Machine Learning Algorithms**				
Type of Course	**Elective**				
Semester	V				
Academic Level	300 - 399				
Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	4	4	-	-	60
Pre-requisites	1. Understanding of basic mathematics and statistics (linear algebra, calculus, probability)				
Course Summary	This course introduces the fundamental concepts, algorithms, and applications of machine learning				

Course Outcomes (CO):

CO	CO Statement	Cognitive Level*	Knowledge Category#	Evaluation Tools used
CO1	Understand basic concepts of machine learning, including supervised learning, unsupervised learning, and reinforcement learning	U	C	Instructor-created exams / Quiz
CO2	Understand the mathematical foundations of machine learning algorithms, including concepts such as optimization, linear algebra, probability, and statistics	U	C	Assignment / Seminar presentations/ Exams

CO3	Demonstrate proficiency in various machine learning algorithms, such as linear regression, logistic regression, decision trees, support vector machines, k-nearest neighbors, clustering algorithms, and neural network	U	P	Seminar Presentation / Group Tutorial Work/ Viva Voce
CO4	Explore techniques for feature engineering and feature selection to improve the performance of machine learning models.	U	P	Instructor-created exams / Home Assignments
CO5	Evaluate machine learning models using appropriate metrics and techniques, including cross-validation, precision, recall, F1 score, ROC curves, and confusion matrices.	Ap	P	Writing assignments/ Exams/ Seminar Presentations
CO6	Develop critical thinking skills to analyze and solve complex problems using machine learning approaches.	Ap	P	Case Study/ Group discussions/ Presentations
* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C) # - Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge (M)				

Detailed Syllabus:

Module	Unit	Content	Hrs (48+12)	Marks (70)
I		**Mathematical Foundation for Machine learning**	**14**	**20**
	1	Introduction to key concepts: features, labels, training, and testing	2	
	2	Designing a Learning system	1	
	3	Types of learning; supervised, unsupervised and reinforcement	2	
	4	Introduction to linear algebra- Vector :-Vector operations: addition, subtraction, scalar multiplication	2	
	5	Matrices- Matrix operations	2	
	6	Eigenvalues and Eigenvectors	2	
	7	Foundations of Probability for ML:- Introduction to probabilty	1	
	8	Random Variable, Probability distributions (Normal and gaussian-basics only), Naïve bayes	2	
II		**Feature Engineering and Preprocessing**	**12**	**15**
	9	Data Preprocessing and Feature Engineering: Data Representation, Data Preprocessing	2	
	10	Features and Types	3	
	11	Dimensionality Reduction – Feature Identification	2	
	12	Feature selection	2	
	13	Feature extraction - Feature Importance	3	
III		**Regression and Classification**	**12**	**20**
	14	Regression: Linear Regression – Non-Linear regression	2	
	15	Evaluation metrics for regression	1	

	16	Classification: Binary, multi-class, and multi-label classification	1	
	17	lazy leaners- (KNN) - tree-based techniques (Decision Tree)- kernel based techniques (SVM) - probabilistic techniques (Naïve bayes)- and ensembled techniques (bagging, boosting, voting)	7	
	18	Evaluation metrics for classification.	1	
IV		**Clustering and Rule Mining**	**10**	**15**
	19	Clustering: Partitioning based (K Means)	2	
	20	Hierarchical based (Divisive)	2	
	21	Rule mining: Apriori algorithm, FB Growth - association rules.	4	
	22	Outlier Detection - LOF	2	
V		**Open Ended Module**	**12**	
	1	Ethical considerations in machine learning	3	
	2	McCulloch-Pitts neurons, Hebb's networks	3	
	3	Hopfield networks, Boltzmann machines	2	
	4	Reinforcement Learning: Markov Decision Processes (MDPs), Q-learning.	4	

References

- Ethem Alpaydin, Introduction to Machine Learning- 3rd Edition, PHI
- Machine Learning by Mitchell, Tom M. (Tom Michael), McGraw-Hill
- Mathematics For Machine Learning, Marc Peter Deisenroth A. Aldo Faisal Cheng Soon Ong
- "Pattern Recognition and Machine Learning" by Christopher M. Bishop.

Mapping of COs with PSOs and POs :

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PO1	PO2	PO3	PO4	PO5	PO6
CO 1	2	1	-	-	1	-						
CO 2	3	-	-	-	1	-						
CO 3	1	3	1	1	2	3						
CO 4	1	-	1	1	2	3						
CO 5	1	-	-	-	2	3						
CO 6	1	2	2	2	3	3						

Correlation Levels:

Level	Correlation
-	Nil
1	Slightly / Low
2	Moderate / Medium

3	Substantial / High

Assessment Rubrics:

- Quiz / Assignment/ Quiz/ Discussion / Seminar
- Midterm Exam
- Programming Assignments (20%)
- Final Exam (70%)

Mapping of COs to Assessment Rubrics :

	Internal Exam	Assignment	End Semester Examinations
CO 1	✓		✓
CO 2	✓		✓
CO 3	✓		✓
CO 4	✓	✓	✓
CO 5	✓	✓	✓
CO 6		✓	

Programme	B. Sc. Computer Science
Course Code	**CSC5EJ306b**
Course Title	**Knowledge Engineering**
Type of Course	**Elective**
Semester	V
Academic Level	300 - 399

Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	4	4	-	-	60

Pre-requisites	2. Understanding of basic mathematics and statistics 3. Basic understanding of computer science concepts
Course Summary	This course introduces students to the principles, techniques, and tools used in Knowledge Engineering. It covers the design and development of knowledge-based systems, including knowledge representation, reasoning, and acquisition.

Course Outcomes (CO):

CO	CO Statement	Cognitive Level*	Knowledge Category#	Evaluation Tools used
CO1	Understand basics of Knowledge Engineering	U	C	Instructor-created exams / Quiz
CO2	Apply methodologies and modelling for agent design and development	Ap	P	Assignment / Seminar presentations/ Exams
CO3	Design and develop ontologies	Ap	P	Seminar Presentation / Group Tutorial Work/ Viva Voce
CO4	Apply reasoning with ontologies and rules	Ap	P	Instructor-created exams / Home Assignments
CO5	Understand learning and rule learning	U	C	Writing assignments/ Exams/ Seminar Presentations
CO6	Develop theoretical knowledge to design a knowledge based system	Ap	P	Case Study/ Group discussions/ Presentations

* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C)
# - Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge (M)

Detailed Syllabus:

Module	**Unit**	**Content**	**Hrs (48+12)**	**Marks (70)**
I		**Reasoning under uncertainity**	**15**	**15**

	1	Understanding the World through Evidence-based Reasoning: - Evidence, Data, and Information, Evidence and Fact, Evidence and Knowledge	2	
	2	Abductive Reasoning	1	
	3	Probabilistic Reasoning: - Enumerative Probabilities: Obtained by Counting, Subjective Bayesian View of Probability	2	
	4	Belief Functions	1	
	5	Baconian Probability, Fuzzy Probability	3	
	6	Evidence-based Reasoning	2	
	7	Artificial Intelligence: - Intelligent Agents, Mixed-Initiative Reasoning	2	
	8	Knowledge Engineering: - An Ontology of Problem-Solving Tasks, Building Knowledge-based Agents	2	
II	**Methodologies and Tools for Agent Design and Development ,Modelling the Problem-Solving Process**		**12**	**20**
	9	A Conventional Design and Development Scenario	2	
	10	Development Tools and Reusable Ontologies	2	
	11	Agent Design and Development Using Learning Technology	2	
	12	Problem Solving through Analysis and Synthesis	1	
	13	Inquiry-driven Analysis and Synthesis for Evidence-based Reasoning	2	
	14	Evidence-based Assessment, Believability Assessment	3	
III	**Ontologies**		**11**	**20**
	15	What Is an Ontology? Concepts and Instances, Generalization Hierarchies	2	
	16	Object Features, Defining Features, Defining Features, Representation of N-ary Features	2	
	17	Transitivity, Inheritance, Ontology Matching	3	
	18	Ontology Design and Development Methodology- Steps in Ontology Development, Domain Understanding and Concept Elicitation, Modeling-based Ontology Specification	4	
IV	**Reasoning with Ontologies and Rules**		**10**	**15**
	19	Production System Architecture	1	
	20	Complex Ontology-based Concepts	1	
	21	Reduction and Synthesis Rules and the Inference Engine, Evidence-based Hypotheses Analysis, Rule for Ontology Matching	4	
	22	Partially Learned Knowledge, Reasoning with Partially Learned Knowledge	4	
V	**Open Ended Module- Learning for Knowledge-based Agents**		**12**	
	1	Generalization and Specialization Rules	4	
	2	Types of Generalizations and Specializations	4	
	3	Analogy-based Generalization	4	

References

- “Knowledge Engineering”, Gheorghe Tecuci, Dorin Marcu, Mihai Boicu, David A. Schum

- "Artificial Intelligence: A Modern Approach" by Stuart Russell and Peter Norvig
- "Knowledge Representation and Reasoning" by Ronald J. Brachman and Hector J. Levesque.

Mapping of COs with PSOs and POs :

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PO1	PO2	PO3	PO4	PO5	PO6
CO 1	2	1	-	-	1	1						
CO 2	3	-	-	-	1	1						
CO 3	1	3	1	1	2	3						
CO 4	1	-	1	1	2	3						
CO 5	1	-	-	-	2	3						
CO 6	1	2	1	1	3	3						

Correlation Levels:

Level	Correlation
-	Nil
1	Slightly / Low
2	Moderate / Medium
3	Substantial / High

Assessment Rubrics:

- Quiz / Assignment/ Quiz/ Discussion / Seminar
- Midterm Exam
- Programming Assignments (20%)
- Final Exam (70%)

Mapping of COs to Assessment Rubrics :

	Internal Exam	Assignment	End Semester Examinations
CO 1	✓		✓
CO 2	✓		✓
CO 3	✓		✓
CO 4	✓	✓	✓
CO 5	✓	✓	✓
CO 6	✓	✓	

Programme	B. Sc. Computer Science				
Course Code	**CSC6EJ311b**				
Course Title	**Soft Computing**				
Type of Course	**Elective**				
Semester	VI				
Academic Level	300-399				
Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	4	4	-	-	60
Pre-requisites	1. Fundamental Mathematics Concepts: Set, Functions, Logic 2. CSC2CJ101 – Fundamentals of Programming				
Course Summary	This course explores implementations of linked list and array-based data structures, delving into the inner workings of basic data structures including lists, stacks, queues, trees, and graphs.				

Course Outcomes (CO):

CO	CO Statement	Cognitive Level*	Knowledge Category#	Evaluation Tools used
CO1	Understand the foundational principles of soft computing and the historical factors influencing its development.	U	C	Instructor-created exams / Quiz
CO2	Analyze the properties of Fuzzy sets and Fuzzy relations	Ap, U	P	Assignment/ Seminar
CO3	Apply fuzzy logic concepts to solve real-world problems, showcasing proficiency in designing and implementing fuzzy systems.	Ap, U	C	Seminar Presentation / Quiz
CO4	Master the concepts of Genetic algorithms and their operations	U	C	Practical Assignment / Seminar
CO5	Design and implement solutions using fuzzy logic, neural networks, and genetic algorithms for diverse applications.	Ap	P	Practical Assignment/ Seminar
CO6	Evaluate and present real-world scenarios where soft computing techniques can be effectively applied	Ap	P	Case study/ Project

* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C)
# - Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge (M)

Detailed Syllabus:

Module	Unit	Content	Hrs
I		**Introduction to Soft Computing**	**7**
	1	Overview of Soft computing, Hard Computing, and Hybrid Computing	2
	2	Areas and Applications of Soft Computing	1
	3	Basic Tools of Soft Computing- Fuzzy Logic, Neural Networks and Evolutionary computing	2
	4	Introduction to Fuzzy logic, Neural Networks, Genetic Algorithm, and Hybrid systems (Concepts only)	2
II		**Introduction to Fuzzy Logic**	**14**
	6	Introduction to Fuzzy Logic	2
	7	Fuzzy sets and crisp sets	2
	8	Fuzzy relations and Crisp relations	2
	9	Tolerance and Equivalence Relations	2
	10	Fuzzy membership functions	3
	11	Fuzzification and Defuzzification	3
III		**Advanced Fuzzy Logic**	**14**
	12	Fuzzy Rules and Fuzzy Reasoning	3
	13	Fuzzy Inference Systems- Mamdani and Sugeno models	4
	14	Fuzzy Control Systems	3
	15	Fuzzy Clustering (Concepts only)	2
	16	Fuzzy Neural Networks (Concepts only)	2
IV		**Genetic Algorithm**	**13**
	17	Introduction to Genetic Algorithm	2
	18	Operators in genetic algorithm - coding - selection - cross over – mutation,	2
	19	Stopping condition for genetic algorithm flow.	2
	20	Constraints in Genetic Algorithm	2
	21	Classification of Genetic Algorithm	3
	22	Genetic Programming (Concepts)	2
V		**Open Ended Module**	**12**
		• Understand the different optimization techniques used. • Explore the real-life applications of soft computing techniques • Discuss hybrid soft computing techniques	

REFERENCES

1. S.N.Sivanandam and S.N.Deepa, "Principles of Soft Computing", Wiley India Pvt Ltd
2. D.K. Pratihar, “Soft Computing: Fundamentals and Applications”, Alpha Science International Ltd

Mapping of COs with PSOs and POs :

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PO1	PO2	PO3	PO4	PO5	PO6
CO 1	2	-	-	-	1	1						
CO 2	2	-	-	1	1	1						
CO 3	2	-	-	2	2	1						
CO 4	2	-	-	1	1	1						
CO 5	1	-	2	3	2	3						
CO 6	1	-	3	3	2	3						

Correlation Levels:

Level	**Correlation**
-	Nil
1	Slightly / Low
2	Moderate / Medium
3	Substantial / High

Assessment Rubrics:

- Assignment/ Quiz/ Discussion / Seminar
- Midterm Exam
- Programming Assignments (20%)
- Final Exam (70%)

Mapping of COs to Assessment Rubrics :

	Internal Exam	Assignment	Practical Evaluation	End Semester Examinations
CO 1	✓			✓
CO 2	✓	✓		✓
CO 3	✓	✓		✓
CO 4	✓	✓		✓
CO 5		✓		✓
CO 6			✓	

Programme	B. Sc. Computer Science
Course Code	**CSC6EJ312b**
Course Title	**Deep Learning**
Type of Course	**Elective**
Semester	VI
Academic Level	300 - 399

Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	4	4	-	-	60

Pre-requisites	4. Introduction to Artificial Intelligence 5. Basic understanding of linear algebra, calculus, and probability. 6. Basics of Machine learning
Course Summary	The theoretical groundwork for comprehending the fundamentals of deep learning is supplied by this course. Theoretical frameworks, optimisation techniques, and mathematical ideas that support deep neural network building and training will be examined by students.

Course Outcomes (CO):

CO	CO Statement	Cognitive Level*	Knowledge Category#	Evaluation Tools used
CO1	Master key concepts of machine learning, understanding various layers of neural network.	U	C	Instructor-created exams / Quiz
CO2	Understand and implement the backpropagation algorithm for training neural networks, demonstrating the ability to compute gradients and update weights.	Ap, U	P	Assignment / Seminar presentations/ Exams
CO3	Analyze and compare different activation functions used in neural networks, explaining their role in the learning process.	U	P	Seminar Presentation / Group Tutorial Work/ Viva Voce
CO4	Design and implement feedforward neural networks for various applications, considering aspects such as model architecture, activation functions, and initialization methods.	Ap	C	Instructor-created exams / Home Assignments
CO5	Master the principles of convolutional neural networks, including convolutional layers, pooling layers, and their applications in computer vision. Master various regularization techniques, such as dropout, batch normalization, and weight regularization, to improve the generalization of neural networks	U	P	Writing assignments/ Exams/ Seminar Presentations

CO6	Apply deep learning concepts to solve real-world problems, demonstrating the ability to choose appropriate architectures and hyperparameters.	Ap	P	Case Study
* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C) # - Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge (M)				

Detailed Syllabus:

Module	Unit	Content	Hrs (48+12)	Marks (70)
I		**Machine Learning Basics**	**10**	**15**
	1	Learning Algorithms -Supervised learning- regression, classification, Unsupervised learning, Reinforcement learning (Introduction only)	2	
	2	Terms - Capacity, Overfitting and Underfitting, Hyperparameters and Validation Sets, Estimators, Bias and Variance	2	
	3	Maximum Likelihood estimation, Bayesian statistics, Stochastic Gradient Descent	3	
	4	Building a Machine Learning Algorithm	1	
	5	Challenges Motivating Deep Learning	2	
II		**Optimisation and Neural Networks**	**15**	**20**
	6	Neural Networks –Perceptron, Gradient Descent solution for Perceptron, Multilayer perceptron	3	
	7	Activation Functions- Sigmoid, Softmax, Relu, LeakyRelu, ERELU	2	
	8	Chain rule, back propagation- Backpropagation Algorithm	3	
	9	Gradient based learning.	2	
	10	Introduction to optimization– Gradient based optimization, linear least squares. Stochastic gradient descent	2	
	11	Regularisation techniques- Drop out, Batch Normalisation, weight regularisation	3	
III		**Convolutional Neural Network**	**12**	**20**
	12	Convolutional Neural Networks – convolution operation, motivation	2	
	13	Pooling	2	
	14	Variants of convolution functions	2	
	15	Structured outputs, data types	2	
	16	CNN Architecture- Alexnet, VGG16	4	
IV		**Deep learning Architectures**	**11**	**15**
	17	Sequence Modeling: Recurrent and Recursive Nets- Basics of Recurrent Neural Networks	2	
	18	Encoder – Decoder Sequence to Sequence Architectures,	2	
	19	Deep Recurrent Networks, Recursive Neural Networks	2	
	20	The Long Short-Term Memory	2	
	21	GRU	2	

	22	Basics of transfer learning techniques (Concept only)	1	
V		**Open ended Module**	**12**	
	1	• Master students Basics of Mathematics required for Machine learning and deep learning- Linear Algebra (Scalars, Vectors, Matrices and Tensors, Eigen values, Eigen Vectors)- concepts only • Probability awareness- Why probability, random variable, probability distributions)- concepts only • Discuss advanced topics in deep learning, including transfer learning, autoencoders, adversarial training, and stay informed about recent developments in the field.)- concepts only		

References

- "Deep Learning" by Ian Goodfellow, Yoshua Bengio, and Aaron Courville.
- Aston Zhang, Zachary C. Lipton, Mu Li, and Alexander J. Smola, "Dive into Deep Learning", August 2019.
- Neural Networks and Deep Learning: A Textbook by Charu C. Aggarwal. Springer.1st edition, 2018.
- "Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow" by Aurélien Géron.

Mapping of COs with PSOs and POs :

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PO1	PO2	PO3	PO4	PO5	PO6
CO 1	1	-	-	-	2	3						
CO 2	2	-	1	1	2	3						
CO 3	2	-	-	-	2	1						
CO 4	2	-	1	1	2	2						
CO 5	2	-	2	1	2	3						
CO 6	2	-	2	1	2	3						

Correlation Levels:

Level	Correlation
-	Nil
1	Slightly / Low
2	Moderate / Medium
3	Substantial / High

Assessment Rubrics:

- Quiz / Assignment/ Quiz/ Discussion / Seminar
- Midterm Exam
- Programming Assignments (20%)
- Final Exam (70%)

Mapping of COs to Assessment Rubrics :

	Internal Exam	Assignment	End Semester Examinations
CO 1	✓		✓
CO 2	✓		✓
CO 3	✓		✓
CO 4		✓	✓
CO 5		✓	✓
CO 6		✓	

Cloud Computing									
No	**Course Code**	**Course Name**	**C**	**Marks**			**Hrs/wk**		
				I	**E**	**T**	**L**	**P**	**T**
29	CSC5EJ305c	Cloud Computing	4	30	70	100	4	0	4
30	CSC5EJ306c	Security and Privacy in Cloud	4	30	70	100	4	0	4
35	CSC6EJ311c	Storage Technologies	4	30	70	100	4	0	4
36	CSC6EJ312c	Virtualization	4	30	70	100	4	0	4

Programme	B. Sc. Computer Science				
Course Code	**CSC5EJ305c**				
Course Title	**Cloud Computing**				
Type of Course	**Elective**				
Semester	V				
Academic Level	300 - 399				
Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	4	4	-	-	60
Pre-requisites	7. Basic understanding of computer networks, operating systems, and programming.				
Course Summary	This course introduces students to the fundamental concepts, technologies, and practices of cloud computing. It covers the basics of cloud infrastructure, deployment models, and service models.				

Course Outcomes (CO):

CO	CO Statement	Cognitive Level*	Knowledge Category#	Evaluation Tools used
CO1	Understand fundamentals of cloud Computing	U	C	Instructor-created exams / Quiz
CO2	Describe and compare Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS)	U	C	Assignment / Seminar presentations/ Exams
CO3	Analyze various deployment models such as public, private, and hybrid clouds.	U	P	Seminar Presentation / Group Tutorial Work/ Viva Voce

CO4	Understand the principles of virtualization and its role in cloud computing.	U	C	Instructor-created exams / Home Assignments
CO5	Compare and contrast different virtualization technologies, including hypervisors and containerization.	U	P	Writing assignments/ Exams/ Seminar Presentations
CO6	Explore various cloud platforms in industry	U	F	Case Study/ Exams
* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C) # - Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge (M)				

Detailed Syllabus:

Module	Unit	Content	Hrs (48+12)	Marks (70)
I		**Introduction to cloud computing**	**8**	**12**
	1	Cloud computing in a glance	2	
	2	Historical context and evolution	1	
	3	Building cloud computing environments- Cloud components	2	
	4	Desired features of cloud	2	
	5	Advantages of Cloud	1	
II		**Cloud computing architecture**	**14**	**20**
	6	Cloud reference model	4	
	7	Types of cloud- private, public, hybrid, community	3	
	8	Cloud service models (IaaS)	2	
	9	Cloud service models (PaaS)	2	
	10	Cloud service models (SaaS)	2	
	11	Open Challenges	1	
III		**Virtualization Technologies**	**16**	**23**
	12	Virtual machine basics	2	
	13	hypervisor	2	
	14	Virtualisation structure	3	
	15	Implementation levels of virtualisation	2	
	16	Virtualisation types- Full Virtualisation, Para Virtualisation, Hardware Virtualisation	3	
	17	Virtualisation of CPU, Memory	2	
	18	Virtualisation of I/O devices	2	
IV		**Virtualisation infrastructure & Dockers**	**10**	**15**
	17	Desktop Virtualisation, Network Virtualisation & Storage Virtualisation	2	
	18	Containers vs Virtual Machines	2	

	19	Basics of Dockers	2	
	20	Docker Components	2	
	21	Docker Containers	1	
	22	Docker Images and repositories	1	
V	**Open Ended Module**		**12**	
	1	• Cloud platforms in Industry ✓ **Amazon web services**- computation services, storage services, communication services ✓ **Google AppEngine**- Architecture and core concepts ✓ **Microsoft Azure**- Azure core concepts		

References

1. “Mastering cloud computing”. Rajkumar Buyya
2. “Cloud Computing: Principles and Paradigms”, Rajkumar Buyya, James Broberg, Andrzej Goscinski
3. "Cloud Computing: Concepts, Technology & Architecture" by Thomas Erl
4. “Introduction to Cloud Computing”, William Voorsluys, James Broberg, Rajkumar Buyya
5. "Cloud Computing: A Hands-On Approach" by Arshdeep Bahga and Vijay Madiset

Mapping of COs with PSOs and POs :

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PO1	PO2	PO3	PO4	PO5	PO6
CO 1	-	2	--	-	1	1						
CO 2	-	2	-	-	1	1						
CO 3	-	1	-	-	1	1						
CO 4	-	1	-	-	2	1						
CO 5	-	1	-	-	2	1						
CO 6	-	1	-	-	2	2						

Correlation Levels:

Level	Correlation

-	Nil
1	Slightly / Low
2	Moderate / Medium
3	Substantial / High

Assessment Rubrics:

- Quiz / Assignment/ Quiz/ Discussion / Seminar
- Midterm Exam
- Programming Assignments (20%)
- Final Exam (70%)

Mapping of COs to Assessment Rubrics :

	Internal Exam	Assignment	End Semester Examinations
CO 1	✓		✓
CO 2	✓		✓
CO 3	✓		✓
CO 4		✓	✓
CO 5		✓	✓
CO 6		✓	

Programme	B. Sc. Computer Science
Course Code	**CSC5EJ306c**
Course Title	**Security and Privacy in Cloud**
Type of Course	**Elective**
Semester	V
Academic Level	300 - 399

Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	4	4	-	-	60

Pre-requisites	8. Basic understanding of computer networks, operating systems, databases, Cloud computing
Course Summary	This course explores the security and privacy challenges in cloud computing environments. Students will learn about the fundamental principles, technologies, and best practices for ensuring the confidentiality, integrity, and availability of data in the cloud. The course also covers legal and ethical considerations related to privacy in cloud computing.

Course Outcomes (CO):

CO	CO Statement	Cognitive Level*	Knowledge Category#	Evaluation Tools used
CO1	Understand fundamentals of security concepts (encryption, decryption)	U	C	Instructor-created exams / Quiz
CO2	Understand security design principles.	U	C	Assignment / Seminar presentations/ Exams
CO3	Analyze various threats to cloud security	U	P	Seminar Presentation / Group Tutorial Work/ Viva Voce
CO4	Understand various cloud security design patterns.	U	C	Instructor-created exams / Home Assignments
CO5	Explore various access control mechanisms and management schemes to ensure security in cloud.	U	P	Writing assignments/ Exams/ Seminar Presentations
CO6	Explore various levels of security in cloud infrastructure	U	F	Case Study/ Exams
* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C) # - Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge (M)				

Detailed Syllabus:

Module	Unit	Content	Hrs (48+12)	Marks (70)

I		**Fundamentals of Security in Cloud**	**14**	**22**
	1	Overview of Cloud Security- Security services- Confidentiality, Integrity, Authentication, Non repudiation, Access control	2	
	2	Basics of Cryptography	2	
	3	Conventional and public key cryptography	4	
	4	Hash functions	2	
	5	Authentications	2	
	6	Digital Signature	2	
II		**Security Design and Architecture for Cloud**	**12**	**18**
	7	Security design principles for cloud computing- comprehensive data protection, end to end access control	2	
	8	Common attack vectors and threats	1	
	9	Network and storage- Secure Isolation strategies, Virtualisation strategies, inter- tenant network segmentation strategies, data protection strategies	3	
	10	Data retention, detection and archiving procedures for tenant data	2	
	11	Encryption, Redaction, Tokenisation, Obfuscation	2	
	12	PKI and key	2	
III		**Access Control and Identity Management**	**12**	**18**
	13	Access control requirements for Cloud infrastructure- user identification, authentication and authorization	2	
	14	Role based access control- multi-factor authentication, single Sign-on	2	
	15	Identity providers and service consumers	2	
	16	Storage and network access control options- OS Hardening and minimization	3	
	17	Intruder detection and prevention	3	
IV		**Cloud Security Design patterns**	**10**	**12**
	18	Introduction to design patterns	2	
	19	Cloud bursting	2	
	20	Geo-tagging	2	
	21	Secure cloud interfaces	2	
	22	Cloud resource access control	2	
V		**Open Ended Module**	**12**	
	1	Infrastructure security: Network level, host level, application level	4	
	2	Security management in the cloud	4	
	3	Audit and compliance	4	

References

1. "Cloud Security and Privacy: An Enterprise Perspective on Risks and Compliance" by Tim Mather, Subra Kumaraswamy, and Shahed Latif
2. “Cloud computing: Principles and Paradigms”. Rajkumar Buyya, James Broberg, Andrzej M. Goscinski, Willey Publications

Mapping of COs with PSOs and POs :

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PO1	PO2	PO3	PO4	PO5	PO6
CO 1	-	2	--	-	1	1						
CO 2	-	2	-	-	1	1						
CO 3	-	1	-	-	1	1						
CO 4	-	1	-	-	2	1						
CO 5	-	1	-	-	2	1						
CO 6	-	1	-	-	2	2						

Correlation Levels:

Level	Correlation
-	Nil
1	Slightly / Low
2	Moderate / Medium
3	Substantial / High

Assessment Rubrics:

- Quiz / Assignment/ Quiz/ Discussion / Seminar
- Midterm Exam
- Programming Assignments (20%)
- Final Exam (70%)

Mapping of COs to Assessment Rubrics :

	Internal Exam	Assignment	End Semester Examinations
CO 1	✓		✓
CO 2	✓		✓
CO 3	✓		✓
CO 4		✓	✓
CO 5		✓	✓
CO 6		✓	

Programme	B. Sc. Computer Science				
Course Code	**CSC6EJ311c**				
Course Title	**Storage Technologies**				
Type of Course	**Elective**				
Semester	VI				
Academic Level	300 - 399				
Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	4	4	-	-	60
Pre-requisites	9. Basic knowledge of computer systems and architecture 10. Fundamental understanding of data structures and algorithms				
Course Summary	This course introduces students to various storage technologies, storage network technologies, storage and virtualization technologies. Course also discuss various back up and recovery strategies.				

Course Outcomes (CO):

CO	CO Statement	Cognitive Level*	Knowledge Category#	Evaluation Tools used
CO1	Understand fundamentals of Information storage	U	C	Instructor-created exams / Quiz
CO2	Examine features of various storage architectures	U	C	Assignment / Seminar presentations/ Exams
CO3	Understand features of Intelligent storage systems	U	P	Seminar Presentation / Group Tutorial Work/ Viva Voce
CO4	Identify features of various Storage technologies	U	C	Instructor-created exams / Home Assignments
CO5	Identify need of backup and recovery and various recovery mechanisms	U	P	Writing assignments/ Exams/ Seminar Presentations
CO6	Infer security needs and management needs for storage technologies	U	F	Case Study/ Exams

* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C)
# - Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge (M)

Detailed Syllabus:

Module	Unit	Content	Hrs (48+12)	Marks (70)
I		**Storage System**	**12**	**18**
	1	**Introduction to Information Storage-** Information Storage, Evolution of Storage Architecture	2	
	2	Data Center Infrastructure and characteristics	1	
	3	Third platform technologies- Cloud storage and its characteristics	2	
	4	Cloud services and deployment models	3	
	5	Storage Architectures- Direct-Attached Storage (DAS) Network-Attached Storage (NAS) (Introduction only)	2	
	6	Storage Area Network (SAN) Cloud storage architectures(Introduction only)	2	
II		**Intelligent Storage Systems & RAID**	**12**	**18**
	7	RAID Implementation Methods, RAID Array Components, RAID Techniques	2	
	8	RAID Levels, RAID Impact on Disk Performance	3	
	9	RAID Comparison	1	
	10	Components of an Intelligent Storage System	1	
	11	Storage Provisioning	2	
	12	Types of Intelligent Storage Systems	3	
III		**Storage Networking Technologies - Fibre Channel Storage Area Networks**	**12**	**18**
	13	Block based stored system, File based storage system, object oriented based storage system (Introduction)	2	
	14	Fibre Channel Storage Area Networks- Components of FC SAN,	2	
	15	Fibre Channel Architecture	2	
	16	Fabric Services	2	
	17	FC SAN Topologies	2	
	18	Virtualization in SAN	2	
IV		**Backup and Archive**	**12**	**16**
	19	Backup Purpose, Backup Considerations, Back up Granularity	3	
	20	Recovery Considerations , Backup Methods	3	
	21	Backup Architecture, Backup Topologies	3	
	22	Backup and Restore Operations	3	
V		**Open Ended Module**	**12**	
	1	Storage Security Domains	3	
	2	Security Implementations in Storage Networking	3	
	3	Securing Storage Infrastructure in Virtualized and Cloud Environments	3	
	4	Storage Infrastructure Management Activities	3	

References

- Information Storage and Management: Storing, Managing, and Protecting Digital Information in Classic, Virtualized, and Cloud Environments, 2nd Edition, Willey Publications

Mapping of COs with PSOs and POs :

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PO1	PO2	PO3	PO4	PO5	PO6
CO 1	-	2	--	-	1	1						
CO 2	-	2	-	-	1	1						
CO 3	-	1	-	-	1	1						
CO 4	-	1	-	-	2	1						
CO 5	-	1	-	-	2	1						
CO 6	-	-	-	-	2	2						

Correlation Levels:

Level	Correlation
-	Nil
1	Slightly / Low
2	Moderate / Medium
3	Substantial / High

Assessment Rubrics:

- Quiz / Assignment/ Quiz/ Discussion / Seminar
- Midterm Exam
- Programming Assignments (20%)
- Final Exam (70%)

Mapping of COs to Assessment Rubrics :

	Internal Exam	Assignment	End Semester Examinations
CO 1	✓		✓
CO 2	✓		✓
CO 3	✓		✓
CO 4	✓	✓	✓
CO 5	✓	✓	✓
CO 6	✓	✓	

Programme	B. Sc. Computer Science				
Course Code	**CSC6EJ312c**				
Course Title	**Virtualization**				
Type of Course	**Elective**				
Semester	VI				
Academic Level	300 - 399				
Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	4	4	-	-	60
Pre-requisites	11. Basic understanding of cloud computing				
Course Summary	This course introduces students to the fundamental concepts, technologies, virtualization, various virtualization tools and virtualization in storage, desktop, network and server				

Course Outcomes (CO):

CO	CO Statement	Cognitive Level*	Knowledge Category#	Evaluation Tools used
CO1	Understand basics of virtualization	U	C	Instructor-created exams / Quiz
CO2	Understand how hypervisors work and their role in virtualization.	Ap	P	Assignment / Seminar presentations/ Exams
CO3	Understand Differences between various types of virtualization, including server virtualization, desktop virtualization, network virtualization, and storage virtualization	Ap	C	Seminar Presentation / Group Tutorial Work/ Viva Voce
CO4	Explore how virtualization technologies are used in the context of cloud services.	U	P	Instructor-created exams / Home Assignments
CO5	Understand the potential risks and vulnerabilities associated with virtualization and learn how to mitigate them.	U	P	Writing assignments/ Exams/ Seminar Presentations
CO6	Compare and analyse various virtualization tools	U	F	Case Study/ Exams
* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C) # - Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge (M)				

Detailed Syllabus:

Module	Unit	Content	Hrs (48+12)	Marks (70)
I		**Introduction to Virtualisation**	**12**	**18**
	1	Virtualization and computing- need for virtualisation,	2	
	2	Cost, administration,	2	
	3	Fast deployment, reduce infrastructure cost	2	
	4	Limitations	1	
	5	Types of hardware virtualization: full virtualisation, partial virtualization, paravirtualization	3	
	6	Types of hypervisors	2	
II		**Server and Desktop virtualization**	**14**	**20**
	7	Virtual machine basics	2	
	8	Types of virtual machines	2	
	9	Understanding server virtualisation- types of server virtualization	3	
	10	Business cases for server virtualization	2	
	11	Uses of virtual server consolidation,	2	
	12	Selecting server virtualisation platform	1	
	13	Desktop virtualisation- types of desktop virtualization	2	
III		**Network Virtualisation**	**12**	**18**
	14	Introduction to network virtualisation	2	
	15	Advantages, functions	2	
	16	Tools for network virtualization	3	
	17	VLAN-WAN architecture	2	
	18	WAN Visualization	3	
IV		**Storage Virtualization**	**10**	**16**
	19	Introduction to memory virtualization	2	
	20	Types of storage virtualization	3	
	21	Risk of storage virtualization	2	
	22	SAN-NAS-RAID	3	
V		**Open Ended Module- Virtualization tools (Any 3- $ hours each)**	**12**	
		VMWare-Amazon AWS Microsoft HyperV Oracle VM Virtual box IBM PowerVM Google Virtualization		

References

- **Cloud Computing a practical approach- Anthony T Velte, Toby T Velte, Robert Elsenpeter, Tata McGraw Hill**
- **Virtualization from Desktop to the Enterprise, Chris Wolf, Eric M Halter**

Mapping of COs with PSOs and POs :

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PO1	PO2	PO3	PO4	PO5	PO6
CO 1	-	2	--	-	1	1						
CO 2	-	2	-	-	1	1						
CO 3	-	1	-	-	1	1						
CO 4	-	1	-	-	2	1						
CO 5	-	1	-	-	2	1						
CO 6	-	1	-	-	2	2						

Correlation Levels:

Level	**Correlation**
-	Nil
1	Slightly / Low
2	Moderate / Medium
3	Substantial / High

Assessment Rubrics:

- Quiz / Assignment/ Quiz/ Discussion / Seminar
- Midterm Exam
- Programming Assignments (20%)
- Final Exam (70%)

Mapping of COs to Assessment Rubrics :

	Internal Exam	Assignment	End Semester Examinations
CO 1	✓		✓
CO 2	✓		✓
CO 3	✓		✓
CO 4		✓	✓
CO 5		✓	✓
CO 6	✓	✓	

General									
No	Course Code	Course Name	C	Marks			Hrs/wk		
				I	E	T	L	P	T
1	CSC8EJ401	Microprocessor and its Applications	4	30	70	100	4	0	4
2	CSC8EJ402	System Software	4	30	70	100	4	0	4
3	CSC8EJ403	Social Network Analysis	4	30	70	100	4	0	4
4	CSC8EJ404	Advanced Distributed Computing	4	30	70	100	4	0	4
5	CSC8EJ405	Cyber Forensic	4	30	70	100	4	0	4
6	CSC8EJ406	Ethical Hacking	4	30	70	100	4	0	4

Programme	B. Sc. Computer Science				
Course Code	**CSC8EJ401**				
Course Title	**Microprocessor and its Applications**				
Type of Course	**Elective**				
Semester	VIII				
Academic Level	400-499				
Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	4	4	-	-	60
Pre-requisites	-				
Course Summary	This course provides a comprehensive understanding of microprocessors with a specific focus on Intel's 8085 & 8086 architectures. The course delves into the key principles, features, and programming techniques associated with 8086. The feature comparison of advanced processors gives an overview of developments in Microprocessor technology.				

Course Outcomes (CO):

CO	CO Statement	Cognitive Level*	Knowledge Category#	Evaluation Tools used
CO1	Recognise the purpose, characteristics, and architecture of the 8085 and 8086.	U	C	Instructor-created exams / Quiz
CO2	Identify the addressing modes and comprehend how the 8086 instructions work.	Ap	P	Practical Assignment / Observation of Practical Skills
CO3	Illustrate simple assembly language programs.	Ap	P	Programming Assignment / Observation of Practical Skills
CO4	Indentify the functions of peripheral integrated circuits (ICs) and how interrupts are handled in the 8086.	U	C	Instructor-created exams / Seminars
CO5	Describe the characteristics of advanced microprocessors.	U	C	Instructor-created exams / Home Assignments
* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C) # - Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge (M)				

Detailed Syllabus:

Module	Unit	Content	Hrs (48+12)	Marks (70)
I		**Introduction to Microprocessors**	**8**	**15**
	1	Basic Architecture of a Computer System	1	
	2	Advances in Semiconductor Technology, Evolution of Microprocessors	2	
	3	Overview of Microprocessors vs Microcontrollers, Computer Languages: High Level, Machine Language, Assembly Language	1	
	4	8085 Microprocessor (Architecture and Pin diagram)	4	
II		**Basics of 8086 Architecture**	**9**	**20**
	5	Features of an n-bit microprocessor, 8086 Architecture (Block diagram, Register Configuration, Address Translation)	5	
	6	8086 Pin Configuration	2	
	7	Minimum and Maximum Mode Configuration	2	
III		**8086 Programming**	**23**	**20**
	8	8086 Addressing Modes	2	
	9	8086 Instruction Set: Data Transfer and Arithmetic Instruction	4	
	10	8086 Instruction Set: Branch and Loop Instructions	3	
	11	8086 Instruction Set: Sting Instructions	2	
	12	8086 Instruction Set: Processor Control Instructions	1	

	13	Assembler Directives	1	
	14	Sample Programs: 1) Assembly Program to find the sum of n numbers given 2) Assembly Program to perform division using repeated subtraction 3) Assembly Program to multiply two 16 bit numbers 4) Assembly Program to find the largest of n numbers given 5) Assembly Program to perform linear search in a set of numbers given. Also find the number occurrence of the searching element. 6) Assembly Program to perform comparison of two strings.	6	
	15	8086 Interrupts and Interrupt Service Routines	2	
	16	Procedures and Macros	2	
IV		**Advanced Microprocessors** *(Study of Architecture and Pin diagram not needed)*	**8**	**15**
	17	Features of Intel 80186 & 80286	2	
	18	Features of Intel 80386 & 80486	2	
	19	Features of Pentium Processors	1	
	20	Features of Multi Core Processors	1	
	21	Features of i series Processors	1	
	22	Features of Mobile Processors	1	
V		**Open Ended Module: 8086 Interfacing**	**12**	
	1	**Introduction to peripheral Interfacing** **Include Case studies of any 3 interfacing ICs like:** 1. Programmable Peripheral Interface (8255) 2. Programmable DMA Controller (8257) 3. Programmable Interrupt Controller (8259) 4. Programmable Interval Timer (8253) 5. Interfacing output displays (8212) 6. Programmable communication interface (8251A)	12	

Mapping of COs with PSOs and POs :

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PO1	PO2	PO3	PO4	PO5	PO6
CO 1	2	3	-	-	-	-						
CO 2	2	2	-	1	-	-						
CO 3	-	-	-	2	-	-						
CO 4	2	2	-	-	-	-						
CO 5	1	1	-	-	-	1						

Correlation Levels:

Level	Correlation
-	Nil
1	Slightly / Low
2	Moderate / Medium
3	Substantial / High

Assessment Rubrics:

- Quiz / Assignment/ Quiz/ Discussion / Seminar
- Midterm Exam
- Programming Assignments (20%)
- Final Exam (70%)

Mapping of COs to Assessment Rubrics :

	Internal Exam	Assignment/Seminar	Programming Assignments	End Semester Examinations
CO 1	✓			✓
CO 2	✓		✓	✓
CO 3	✓		✓	✓
CO 4	✓	✓		
CO 5		✓		✓

Programme	B. Sc. Computer Science				
Course Code	**CSC8EJ402**				
Course Title	**System Software**				
Type of Course	Elective				
Semester	VIII				
Academic Level	400 - 499				
Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	4	4	-	-	60
Pre-requisites	13. Introduction to Computer Science 14. Data Structures and Algorithms 15. Computer Organization and Architecture				
Course Summary	With an emphasis on the creation and use of system software, this course examines the ideas and methods of system programming. Compiler design, system calls, loaders and linkers, and debugging methods are among the topics covered.				

Course Outcomes (CO):

CO	CO Statement	Cognitive Level*	Knowledge Category#	Evaluation Tools used
CO1	Define the key concepts in system programming, such as compilers, assemblers, linkers, and loaders. Identify the various stages in the compilation process and understand the purpose of each stage.	U	C	Instructor-created exams / Quiz
CO2	Master different types of system calls and their role in system programming. Master the principles of lexical and syntax analysis in the context of compiler design. Master various linking and loading schemes	Ap	P	Assignment / Seminar presentations/ Exams
CO3	Interpret and understand the process of debugging, including the use of debugging tools and techniques.	Ap	P	Seminar Presentation / Group Tutorial Work/ Viva Voce
CO4	Analyze the impact of different optimization techniques in the compilation process. Evaluate advantages and disadvantages of various linking and loading schemes.	U	C	Instructor-created exams / Home Assignments

CO5	Implement programs using system calls to perform various system-	Ap	P	Writing assignments
	level tasks, such as file operations and process management			
CO6	Apply principles of compiler design to write a simple compiler using a programming language.	Ap	P	Case Study
* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C) # - Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge (M)				

Detailed Syllabus:

Module	Unit	Content	Hrs
I		**Introduction to System Programming & Assemblers**	**14**
	1	Introduction to System Programming- Goals of System Software, System Programs and Systems Programming	3
	2	Language Processors- Overview, Kinds of Language processors, language processing activities, program execution	4
	3	System Tables	1
	4	Assemblers- Elements of Assembly Language Programming	2
	5	Design of two pass assembler	4
II		**Macros and Macro Processors**	**10**
	6	Introduction to macros and macro processors- macro definition and call, macro expression	4
	7	Nested macro calls	2
	8	Design of macro processor	4
III		**Linkers and Loaders, Scanning and Parsing**	**10**
	9	Relocation and linking concepts	2
	10	Design of linkers	2
	11	Self locating program	2
	12	Loaders- absolute loader, relocating loader	4
IV		**Compilers, System calls and libraries**	**14**
	13	Data structures used in compilers	1
	14	Phases of a compiler – Introduction	1
	15	Lexical Analysis (Scanning)	2
	16	Syntax Analysis (Parsing)	2
	17	Semantic Analysis	1
	18	Intermediate code generation	1
	19	Code optimisation- optimisation transformation, local optimisation, global optimisation, Code Generation	2
	20	Passes of Compiler	1
	21	System calls and their implementation	1
	22	Standard C library functions for system calls	2
V		**Open Ended Module: Case Studies**	**12**
	1	• Case studies of lexical and syntax analyzers: LEX and YAAC. • System programs using system calls	

References

- D.M. Dhamdhere, Systems Programming and Operating Systems
- John J Donovan, Systems programming
- Jim Welsh and R M Mckeag, Structured System Programming, Prentice Hall.

Mapping of COs with PSOs and POs :

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PO1	PO2	PO3	PO4	PO5	PO6
CO 1	2	1	-	1	-	-						
CO 2	2	3	-	1	-	-						
CO 3	1	-	1	1	1	-						
CO 4	2	2	1	1	-	-						
CO 5	2	3	1	-	-	-						
CO 6	2	3	1	-	-	-						

Correlation Levels:

Level	Correlation
-	Nil
1	Slightly / Low
2	Moderate / Medium
3	Substantial / High

Assessment Rubrics:

- Quiz / Assignment/ Quiz/ Discussion / Seminar
- Midterm Exam
- Programming Assignments (20%)
- Final Exam (70%)

Mapping of COs to Assessment Rubrics :

	Internal Exam	Assignment	End Semester Examinations
CO 1	✓		✓
CO 2	✓		✓
CO 3	✓		✓
CO 4		✓	✓
CO 5		✓	✓
CO 6		✓	

Programme	B. Sc. Computer Science				
Course Code	**CSC8EJ403**				
Course Title	**Social Networks Analysis**				
Type of Course	Elective				
Semester	VIII				
Academic Level	400 - 499				
Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	4	4	-	-	60
Pre-requisites	Knowledge in Fundamentals of Data Mining				
Course Summary	The syllabus is prepared with the view of preparing the BSc Computer Science Graduates to build a basic understanding of what social network analysis is and how it can be applied. Topics covered include network structure and methods for social network analysis, link analysis and network community detection, information propagation on the web and some applications				

Course Outcomes (CO):

CO	CO Statement	Cognitive Level*	Knowledge Category#	Evaluation Tools used
CO1	Understand the basic notation and terminology used in social network analysis.	U	C	Instructor-created exams / Quiz
CO2	Compare and interpret social network structure, size and its connectivity pattern.	U	C	Instructor-created exams / Quiz
CO3	Discover community structure in complex network using statistical techniques	U	P	Instructor-created exams / Quiz
CO4	Apply link prediction techniques to discover new links in the social network	U	C	Instructor-created exams / Case studies
CO5	Describe influence in social media, perform recommendations	U	P	Instructor-created exams / Quiz Case studies
CO6	Perform Social Influence Analysis	U	P	Instructor-created exams / Quiz / Case studies
* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C)# - Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge (M)				

Detailed Syllabus:

Module	Unit	Content	Hrs	Marks
I		**Introduction to Social Network Data Analytics**	**12**	**15**
	1	Introduction to Social Network Analysis	2	
	2	Online social networks Research Issues and Topics	2	
	3	Statistical properties of social networks: Preliminaries	2	
	4	Static properties, Dynamic properties	3	
	5	Challenges of Social Network Streams	3	
II		**Random Walks in Social Networks**	**12**	**15**
	6	Random walks on Graphics, Walks based on proximity measures	2	
	7	Other graph based proximity measures	1	
	8	Graph theoretic measures for semi supervised learning	3	
	9	Clustering with random walk based measures	3	
	10	Applications in computer vision Text Analysis, Evaluation and datasets	1	
	11	Link prediction and data sources	2	
III		**Community Discovery in Social Networks**	**12**	**20**
	12	Communities in Context	2	
	13	Core Methods – KL Algorithm, Special algorithms	2	
	14	Markov Clustering, other approaches	2	
	15	Emerging Fields and problems : Community Discovery in dynamic networks	2	
	16	Heterogeneous networks, Directed networks,	2	
	17	Coupling content and relationship information for community discovery	2	
IV		**Link Prediction in Social Networks**	**12**	**20**
	18	Background, Feature based Link Prediction, Bayesian Probabilistic Models	3	
	19	Probabilistic Relational Models	2	
	20	Linear Algebraic Methods	2	
	21	Link Predictions: The Katz Score, Hitting & Commute Time	2	
	22	Rooted PageRank, SimRank	3	
V		**Open Ended Module**	**12**	
		CASE STUDY: Social Influence Analysis Influence Related Statistics, Social Similarity and Influence, Influence Maximization in Viral Marketing,		

Reference Books:

1. Charu.C. Aggarwal, Social Network Data Analytics, Springer Science+Business Media, LLC 2011.

2. R. Zafarani, M. A. Abbasi, and H. Liu, Social Media Mining: An Introduction, Cambridge University Press, 2014.

3. Krishna Raj P M, Ankith Mohan, K G Srinivasa ,Practical Social Network Analysis with Python , Springer Liu, Bing. Web data mining. Springer-Verlag Berlin Heidelberg, 2007.

4. Chakrabarti, Soumen. Mining the Web: Discovering knowledge from hypertext data. Morgan Kaufmann, 2003.

5. Scime, Anthony, ed. Web mining: applications and techniques. IGI Global, 2005

Mapping of COs with PSOs and POs :

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PO1	PO2	PO3	PO4	PO5	PO6
CO1	-	2	-	1	1	3						
CO2	-	3	-	1	1	2						
CO3	1	3	-	1	1	1						
CO4	1	2	-	-	1	1						
CO5	-	1	1	-	1	1						
CO6	-	1	1	-	1	1						

Correlation Levels:

Level	Correlation
-	Nil
1	Slightly / Low
2	Moderate / Medium
3	Substantial / High

Assessment Rubrics:

- Quiz / Assignment/ Quiz/ Discussion / Seminar
- Midterm Exam
- Assignments (20%)
- Final Exam (70%)

Mapping of COs to Assessment Rubrics :

	Internal Exam	Assignment	Project Evaluation	End Semester Examinations
CO 1	✓			✓
CO 2	✓			✓
CO 3	✓	✓		✓
CO 4		✓		✓
CO 5		✓		✓
CO 6			✓	

Programme	B. Sc. Computer Science				
Course Code	**CSC8EJ404**				
Course Title	**Advanced Distributed Computing**				
Type of Course	Elective				
Semester	VIII				
Academic Level	400 - 499				
Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	4	4	-	-	60
Pre-requisites	Basic knowledge in data structures and operating systems.				
Course Summary	The syllabus is prepared with the view of preparing the BSc Computer Science Graduates to understand the system models, algorithms and protocols that allow computers to communicate and coordinate their actions to solve a problem. This course helps the learner to understand the distributed computation model and various concepts like global state, termination detection, mutual exclusion, deadlock detection, shared memory				

Course Outcomes (CO):

CO	CO Statement	Cognitive Level*	Knowledge Category#	Evaluation Toolsused
CO1	Summarize various aspects of distributed computation model	U	C	Instructor-createdexams / Quiz
CO2	Appreciate and apply Distributed Computing and Communication design principles	U	C	Instructor-createdexams / Quiz
CO3	Illustrate election algorithm, global snapshot algorithm and termination detection algorithm.	U	P	Instructor-createdexams / Quiz
CO4	Compare token based, non-token based and quorum based mutual exclusion algorithms.	U	C	Instructor-created exams / Case studies
CO5	Recognize the significance of deadlock detection and shared memory in distributed systems	U	P	Instructor-created exams / Quiz Case studies
CO6	Understand the concepts of failure recovery and consensus	U	P	Instructor-createdexams / Quiz /Case studies
* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C)# - Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge (M)				

Detailed Syllabus:

Module	Unit	Content	Hrs	Marks
I		**Introduction to Distributed Systems**	**12**	**15**
	1	Introduction to Distributed Systems: Goals of the Distributed Systems, Relation to parallel systems	2	
	2	Synchronous versus asynchronous execution, Design issues and challenges	2	
	3	Types of Distributed Systems	2	
	4	Distributed System Models	3	
	5	Hardware and software concepts related to distributed systems, middleware models .	3	
II		**Distributed Computing and Communication**	**12**	**15**
	6	Distributed Computing and Communication design principles: A Model of distributed executions	2	
	7	Models of communication networks, Global state of distributed system,	1	
	8	Models of process communication.	3	
	9	Communication and Coordination: Shared Memory, Consistency, Atomicity	3	
	10	Message- Passing, Consensus, Conditional Actions, Critical Paths	1	
	11	Scalability, and cache coherence in multiprocessor systems, synchronization mechanism.	2	
III		**Election algorithm, Global state and Termination detection**	**12**	**20**
	12	Logical time – A framework for a system of logical clocks, Scalar time, Vector time	2	
	13	Leader election algorithm – Bully algorithm, Ring algorithm	2	
	14	System model and definitions, Snapshot algorithm for FIFO channels – Chandy Lamport algorithm.	2	
	15	Termination detection – System model of a distributed computation,	2	
	16	Termination detection using distributed snapshots,	2	
	17	Termination detection by weight throwing ,Spanning-tree-based algorithm	2	
IV		**Mutual exclusion and Deadlock detection**	**12**	**20**
	18	Distributed mutual exclusion algorithms – System model, Requirements of mutual exclusion algorithm	2	
	19	Quorum-based mutual exclusion algorithms – Maekawa’s algorithm	2	
	20	Token-based algorithm – Suzuki–Kasami’s broadcast algorithm.	3	
	21	Deadlock detection in distributed systems – System model, Deadlock handling strategies	2	
	22	Issues in deadlock detection, Models of deadlocks	3	

V	Open Ended Module	12	
	CASE STUDY: Distributed shared memory and Failure recovery Lamport's bakery algorithm. Check pointing and rollback recovery – System model, consistent and inconsistent states, Different types of messages, Issues in failure recovery, Checkpoint based recovery, Log based roll back recovery.		

Reference Books:

1.Ajay D. Kshemkalyani and Mukesh Singhal, Distributed Computing: Principles, Algorithms, and Systems, Cambridge University Press, 2011

2. George Coulouris, Jean Dollimore, Tim Kindberg and Gordon Blair. Distributed Systems: Concepts and Design, Addison Wesley, Fifth edition.

3. Kai Hwang, Geoffrey C Fox, Jack J Dongarra, Distributed and Cloud Computing – From Parallel Processing to the Internet of Things, Morgan Kaufmann Publishers, 2012.

4. Sukumar Ghosh, Distributed Systems: An Algorithmic Approach, CRC Press, Second edition, 2015.

5. Maarten Van Steen, Andrew S. Tanenbaum, Distributed Systems, Prentice Hall of India,Third edition, 2017.

6. Randy Chow and Theodore Johnson, Distributed Operating Systems and Algorithm Analysis, Pearson Education India, First edition, 2009.

7. Valmir C. Barbosa, An Introduction to Distributed Algorithms, MIT Press, 2003.

Mapping of COs with PSOs and POs :

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PO1	PO2	PO3	PO4	PO5	PO6
CO1	-	2	-	-	-	-						
CO2	-	3	-	-	-	-						
CO3	-	2	1	-	1	1						
CO4	-	2	1	2	1	2						
CO5	-	3	1	-	-	2						
CO6	-	2	-	-	-	2						

Correlation Levels:

Level	Correlation

-	Nil
1	Slightly / Low
2	Moderate / Medium
3	Substantial / High

Assessment Rubrics:

- Quiz / Assignment/ Quiz/ Discussion / Seminar
- Midterm Exam
- Assignments (20%)
- Final Exam (70%)

Mapping of COs to Assessment Rubrics :

	Internal Exam	Assignment	Project Evaluation	End Semester Examinations
CO 1	✓			✓
CO 2	✓			✓
CO 3	✓	✓		✓
CO 4		✓		✓
CO 5		✓		✓
CO 6			✓	

Programme	BSc Computer Science				
Course Code	CSC8EJ405				
Course Title	Cyber Forensic				
Type of Course	Elective				
Semester	VIII				
Academic Level	400-499				
Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	4	4	-	-	60
Pre-requisites	1. Understanding concept Computer Hardware, Operating System 2. Knowledge of information security concepts, including confidentiality, integrity, and availability 3. Knowledge of legal and ethical issues surrounding digital evidence collection, preservation, and analysis is crucial for conducting forensic investigations in compliance with applicable laws and regulations.				
Course Summary	This course provides an overview of cyber forensics and cyber laws, focusing on the principles, techniques, and legal considerations involved in investigating cybercrimes, preserving digital evidence, and navigating legal frameworks governing cybersecurity.				

Course Outcomes (CO):

CO	CO Statement	Cognitive Level*	Knowledge Category#	Evaluation Tools used
CO1	To understand the fundamental concepts, principles, and methodologies of cyber forensics	Ap	C	
CO2	To develop skills in acquiring preserving, and analysing digital evidence from various source	Ap	P	
CO3	To learn and understand techniques and tools to investigate cybercrimes, security incidents, and data breaches.	Ap	P	
CO4	Demonstrate proficiency in conducting network, disk, memory, and mobile device forensics examinations.	Ap	P	
CO5	Evaluate ethical, legal, and privacy considerations in cyber forensics investigations and evidence handling.	E	M	
CO6	Apply critical thinking, problem-solving, and decision-making skills to address challenges in cyber forensics and cybersecurity.	Ap	P	

* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C)
# - Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge (M)

Detailed Syllabus

Module	Unit	Content	Hrs (48+12)	Marks
I		**INTRODUCTION TO CYBER FORENSICS**	**10**	**15**
	1	Computer Forensics Fundamentals: What is Computer Forensics? Use of Computer Forensics in Law Enforcement, Computer Forensics Services	2	
	2	Computer Forensics Assistance: Human Recourses/Employment Proceedings, Benefits of professional Forensics Methodology, Steps taken by Computer Forensics Specialists	2	
	3	Computer Forensics Technology: Business Computer Forensic Technology, Military Computer Forensic Technology, Law Enforcement	2	
	4	Vendor and Computer Forensics Services: Types of services provided by vendors, Criteria for selecting a computer forensics vendor, Vendor Engagement and Contracts, Evaluation of vendor capabilities, expertise and reputation	2	
	5	Cyber forensics tools and case studies: Disk Imaging (EnCase, FTK), File Analysis (FileInsight and ExifTool),	2	
II		**COMPUTER FORENSICS EVIDENCE**	**10**	**15**
	6	Computer forensics evidence and capture: Why Collect Evidence, Types of Evidence, The Rules of Evidence, Volatile Evidence,	2	
	7	Data Recovery: Definition, Data Back-up and Recovery, The Role of Back-up in Data Recovery, The Data -Recovery Solution	2	
	8	General Procedure for Data Collection: Collection and Archiving, Methods of Collection	2	
	9	Identification of Data: Timekeeping, Forensic Identification and Analysis of Technical Surveillance Devices, Reconstructing Past Events.	2	
	10	Controlling Contamination: The Chain of Custody, Reconstructing the Attack, The digital crime scene, Investigating Cybercrime, Investigating Web attacks, Investigating network Traffic	2	
III		**FORENSIC ANALYSIS AND VALIDATION**	**14**	
	11	Computer image Verification and Authentication: Special needs of Evidential Authentication,	2	**20**
	12	Computer forensic analysis: Determining what data to collect and analyse, validating forensic data, addressing data-hiding techniques, performing remote acquisitions	2	
	13	Computer forensic validation: Validating forensic data, addressing data-hiding techniques, performing remote acquisitions	2	
	12	Network Forensics: Network forensic overview, Performing live acquisitions, Developing standard procedures for network forensics	2	
	13	Network Forensic Tools: Overview, Wireshark, tcpdump, and NetworkMiner, Network Traffic Analysis Tools	2	

	14	Ethical Hacking: Essential Terminology, Windows Hacking, Malware, Scanning, Cracking.	2	
	15	Tactics of the Military, Tactics of Terrorist and Rogues, Tactics of Private Companies	2	
IV		**CYBER CRIME AND CYBER LAW**	**14**	
	16	Mobile device forensics: Understanding mobile device forensic, understanding acquisition procedures for cell phones and mobile devices.	2	**20**
	17	Cyber Crimes: Types of cybercrimes against individuals and institution, States-various offenses and punishments	2	
	18	Digital Signature: Concepts of public key and private key, Certification Authorities and their role, Creation and authentication of digital signature.	2	
	19	E-contracting: Features of E-contracts, Formation of E-contracts and types	2	
	20	E-governance: E-governance models, E-commerce- salient features and advantages.	2	
	21	Cyber Law: Understanding cyber space, Defining cyber law, Scope and jurisprudence	2	
	22	Indian Cyber Law: Overview of Indian legal system, Introduction to IT Act 2000, Amendment in IT Act.	2	
V		**Open Ended Module- Trends in Software Engineering**		
		1. Case Study. 2. Simulate real-world cyber incidents and develop incident response plans. 3. An activity that emphasizes teamwork, communication, and decision-making under pressure. 4. Work on a comprehensive cyber forensics project that integrates concepts from multiple areas of study. 5. Apply forensic techniques to investigate a real or simulated cyber incident and produce a detailed report.		

Mapping of COs with PSOs and POs:

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PO1	PO2	PO3	PO4	PO5	PO6
CO 1	3	-	-	-	2	1						
CO 2	2	-	-	-	2	1						
CO 3	2	-	-	-	3	2						
CO 4	1	-	-	-	1	1						
CO 5	2	--	-	--	3	1						
CO 6	3		-		2	1						

Correlation Levels:

Level	Correlation
-	Nil
1	Slightly / Low
2	Moderate / Medium
3	Substantial / High

Assessment Rubrics:

- Quiz / Assignment/ Quiz/ Discussion / Seminar
- Midterm Exam
- Programming Assignments (20%)
- Final Exam (70%)

Mapping of COs to Assessment Rubrics:

	Internal Exam	Assignment	Practical Evaluation	End Semester Examinations
CO 1	✓	✓		✓
CO 2	✓	✓		✓
CO 3	✓	✓		✓
CO 4	✓	✓		✓
CO 5	✓	✓		✓
CO 6	✓	✓		✓

References:

1. Computer Forensics: Computer Crime Scene Investigation, 2nd Edition, John R. Vacca, Charles River Media, 2005
2. Cyber Forensics - Concepts and Approaches, Ravi Kumar & B Jain,2006, ICFAI university press
3. Understanding Cryptography: A Textbook for Students and Practitioners, ChristofPaar, Jan Pelzl, 2010, Second Edition, Springer’s.
4. Live Hacking: The Ultimate Guide to Hacking Techniques & Countermeasures for Ethical Hackers & IT Security Experts, Ali Jahangiri, First edition, 2009
5. Computer Forensics: Investigating Network Intrusions and Cyber Crime (Ec-Council Press Series: Computer Forensics), 2010

Programme	B. Sc. Computer Science				
Course Code	CSC8EJ406				
Course Title	Ethical Hacking				
Type of Course	**Elective**				
Semester	VIII				
Academic Level	400-499				
Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	4	4	-	-	60
Pre-requisites	1. Understanding of the fundamental networking and protocols concepts 2. Familiarity with various operating systems, file systems and basic system administration tasks				
Course Summary	This course provides the skills to identify, analyze, and address security vulnerabilities in systems, networks, and web applications. It aims to learn to perform penetration testing, conduct reconnaissance, exploit vulnerabilities, and maintain access ethically and legally.				

Course Outcomes (CO):

CO	CO Statement	Cognitive Level*	Knowledge Category#	Evaluation Tools used
CO 1	Understand the fundamentals of Ethical Hacking	U	C	Instructor-created exams / Quiz
CO 2	Learn the features of Foot Printing and Reconnaissance	Ap	P	Assignment / Seminar presentations/ Exams
CO 3	Apply the System Hacking methods	Ap	P	Seminar Presentation / Group Tutorial Work/ Viva Voce
CO 4	Understand attacks and type of attacks Apply reasoning with ontologies and rules	U	C	Instructor-created exams / Home Assignments
CO 5	Apply varios Penetration Testing methods	Ap	C	Writing assignments/ Exams/ Seminar Presentations
CO 6	Develop theoretical on various types of attach and apply the platforms to explore	Ap	P	Case Study/ Group discussions/ Presentations

* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C)
# - Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge (M)

.

Detailed Syllabus:

Module	Unit	Content	Hrs (48+12)	Marks (70)
I	**Fundamentals of Ethical Hacking**		**15**	**15**
	1	Information security overview, Introduction to Hacking, importance of Security – Elements of Security	2	
	2	Hacking Concepts and Hacker Classes - Phases of Hacking Cycle,	3	
	3	Ethical Hacking Tools - Threat and Threat Sources - Malware and Components of Malware -	4	
	4	Types of Malware, Types of Hackers	3	
	5	Common Hacking Methodologies, Benefits and challenges of Ethical Hacking,	3	
II	**Foot Printing & Reconnaissance**		**12**	**20**
	6	Foot Printing & Reconnaissance: Foot printing concepts, Use of foot printing,	2	
	7	information gathering, Types of foot printing, Website Foot printing	2	
	8	Foot printing through Search Engines, Foot Printing through Social Networking sites	2	
	9	Foot Printing tools, Understanding the information gathering process,	2	
	10	Website Foot printing, WHOIS Foot printing,	2	
	11	Network Scanning, Port scanning,	2	
	12	Tools used for the reconnaissance phase		
III	**System Hacking**		**11**	**20**
	13	Password Cracking - Types of Password Attacks	1	
	14	Password Cracking Tools and vulnerabilities	1	
	15	Identity Theft - Social Engineering and tools	2	
	17	Types of attacks and their common prevention mechanisms :	2	
	17	Keystroke Logging, Denial of Service (DoS /DDoS),	2	
	18	Waterhole attack, brute force, phishing and fake WAP, Session Hijacking	3	
IV	**Penetration Testing**		**10**	**15**

	19	Introduction to Penetration Testing, Types of Penetration Testing-	2	
	20	Phases of PenetrationTesting,	3	
	21	pen testing, type of pen testing.	3	
	22	Tools of Penetration Testing , Test web applications for vulnerabilities	2	
V	**Open Ended Module- Mobile, cloud and IoT Based attacks, Kali Linux**		**12**	
	1	Mobile Platform Attack	3	
	2	Cloud level Attacks and Tools	2	
	3	IoT based attacking Tools	3	
	4	Kali Linux	4	

References

- 1. Michael.T.Simpson, Kent Backman, James.E.Corley, "Hands on Ethical Hacking and Network Defense", Cengage Learning, 2013
- 2. EC-Council, "Ethical Hacking and Countermeasures Attack Phases", Cengage Learning
- 3. James S. Tiller, "The Ethical Hack: A Framework for Business Value Penetration Testing", Auerbach Publications, CRC Press
- Rob Wilson, "Hands-On Ethical Hacking and Network Defense", Cengage Learning, 2022
- Rafay Boloch, ―Ethical Hacking and Penetration Testing Guide‖, CRC Press, 2014

Mapping of COs with PSOs and POs :

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PO1	PO2	PO3	PO4	PO5	PO6
CO 1	2	1	-	-	1	1						
CO 2	3	-	-	-	1	1						
CO 3	1	3	1	1	2	3						
CO 4	1	-	1	1	2	3						
CO 5	1	-	-	-	2	3						
CO 6	1	2	1	1	3	3						

Correlation Levels:

Level	Correlation
-	Nil
1	Slightly / Low
2	Moderate / Medium

3	Substantial / High

Assessment Rubrics:

- Quiz / Assignment/ Quiz/ Discussion / Seminar
- Midterm Exam
- Programming Assignments (20%)
- Final Exam (70%)

Mapping of COs to Assessment Rubrics :

	Internal Exam	Assignm ent	End Semester Examinations
CO 1	✓		✓
CO 2	✓		✓
CO 3	✓		✓
CO 4	✓	✓	✓
CO 5	✓	✓	✓
CO 6	✓	✓	

Programme	B. Sc. Computer Science				
Course Code	CSC8EJ407				
Course Title	Expert Systems and Fuzzy Logic				
Type of Course	Elective				
Semester	VIII				
Academic Level	400 - 499				
Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	4	4	-	-	60
Pre-requisites	1. Familiarity with basic logic and set theories. 2. Understanding the fundamentals of computer science, such as algorithms and data structures, can be beneficial for the implementation aspects of expert systems. 3. A basic understanding of probability and statistics is often required.				
Course Summary	The Fuzzy logic and expert systems course introduce two interconnected fields in artificial intelligence: fuzzy logic and expert systems. Fuzzy logic deals with reasoning under uncertainty and imprecision, while expert systems involve the development of computer-based systems that emulate human expertise in specific domains.				

Course Outcomes (CO):

CO	CO Statement	Cognitive Level*	Knowledge Category#	Evaluation Tools used
CO1	Explain the fundamental concepts of fuzzy set theory and interpret membership functions and linguistic variables.	U	F	Instructor-created exams / Quiz
CO2	Design and implement fuzzy controllers for decision-making. Develop fuzzy inference systems (FIS) for various applications and apply fuzzy clustering techniques for pattern recognition.	U	C	Practical Assignment / Observation of Practical Skills
CO3	Describe the role of expert systems in artificial intelligence and Understand knowledge representation techniques in expert systems.	Ap	P	Practical Assignment / Observation of Practical Skills
CO4	Explain the functioning of inference engines in rule-based systems.	Ap	P	Practical Assignment / Observation of Practical Skills
CO5	Acquire domain knowledge for expert system development.	An	C	Instructor-created exams / Quiz

CO6	Construct a knowledge base and define rules for an expert system and implement validation and refinement techniques for expert systems.	Ap	P	Practical Assignment / Observation of Practical Skills
* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C) # - Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge (M)				

Detailed Syllabus

Module	Unit	Content	Hrs (48+12)	Mark
I		**Introduction to Fuzzy Logic**	**8**	**10**
	1	Overview of Fuzzy Logic	1	
	2	Fuzzy Sets and Membership Functions	2	
	3	Fuzzy Operations (Union, Intersection, Complement)	2	
	4	Basic principles of fuzzy logic. Fuzzification and defuzzification.	2	
	5	Linguistic variables and terms.	1	
II		**Fuzzy Inference Systems (FIS) and Fuzzy Logic Applications**	**12**	**20**
	6	Mamdani FIS-Rule-based systems in fuzzy logic, Rule base and implication methods.	2	
	7	Sugeno FIS-Structure and operation of Sugeno FIS. Comparison with Mamdani FIS.	2	
	8	Basic structure of fuzzy logic controllers (FLCs)	3	
	9	Rule-based systems and fuzzy inference	3	
	10	Applications of fuzzy logic controllers	2	
III		**Introduction to Expert Systems and Rule-Based Systems**	**12**	**20**
	11	Definition and characteristics of expert systems.	2	
	12	Knowledge representation and reasoning.	3	
	13	Expert system components: knowledge base, inference engine, user interface. Examples and applications of expert systems	3	
	14	Rule-based systems and production rules, Forward and backward chaining.	2	
	15	Inference mechanisms in expert systems, Examples of rule-based expert systems.	2	
IV		**Introduction to SCILAB/MATLAB Programming**	**16**	**20**
	16	SCILAB/MATLAB environment and basic navigation, Variables, data types, and basic operations, Script files and running SCILAB/MATLAB code. Introduction to functions and function files.	3	
	17	Introduction to functions and function files, Conditional statements (if, else, elseif), Loop structures (for, while).	2	
	18	Logical operators and relational expressions, Vectorized operations and element-wise operations.	2	
	19	Introduction to arrays, matrices, and vectors, Cell arrays and structures, Indexing and slicing in SCILAB/MATLAB, Working with multidimensional arrays.	2	

	20	Basic file input/output operations, Reading and writing data files (text, CSV, Excel), Data visualization using plotting functions.	2	
	21	Statistical analysis and plotting techniques, Fuzzy logic toolbox in SCILAB/MATLAB.	2	
	22	Expert system development tools in SCILAB/MATLAB, Building expert systems using SCILAB/MATLAB.	3	
V	**Open end**		**12**	
		Case Studies: Real-world applications and their impact. Technological Challenges: Addressing the limitations and exploring new solutions. Future Prospects: Predictions and potential advancements in the field.		

Mapping of COs with PSOs and POs:

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PO1	PO2	PO3	PO4	PO5	PO6
CO 1	1	3	-	-	-	1						
CO 2	1	3	-	-	1	-						
CO 3	1	3	-	-	2	2						
CO 4	1	3	-	-	2	2						
CO 5	2	1	3	1	1	-						
CO 6	2	1	3	2	2	1						

Correlation Levels:

Level	**Correlation**
-	Nil
1	Slightly / Low
2	Moderate / Medium
3	Substantial / High

Assessment Rubrics:

- Quiz / Assignment/ Quiz/ Discussion / Seminar
- Midterm Exam
- Programming Assignments (20%)
- Final Exam (70%)

Mapping of COs to Assessment Rubrics:

	Internal Exam	Assignment	Practical Evaluation	End Semester Examinations
CO 1	✓	✓		✓
CO 2	✓	✓	✓	✓
CO 3	✓	✓	✓	✓
CO 4		✓	✓	✓
CO 5		✓	✓	✓
CO 6	✓	✓	✓	✓

References:

1. "Fuzzy Logic with Engineering Applications" by Timothy J. Ross

2. "Expert Systems: Principles and Programming" by Joseph C. Giarratano and Gary D. Riley

3. "Fuzzy Sets and Fuzzy Logic: Theory and Applications" by George J. Klir and Bo Yuan

4. "Expert Systems: Principles and Case Studies" by Efraim Turban, Jay E. Aronson, and Ting-Peng Liang

5. “Introduction to Fuzzy Logic using MATLAB” by S.N. Sivanandam, S. Sumathi, and S. N. Deepa.

6. Nagar, S. (2017). Introduction to Scilab: For Engineers and Scientists. Apress.

Detailed Syllabus of Minor Courses

Programme	B. Sc. Computer Science				
Course Code	CSC1MN101				
Course Title	Exploring Computer Basics & Computational Thinking				
Type of Course	**Minor**				
Semester	I				
Academic Level	100-199				
Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	4	3	-	2	75
Pre-requisites	1. Foundation on Mathematics at Plus Two level 2. Foundation on Basic Science at Plus Two Level				
Course Summary					

Course Outcomes (CO):

CO	CO Statement	Cognitive Level*	Knowledge Category#	Evaluation Tools used
CO1	Understanding of computer hardware, software, and basic operation principles	U	C	Exams/ Assignments/ Quizzes/ Seminars/ Practical
CO2	Understand and identify computer hardware components	U, Ap	C	Exam/ Assignments/ Quizzes/ Seminars/ v
CO3	Understand how data is represented and manipulated within a computer system.	U	C	Exam/ Assignments/ Quizzes/ Seminars
CO4	Understand the basics of computer languages, operating systems, and their comparison	U	C	Exam/ Assignments/ Quizzes/ Seminars

CO5	Learn to design and implement algorithms to solve simple computational problems.	U	P	Exam/ Assignments/ Quizzes/ Seminars/ / Practical
CO5	Develop computational thinking skills essential for problem-solving in various domains	Ap	P	Exam/ Assignments/ Quizzes/ Seminars/ / Practical
* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C) # - Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge (M)				

Detailed Syllabus:

Module	Unit	Content	Hrs	Marks
I	**History, Evolution of Computers, and Number System**		**8**	**15**
	1	Introduction to Computers, Characteristics of Computers	1	
	2	Generations of Computers	1	
	3	Classification of Computers: Super Computers, Main Frame Computers, Mini Computers, Micro Computers	1	
	4	Number Systems (Binary, Decimal, Octal, Hexadecimal) and Conversion	3	
	5	Computer Codes: BCD Code, Excess 3 Code, ASCII Code, Unicode, Gray Code	2	
II	**Basic Computer Organization and Concept of Hardware**		**14**	**20**
	6	● Basic Computer Organization: Input Unit, Storage Unit, Processing Unit, Control Unit, Output Unit	1	
	7	● Concept of hardware	1	
	8	● CPU: Arithmetic and Logic Unit, Control unit	1	
	9	● Memory: Primary Memory, Secondary Memory, Access Time, Storage Capacity-bit, byte, nibble, volatile memory	2	
	10	● Memory hierarchy: Register memory, Cache memory, RAM (Static, Dynamic), ROM(Masked ROM, PROM and EPROM), Secondary storage devices (Magnetic tape, Hard disk, SSD and CD drive)	5	

		● Inside CPU: SMPS, Motherboard, Processor, Storage Devices (HDD, SSD, RAM, ROM).	1	
	11	● Motherboard Components: Processor Slot, Cooling Fan, RAM, Expansion Slots (PCIe), Input/Output Ports, Chipset, BIOS/UEFI Chip, SATA/NVMe Slots, Network Interface, Ports- Ethernet, VGA port, HDMI port, USB port.	3	
III	**Input-Output Devices, Concept of Software**		**12**	**15**
	12	● Input Devices: keyboard, pointing devices (mouse, touchpad), Video digitizer, remote control, joystick, scanner, digital camera, microphone, sensor	2	
	13	● Output Devices: monitor, printer (laser, inkjet, dot-matrix), plotter, speaker, control devices (lights, buzzers, robotic arms, and motors)	2	
	14	● Types of Software: System Software vs. Application Software, Proprietary Vs Open Source	2	
	15	Operating Systems: Functions, types of OS (batch, multiprogramming, time-sharing, real-time, and distributed)	2	
	16	Programming Languages (Machine, assembly & High level),	2	
	17	language Translators (Assembler, Interpreter and Compiler)	2	
1V	**Problem-solving and logical Thinking**		**11**	**20**
	18	Introduction to Problem Solving: Understanding the importance of problem-solving in computer science, Identifying and defining problems in a computational context.	2	
	19	Algorithm and its characteristics	1	
	20	Algorithm Development: Steps involved in designing algorithms, Pseudocode is an intermediate step in algorithm development.	2	
	21	Flowchart Basics: Introduction to flowcharts as a visual representation of algorithms, Understanding flowchart symbols and their meanings	2	
	22	Drawing simple flowcharts	4	
V	**Hands-on Data Structures:** **Practical Applications, Case Study and Course Project**		**30**	

	1	**Hardware:** 1. Identify the given motherboard components. 2. Identify and describe various ports and connectors on the motherboard.	5	
	2	**Software:** 1. Check the hardware compatibility and Install an operating system on a given computer. 2. Install any device driver on a given computer system to communicate with peripheral devices like Printers, Scanner, etc..	5	
	3	**Design Algorithm and visualize it using RAPTOR software** Problem 1: Calculate the Sum of Two Numbers Problem 2: Find the Larger of Two Numbers Problem 3: Check if a Number is Even or Odd Problem 4: Calculate the Factorial of a Number Problem 5: Temperature Conversion Problem 6: Simple Interest Calculation Problem 7: Calculate the Sum of Digits in a Number Problem 8: Check if a Number is Positive, Negative, or Zero Problem 9: Determine if a Triangle is Equilateral, Isosceles, or Scalene Problem 10: Check if a Number is Prime or Composite	20	

Reference Books:

1. Brookshear, J. Glenn. Computer Science: An Overview. 13th ed., Pearson, 2014.

2. Norton, Peter. Introduction to Computers. 7th ed., McGraw-Hill, 2016.

3. Patterson, David A. and John L. Hennessy. Computer Organization and Design: The Hardware/Software Interface. 5th ed., Morgan Kaufmann, 2013.

4. Sedgewick, Robert, and Kevin Wayne. Algorithms. 4th ed., Addison-Wesley Professional, 2011.

5. Knuth, Donald E. The Art of Computer Programming, Volumes 1-4A Boxed Set. Addison-Wesley Professional, 2011.

6. Grover, Aditya Bhargava. Grokking Algorithms: An Illustrated Guide for Programmers and Other Curious People. Manning Publications, 2016.

Mapping of COs with PSOs and POs :

	PSO1	PSO2	PSO3	PSO4	PS O5	PSO 6	PO1	PO2	PO3	PO4	PO5	PO6
CO 1	1	2	-	-	-	-	-					

CO 2	1	2	-	-	-	-	-					
CO 3	1	2	-	-	-	-	-					
CO 4	-	2	2	2	-	-	-					
CO 5	-	2	2	2	-	-	-					
CO 6	-	2	2	2	-	1	-					

Correlation Levels:

Level	**Correlation**
-	Nil
1	Slightly / Low
2	Moderate / Medium
3	Substantial / High

Assessment Rubrics:

- Quiz / Assignment/ Quiz/ Discussion / Seminar
- Midterm Exam
- Programming Assignments (20%)
- Final Exam (70%)

Mapping of COs to Assessment Rubrics:

	Internal Exam	Assignment	Project Evaluation	End Semester Examinations
CO 1		✓		✓
CO 2	✓	✓		✓
CO 3	✓	✓		✓
CO 4	✓	✓		✓
CO 5	✓	✓		✓

CO 6	✓	✓	✓	✓

Programme	B. Sc. Computer Science				
Course Code	CSC2MN101				
Course Title	Foundations of C Programming				
Type of Course	**Minor**				
Semester	II				
Academic Level	100-199				
Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	4	3	-	2	75
Pre-requisites	1. Basic Computer Literacy 2. Basic Problem-Solving Skills				
Course Summary	This course teaches the basics of programming using the C language. C is a powerful and widely used programming language known for its efficiency and flexibility. Through this course, students will learn how to write, understand, and debug C code to solve various problems and build simple applications.				

Course Outcomes (CO):

CO	CO Statement	Cognitive Level*	Knowledge Category#	Evaluation Tools used
CO1	Demonstrate a solid understanding of fundamental programming concepts	An	P	Instructor-created lab exams / Quiz
CO2	Develop effective problem-solving skills by applying algorithmic thinking and logical reasoning.	An	P	Problem-solving assessments
CO3	Gain proficiency in writing, compiling, debugging, and executing C programs to implement algorithms, solve	Ap	P	Modeling Assignments

	problems, and create applications.			
CO4	Learn techniques to write efficient and optimized C code, including memory management, algorithm design, and performance tuning, to produce high-quality and scalable software solutions.	Ap	P	Modeling Assignments/ / Case studies
CO5	Understand and apply software development practices such as modular programming, code documentation, and debugging techniques to write maintainable and robust C programs.	Ap	P	Modeling Assignments/ / Case studies
CO6	Develop critical thinking skills by analyzing and evaluating C code, identifying errors and inefficiencies, and proposing solutions to improve code quality and performance.	Ap	P	Hands-on exercises
* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C) # - Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge (M)				

Detailed Syllabus:

Module	Unit	Content	Hrs	Marks
I	**Problem-solving and logical Thinking**		**10**	**15**
	1	Overview of computational thinking concepts. Definition of algorithm and its characteristics.Understanding the importance of algorithms in problem-solving	2	
	2	Algorithm Development: Steps involved in designing algorithms	2	
	3	Pseudocode as an intermediate step in algorithm development.	1	

	4	Understanding flowchart symbols and their meanings.Learning to represent algorithms using flowcharts.	2	
	5	Raptor as a precursor to text-based programming languages	2	
	6	Drawing simple flowcharts	1	
II	**Introduction to C**		**10**	**20**
	7	Structure of C program	2	
	8	C Character Set, Keywords, Identifiers	1	
	9	Data Types, Variables, Declarations, Symbolic Constants	2	
	10	Operators:Arithmetic, Logical, Relational & Equality, and Unary, Operator Precedence and Associativity	2	
	11	Library Functions, Comments	1	
	12	I/O functions- Formatted scanf() & printf().	2	
III	**Control Statements, Arrays & Strings**		**14**	**20**
	13	Selection Statements:if, if-else, switch	3	
	14	iteration: while, do while, for	4	
	15	Arrays: One dimensional and Two Dimensional(introduction only)	3	
	16	Strings: Basic string handling functions	2	
	17	Structure:Definition, Processing-period Operator, Union(Concepts only)	2	
1V	**User-defined Functions**		**11**	**15**
	18	Definition of function, Advantages, Understanding function prototypes, and declarations	3	
	19	Introduction to function definitions and function calls	3	
	20	Exploring function parameters: actual and formal parameters	2	
	21	Recursion	2	
	22	Pointers-declarations(Basic concept only)	1	
V	**Hands-on C:** **Practical Applications, Case Study and Course Project**		**30**	

	1	Write a C program using Variables and Data Types Write a C program using Arithmetic Operations Write a C program using Loops Write a C program using Arrays Write a C program using Functions Write a C program using Strings	20	
	2	Case study: 1. Library Management System: Develop a program to manage a library's collection of books. Implement functions for adding, removing, and searching for books. 2. Ticket Booking System: Design a program to manage ticket bookings for a cinema or theater.	5	
	3	Capstone/Course Project: Design a real-time project in C	5	

Reference:

1. Balagurusamy, E. Programming in ANSI C. Tata McGraw-Hill Education, 2019.
2. King, K. N. C Programming: A Modern Approach. 2nd ed., W. W. Norton & Company, 2008.
3. Kernighan, Brian W., and Dennis M. Ritchie. The C Programming Language. 2nd ed., Prentice Hall, 1988.
4. Prata, Stephen. C Primer Plus. 6th ed., Addison-Wesley, 2013.
5. Perry, Greg. Absolute Beginner's Guide to C. 3rd ed., Que Publishing, 2014.
6. Oualline, Steve. Practical C Programming. 3rd ed., O'Reilly Media, 1997.
7. Hanly, Jeri R., and Elliot B. Koffman. Problem-Solving and Program Design in C. 8th ed., Pearson, 2016.
8. Gottfried, Byron S. Programming with C. 2nd ed., McGraw-Hill, 1996.
9. Holmes, Dan. C in a Nutshell. 2nd ed., O'Reilly Media, 2015.

Mapping of COs with PSOs and POs :

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PO1	PO2	PO3	PO4	PO5	PO6
CO 1	3	1	-	-	-	1						
CO 2	1	-	2	-	-	-						
CO 3	-	-	2	-	-	-						
CO 4	-	1	3	3	-	3						
CO 5	-	2	3	3	-	3						
CO 6	-	-	-	-	-	3						

Correlation Levels:

Level	**Correlation**
-	Nil
1	Slightly / Low
2	Moderate / Medium
3	Substantial / High

Assessment Rubrics:

- Quiz / Assignment/ Quiz/ Discussion / Seminar
- Midterm Exam
- Programming Assignments (20%)
- Final Exam (70%)

Mapping of COs to Assessment Rubrics :

	Internal Exam	Assignment	Project Evaluation	End Semester Examinations
CO 1		✓		✓
CO 2	✓	✓		✓
CO 3		✓		✓
CO 4	✓			✓
CO 5	✓		✓	✓
CO 6	✓		✓	✓

Programme	B. Sc. Computer Science				
Course Code	CSC3MN201				
Course Title	Python Programming				
Type of Course	**Minor**				
Semester	III				
Academic Level	200-299				
Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	4	3	-	2	75
Pre-requisites	Have an understanding of algorithms and flowcharts				
Course Summary	This course explores the versatility of Python language in programming and teaches the application of various data structures using Python.				

Course Outcomes (CO):

CO	**CO Statement**	**Cognitive Level***	**Knowledge**	**Evaluation Tools used**
CO1	Understand the basic concepts of Python programming	U	C	Instructor- created exams / Quiz
CO2	Apply problem-solving skills using different control structures and loops	Ap	P	Coding Assignments/ Code reading and review
CO3	Design simple Python programs to solve basic computational problems and acquire knowledge of Python's error-handling mechanisms to effectively debug programs	Ap	P	Coding Assignments/ Exams
CO4	Analyze the various data structures and operations on it using Python	An	P	Instructor-created exams / Case studies
CO5	Apply modular programming using functions	U	C	Instructor- created exams / Quiz

CO6	Identify the necessary Python packages in the domain and create simple programs with it	U, Ap	C, P	Coding
* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C) # - Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge (M)				

Detailed Syllabus:

Module	Unit	Content	Hrs	Marks
I	**Introduction to Python**		**12**	15
	1	Features of Python, Different methods to run Python,Python IDE	2	
	2	Comments, Indentation, Identifiers, Keywords, Variables	2	
	3	Standard Data Types	2	
	4	Input Output Functions, Import Functions, range function	1	
	5	Operators and Operands, Precedence of Operators, Associativity	2	
	6	Type Conversion, Multiple Assignment	1	
	7	Expressions and Statements, Evaluation of Expressions	1	
	8	Boolean Expressions	1	
II	**Control Structures**		**12**	20
	9	Decision Making- if statement, if…else statement, if…elif…else statement, Nested if statement	5	
	10	Loops - for loop, for loop with else, while loop, while loop with else, Nested Loops	5	
	11	Using indentation in Python to define code blocks	1	
	12	Control Statements- break, continue, pass	1	

III	**Data Structures in Python**		**12**	15
	13	Working with strings and string manipulation	3	
	14	List - creating list, accessing, updating and deleting elements from a list	2	
	15	Basic list operations	1	
	16	Tuple- creating and accessing tuples in python	2	
	17	Basic tuple operations	1	
	18	Dictionary, built in methods to create, access, and modify key-value pairs	2	
	19	Set and basic operations on a set	1	
	Functions		**9**	
IV	20	Built-in functions - mathematical functions, date time functions, random numbers	1	20
	21	Writing user defined functions - function definition, function call, flow of execution, parameters and arguments, return statement	6	
	22	Recursion. Introduction to basic Python libraries (e.g., math, random)	2	
	Hands-on Data Structures: **Practical Applications, Case Study and Course Project**		**30**	
Design programs from the concepts listed below. Select the topics and programs suited for your domain				
V	1	Programs to: • Run instructions in Interactive interpreter and as Python Script • Perform calculations involving integers and floating point numbers using Python arithmetic operators **Data Structures in Python** • String - Create a string , Indexing / Looping / Slicing		

		• Lists - Create a list , Indexing /Looping / Slicing , Adding items / Modifying items / Removing items • Tuples - Create a tuple , Indexing / Looping / Slicing / Adding items to a tuple • Dictionary - Create a dictionary and access values with key / Adding a key- value pair / Adding to an empty dictionary /Modifying values in a dictionary / Removing key-value pair **Function** • Call functions residing in the math module • Define a function for later use • Pass one or more values into a function • Return one or more results from a function		
		Case study: • Create a Todo List Manager where Users should be able to add, remove, and view tasks • Create Student Grade Tracker: Allow users to add students, add grades for subjects, and calculate average grades.		

Reference Books:

1. Jose, Jeeva. Taming Python By Programming. Khanna Book Publishing, 2017. Print.
2. Downey, Allen. Think Python. Green Tea Press, 2nd ed. 2009

Mapping of COs with PSOs and POs :

	PSO1	PSO 2	PSO 3	PSO4	PSO5	PSO6	PO 1	PO2	PO3	PO4	PO5	PO6
CO 1	-	1	2	3	1	1						
CO 2	-	1	2	3	1	1						
CO	-	2	2	3	1	1						

3												
CO 4	1	1	-	-	1	-						
CO 5	1	1	2	2	1	-						
CO 6	-	1	2	2	2	1						

Correlation levels

Level	**Correlation**
-	Nil
1	Slightly / Low
2	Moderate / Medium
3	Substantial / High

Assessment Rubrics:

- Quiz / Assignment/ Quiz/ Discussion / Seminar
- Midterm Exam
- Programming Assignments (20%)
- Final Exam (70%)

Mapping of COs to Assessment Rubrics :

	Internal Exam	Assignment	Project Evaluation	End Semester Examinations
CO 1	✓			✓
CO 2	✓	✓	✓	✓
CO 3	✓		✓	✓
CO 4	✓	✓	✓	✓
CO 5	✓			✓
CO 6	✓			✓

Programme	B. Sc. Computer Science				
Course Code	CSC1MN102				
Course Title	Python Programming				
Type of Course	**Minor**				
Semester	I				
Academic Level	100-199				
Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	4	3	-	2	75
Pre-requisites	Have an understanding about algorithms and flowchart				
Course Summary	This course explores the versatility of Python language in programming and teaches the application of various data structures using Python.				

Course Outcomes (CO):

CO	**CO Statement**	**Cognitiv e Level***	**Knowledg e**	**Evaluation Tools used**
CO1	Understand the basic concepts of Python programming	U	C	Instructor- created exams / Quiz
CO2	Apply problem- solving skills using different control structures and loops	Ap	P	Coding Assignments/ Code reading and review
CO3	Design simple Python programs to solve basic computational problems and acquire knowledge of Python's error handling mechanisms to effectively debug	Ap	P	Coding Assignments/ Exams

	programs			
CO4	Analyse the various data structures and operations on it using Python	An	P	Instructor-created exams / Case studies
CO5	Apply modular programming using functions	U	C	Instructor- created exams / Quiz
CO6	Identify the necessary Python packages in the domain and create simple programs with it	U, Ap	C, P	Coding
* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C) # - Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge (M)				

Detailed Syllabus:

Module	**Unit**	**Content**	**Hrs**	**Marks**
I	**Introduction to Python**		**12**	20
	1	Features of Python, Different methods to run Python,Python IDE	2	
	2	Comments, Indentation, Identifiers, Keywords, Variables	2	
	3	Standard Data Types	2	
	4	Input Output Functions, Import Functions, range function	1	
	5	Operators and Operands, Precedence of Operators, Associativity	2	
	6	Type Conversion, Multiple Assignment	1	

	7	Expressions and Statements, Evaluation of Expressions	1	
	8	Boolean Expressions	1	
II	**Control Structures**		**12**	20
	9	Decision Making- if statement, if…else statement, if…elif…else statement, Nested if statement	5	
	10	Loops - for loop, for loop with else, while loop, while loop with else, Nested Loops	5	
	11	Using indentation in Python to define code blocks	1	
	12	Control Statements- break, continue, pass	1	
III	**Data Structures in Python**		**12**	20
	13	Working with strings and string manipulation	3	
	14	List - creating list, accessing, updating and deleting elements from a list	2	
	15	Basic list operations	1	
	16	Tuple- creating and accessing tuples in python	2	
	17	Basic tuple operations	1	
	18	Dictionary, built in methods to create, access, and modify key-value pairs	2	
	19	Set and basic operations on a set	1	
	Functions		**9**	15
IV	20	Built-in functions - mathematical functions, date time functions, random numbers	1	
	21	Writing user defined functions - function definition, function call, flow of execution, parameters and arguments, return statement	6	
	22	Recursion. Introduction to basic Python libraries (e.g., math, random)	2	

<table>
<tr><td></td><td colspan="2">Hands-on Data Structures:

Practical Applications, Case Study and Course Project</td><td>30</td><td></td></tr>
<tr><td colspan="5">Design programs from the concepts listed below. Select the topics and programs suited for your domain</td></tr>
<tr><td>V</td><td>1</td><td>Programs to:
• Run instructions in Interactive interpreter and as Python Script
• Perform calculations involving integers and floating point numbers using Python arithmetic operators

Data Structures in Python
• String - Create a string , Indexing / Looping / Slicing
• Lists - Create a list , Indexing /Looping
/ Slicing , Adding items / Modifying items / Removing items
• Tuples - Create a tuple , Indexing / Looping / Slicing / Adding items to a tuple
• Dictionary - Create a dictionary and access values with key / Adding a key- value pair / Adding to an empty dictionary /Modifying values in a dictionary / Removing key-value pair

Function
• Call functions residing in the math module
• Define a function for later use
• Pass one or more values into a function
• Return one or more results from a function</td><td></td><td></td></tr>
<tr><td></td><td></td><td>Case study:
• Create a Todo List Manager where Users should be able to add, remove, and view tasks
• Create Student Grade Tracker: Allow users to add students, add grades for subjects, and calculate average grades.</td><td></td><td></td></tr>
</table>

Mapping of COs with PSOs and POs :

	PSO 1	PSO 2	PSO 3	PSO4	PSO5	PSO6	PO 1	PO2	PO3	PO4	PO5	PO6
CO 1	-	1	2	3	1	1						
CO 2	-	1	2	3	1	1						
CO 3	-	2	2	3	1	1						
CO 4	1	1	-	-	1	-						
CO 5	1	1	2	2	1	-						
CO 6	-	1	2	2	2	1						

Correlation Levels:

Level	**Correlation**
-	Nil
1	Slightly / Low
2	Moderate / Medium
3	Substantial / High

Assessment Rubrics:

- Quiz / Assignment/ Quiz/ Discussion / Seminar
- Midterm Exam
- Programming Assignments (20%)
- Final Exam (70%)

Mapping of COs to Assessment Rubrics :

	Internal Exam	Assignment	Project Evaluation	End Semester Examinations
CO 1	✓			✓
CO 2	✓	✓	✓	✓
CO 3	✓		✓	✓
CO 4	✓	✓	✓	✓
CO 5	✓			✓
CO 6	✓			✓

Reference Books:

1. Jose, Jeeva. Taming Python By Programming. Khanna Book Publishing, 2017. Print.

2. Downey, Allen. Think Python. Green Tea Press, 2nd ed. 2009

Programme	B. Sc. Computer Science				
Course Code	CSC2MN102				
Course Title	Introduction to Data Science				
Type of Course	**Minor**				
Semester	II				
Academic Level	100-199				
Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	4	3	-	2	75
Pre-requisites	1. Python Programming 2. Linear Algebra				
Course Summary	This course provides a comprehensive overview of data science, covering the various types of data and their applications. The students will acquire a deep understanding of exploratory data analysis along with hands-on implementation skills. . The curriculum introduces both supervised and unsupervised and techniques of Machine learning. Additionally, the data pre-processing techniques are introduced Overall, the course provides a comprehensive understanding of the fundamental data science principles, guiding students through the data science process and illustrating practical applications.				

Course Outcomes (CO):

CO	CO Statement	Cognitive Level*	Knowledge Category#	Evaluation Tools used
CO1	Understand the types of data and the applications of data science	U	C	Instructor-created exams / Quiz
CO2	Analyse the irregularities present in the data and perform data cleaning	An	C	Problem-solving assessments

CO3	Implement various visualisation techniques on different data types	Ap	P	Modelling Assignments
CO4	Create prediction models using supervised techniques	Ap	P	Modelling Assignments/ / Case studies
CO5	Assess the similarity among the data using unsupervised techniques.	Ap	P	Modelling Assignments/ / Case studies
CO5	Gain insights on advanced data pre-processing techniques	U	C	Instructor-created exams / Quiz
* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C) # - Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge (M)				

Detailed Syllabus:

Module	**Unit**	**Content**	**Hrs**	**Marks** **(70)**
I	**Introduction to Data Science**		**10**	**10**
	1	Introduction to Data: Types of Data – Structured Data, Semi-Structured Data, Unstructured Data and Data Streams, Statistical Data Types - Quantitative Data (Ratio and Interval Scale) and Qualitative Data (Nominal and ordinal)	2	
	2	Basic Methods of Data Analysis- Descriptive Data Analysis, Diagnostic Data Analysis or Exploratory Data Analysis, Inferential Data Analysis and Predictive Analysis.	1	
	3	Inferential Statistics: Statistical Inference, Population and Sample, Statistical Modeling, Probability Distributions – Normal, Uniform	3	
	4	Introduction to Data Science: Big Data and Data Science , Data Science Process	2	
	5	Applications of Data Science , Issues and challenges in Data Science	2	

II	**Exploratory Data Analysis**		**14**	**10**
	6	Exploratory Data Analysis (EDA) : Types of EDA - Univariate non-graphical, Multivariate non- graphical, Univariate graphical, and, Multivariate graphical. Methods of EDA – Descriptive Statistics and Data Visualization	5	
	7	Descriptive Statistics - Measures of Central Tendencies, Dispersion, Skewness and Kurtosis.	5	
	8	Data Visualization - Histograms , Box plots , Quantile-Quantile plots Scatter plots , Heat map, Bubble chart , Bar chart, Distribution plot , Pair plot , Line graph , Pie chart, Area chart	4	
III	**Data Preparation for Analysis**		**6**	**15**
	9	Data Cleaning: Handling Missing and Noisy Data, Removing outliers	2	
	10	Data Integration	1	
	11	Data Transformation: Standardization, Normalization	2	
	12	Data Reduction: Dimensionality Reduction - Principal Component Analysis	1	
1V	**Introduction to Machine Learning**		**15**	**15**
	13	Machine Learning Algorithms : Supervised Learning– Classification, Regression, Unsupervised Learning – Clustering, Dimensionality Reduction , Reinforcement Learning	3	
	14	Test /Train Split, Model Training, Bias and Variance, Overfitting and Underfitting	3	
	15	Evaluation	2	
	16	Linear Regression	1	
	17	k-Nearest Neighbors (k-NN)	1	
	18	k-means Clustering	1	
	19	Naive Bayes	1	
	20	Application of Naive Bayes - Spam Filtering	1	
	21	Singular Value Decomposition	1	
	22	Applications of Supervised, Unsupervised and Reinforcement	1	

		Learning		
V	**Hands-on Data Structures:** **Practical Applications, Case Study and Course Project**		**30**	**20**
	1	Implementation of the concepts or the algorithms learned [Binary Classification, Linear Regression, k-NN, k-means clustering, Spam Filtering]	15	
	2	Case study: Perform exploratory data analysis on a real world dataset using Python. Using appropriate Python packages parse, clean and visualize the data .	5	
	3	Capstone/Course Project: Perform an end-to-end project of the data science process.		

Mapping of COs with PSOs and POs :

	PSO1	PSO2	PSO3	PSO4	PSO 5	PSO 6	PO1	PO2	PO3	PO4	PO5	PO6
CO 1	3	-	-	-	-	1						
CO 2	1	-	2	-	-	-						
CO 3	-	-	2	-	-	-						
CO 4	-	2	3	3	-	1						
CO 5	-	2	3	3	-	1						
CO 6	-	-	-	-	-	2						

Correlation Levels:

Level	**Correlation**
-	Nil

1	Slightly / Low
2	Moderate / Medium
3	Substantial / High

Assessment Rubrics:

- Quiz / Assignment/ Quiz/ Discussion / Seminar
- Midterm Exam
- Programming Assignments (20%)
- Final Exam (70%)

Mapping of COs to Assessment Rubrics :

	Internal Exam	Assignment	Project Evaluation	End Semester Examinations
CO 1		✓		✓
CO 2	✓	✓		✓
CO 3		✓		✓
CO 4	✓			✓
CO 5	✓		✓	✓
CO 6	✓		✓	✓

References

1. O'Neil, Cathy, and Rachel Schutt. *Doing data science: Straight talk from the frontline*. " O'Reilly Media, Inc.", 2013.
2. Han, Jiawei, et al. Data Mining: Concepts and Techniques. Netherlands, Elsevier Science, 2011.
3. Shah, Chirag. A Hands-On Introduction to Data Science. United Kingdom, Cambridge University Press, 2020.
4. Chopra, Rohan, et al. Data Science with Python: Combine Python with Machine Learning Principles to Discover Hidden Patterns in Raw Data. United Kingdom, Packt Publishing, 2019.

Programme	B. Sc. Computer Science				
Course Code	CSC3MN202				
Course Title	Introduction to AI and Machine Learning				
Type of Course	**Minor**				
Semester	III				
Academic Level	200 - 299				
Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	4	3	-	2	75
Pre-requisites	1. Fundamental Mathematics Concepts: Sets 2. Fundamentals of Python Programming				
Course Summary	This course provides an introduction to the ideas, techniques, and applications of artificial intelligence (AI) is given in this course. The fundamentals of knowledge representation, machine learning, and problem solving will be taught to the students.				

Course Outcomes (CO):

CO	CO Statement	Cognitive Level*	Knowledge Category#	Evaluation Tools used
CO1	Explain the basic concepts of Artificial Intelligence	U	C	Instructor-created exams / Quiz
CO2	Master Problem-Solving Techniques. Apply a problem solving technique to solve standard AI problems	Ap	P	Practical Assignment / Observation of Practical Skills
CO3	Master various packages required to develop AI and machine learning applications	Ap	C	Seminar Presentation / Group Tutorial Work/ Viva Voce
CO4	Understand few AI tools and an insight to	U	C	Instructor-

	Machine learning, Deep learning concepts			created exams / Home Assignments
CO5	Implement and analyse Machine learning algorithms to solve practical problems.	Ap	P	Writing assignments/ Exams/ Practical
CO6	Apply Concepts in Real-World Projects	Ap	P	Case Study/ mini Project
* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C) # - Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge (M)				

Detailed Syllabus:

Module	**Unit**	**Content**	**Hrs**	**Marks(70)**
I	**Introduction to Artificial Intelligence & Problem Solving**		**15**	**12**
	1	Introduction to AI – Evolution of AI, AI problems, AI Techniques, AI Applications	4	
	2	Various AI Domains (Introduction only)	2	
	3	Problem Solving Techniques - Search Algorithms, Knowledge representation and reasoning (Concepts only)	3	
	4	Problem Solving Techniques - constraint satisfaction problems, Game playing (Concepts only)	3	
	5	Problem Solving Techniques - Machine learning, Simulated Annealing (Concepts only)	3	
II	**Introduction to Neural Networks**		**8**	**12**
	6	Introduction to Artificial Neural Network	2	
	7	Understanding Brain & Perceptron Model	2	
	8	Single Layer Perceptron Model & Learning in Single layer Perceptron Model	2	
	9	Multi-Layer Perceptron Model & Learning in Multi-layer Perceptron Model	2	

III	**Python Packages for AI**		**15**	**10**
	10	Pandas	3	
	11	MatplotLib	3	
	12	Keras	3	
	13	Scikit-learn:	3	
IV		**Machine Learning Fundamentals**	**7**	**16**
	15	Introduction to Machine learning-	1	
	16	Applications of Machine Learning	1	
	17	Supervised machine learning- Classification, regression (concepts only)	2	
	18	Unsupervised machine learning	1	
	19	clustering, Dimensionality Reduction (concepts only)	1	
	20	Basics of reinforcement learning	1	
	21	Definition and history of deep learning	1	
	22	Key differences between traditional machine learning and deep learning	1	
V	**Hands-on Artificial Intelligence & Machine Learning using Python:** **Practical Applications, Case Study and Course Project**		**30**	**20**
	1	**1. Neural Network** Building a single layer perceptron using Keras **2. Multi-layer Neural Network** Setting up a multi-layer perceptron model **4. Supervised machine learning** Linear regression Decision tree **5. Unsupervised machine learning** K means clustering	20	

		PCA **6. Feature Engineering** Feature selection from a dataset		
	2	Case study – AI tools / Use of AI in any movie	3	
	3	**Implementation of Comparison of any two machine learning algorithms on a dataset**	7	

References

- Elaine Rich, Kevin Knight, Shivsankar B Nair, "Artificial Intelligence", Third Edition, Tata McGraw Hill Publisher
- Tom M. Mitchell, Machine Learning, McGraw-Hill, 1st Ed.
- Ethem Alpaydin, Introduction to Machine Learning- 3rd Edition, PHI.

Mapping of COs with PSOs and POs :

	PSO1	PSO2	PSO3	PSO4	PS O5	PSO6	PO1	PO2	PO3	PO4	PO5	PO6
CO 1	2	1	1	1	2	1						
CO 2	2	1	2	3	2	2						
CO 3	2	1	2	3	2	3						
CO 4	3	-	1	2	-	-						
CO 5	1	-	2	3	3	3						
CO 6	2	-	3	3	3	3						

Correlation Levels:

Level	**Correlation**
-	Nil
1	Slightly / Low
2	Moderate / Medium
3	Substantial / High

Assessment Rubrics:

- Quiz / Assignment/ Quiz/ Discussion / Seminar
- Midterm Exam
- Programming Assignments (20%)
- Final Exam (70%)

Mapping of COs to Assessment Rubrics :

	Internal Exam	Assignment	Practical Evaluation	End Semester Examinations
CO 1	✓	✓		✓
CO 2	✓	✓		✓
CO 3	✓	✓		✓
CO 4	✓	✓		✓
CO 5	✓	✓	✓	✓
CO 6	✓	✓	✓	

Programme	B. Sc. Computer Science				
Course Code	CSC1MN103				
Course Title	Data analysis using Spreadsheet				
Type of Course	**Minor**				
Semester	I				
Academic Level	100-199				
Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	4	3	-	2	75
Pre-requisites	1. Basic mathematics knowledge 2. Basic computer knowledge				
Course Summary	This syllabus aims to cover a broad spectrum of Excel skills, catering to participants with varying levels of expertise.				

Course Outcomes (CO):

CO	**CO Statement**	**Cognitive Level***	**Knowledge Category#**	**Evaluation Tools used**
CO1	Demonstrate the ability to enter data accurately and efficiently into Excel worksheets	Ap	P	Instructor-created exams / Quiz
CO2	Use of Excel formulas, including basic arithmetic operations, application of common functions calculations in spreadsheets.	Ap	C	Problem-solving assessments
CO3	Use Excel for data analysis, including sorting, filtering, and the creation of Tables.	Ap	P	Instructor-created exams / Quiz
CO4	Demonstrate proficiency	Ap	P	Instructor-created

	in utilizing advanced Excel functions			exams / Quiz
CO5	Demonstrate collaboration skills and the ability represent real world data and create reports	Ap	P	Modelling Assignments/ / Case studies
* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C) # - Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge (M)				

Detailed Syllabus:

Module	Unit	Content	Hrs	Marks
I	**Introduction to Spreadsheets**		**12**	**18**
	1	Overview - Overview of spreadsheet software (Microsoft Excel, Google Sheets) and their application	2	
	2	Excel Interface and Navigation-Ribbon,Row ,Column, Cell Worksheet,Workbook,Cell Address,Data range,Formula, Chart)	2	
	3	Basic navigation techniques within the workbook	2	
	4	Creating and Saving Workbooks - Creating a new workbook and saving it , Different file formats and when to use them	2	
	5	Inserting or deleting rows or columns	2	
	6	Basic Cell Formatting - Formatting text, numbers, and dates,	2	
II	Data Management		**11**	**18**
	7	Find and select -Find,Replace,Go To,Go To Special	2	

	8	Cell Referencing-Relative, Absolute and Mixed	1	
	9	Sorting data-Quick Sorting,Sorting by Multiple Criteria	2	
	10	Filtering data-Quick Filtering, Filtering by Multiple Criteria , Performing Calculations on Filtered Data	2	
	11	AutoFill and Flash Fill	1	
	12	Remove Duplicates	1	
	13	Get External Data - From web,from text and from other sources	2	
III	Excel Functions and formulas		**10**	**18**
	14	Mathematical and Statistical functions(-SUM, AVERAGE, MAX, MIN, ROUND, ABS, SQRT, MOD.,COUNT, COUNTIF, SUMIF, AVERAGEIF, MEDIAN, STDEV, VAR)	2	
	15	Logical Functions(IF, AND, OR, NOT, XOR, IFERROR, IFNA, SWITCH.)	2	
	16	Text Functions (CONCATENATE, LEFT, RIGHT, MID, LEN, SUBSTITUTE, FIND, SEARCH.)	2	
	17	Date & Time Functions-(TODAY, DATE, DAY, MONTH, YEAR, HOUR, MINUTE, SECOND.)	2	
	18	Using formula :Witing a formula ,Cell reference	2	
1V	Data Analysis and Manipulation		**12**	**16**
	19	Introduction to Tables and Data Organization - Creating and formatting tables for effective data management, Sorting and filtering data within tables	3	
	20	Data Analysis Techniques - Advanced functions (VLOOKUP, HLOOKUP, INDEX, MATCH)	3	
	21	PivotTables and PivotCharts - Understanding PivotTables for data analysis, Creating PivotCharts for visual representation	3	
	22	Data Visualization: Creating and customizing various chart types, Effective use of charts for data presentations	3	
V	**Project and Practical Applications**		**30**	

	1	Practical session on real-world applications (Eg: Use advanced functions relevant to field of study, Tabulation of Lab experiments data for better analysis and visualisation)	15	
	2	Course Project: Creating a comprehensive project using Excel features.	15	

References

1. "Microsoft Excel 2019 Step by Step" by Curtis Frye
2. "Excel 2019 Bible" by Michael Alexander and Richard Kusleika
3. "Microsoft Excel 2019 Data Analysis and Business Modeling" by Wayne Winston

Mapping of COs with PSOs and POs :

	PSO1	PSO2	PSO3	PSO4	PS O5	PSO6	PO1	PO2	PO3	PO4	PO5	PO6
CO 1	-	-	-	-	2	1						
CO 2	-	-	2	-	2	1						
CO 3	-	-	2	-	2	1						
CO 4	-	-	2	-	2	1						
CO 5	-	-	3	-	2	1						

Correlation Levels:

Level	**Correlation**
-	Nil
1	Slightly / Low
2	Moderate / Medium

3	Substantial / High

Assessment Rubrics:

- Quiz / Assignment/ Quiz/ Discussion / Seminar
- Midterm Exam
- Programming Assignments (20%)
- Final Exam (70%)

Mapping of COs to Assessment Rubrics :

	Internal Exam	Assignment	Project Evaluation	End Semester Examinations
CO 1		✓		✓
CO 2	✓	✓		✓
CO 3		✓		✓
CO 4	✓			✓
CO 5	✓		✓	✓
CO 6	✓		✓	✓

Programme	B. Sc. Computer Science				
Course Code	CSC2MN103				
Course Title	**Fundamentals of SPSS and R programming**				
Type of Course	**Minor**				
Semester	II				
Academic Level	100-199				
Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	4	3	-	2	75
Pre-requisites	1. Basic computer knowledge 2. Spreadsheet essentials				
Course Summary	This course offers SPSS basics including data management, transformation, visualization and statistical analysis techniques. Also introduces fundamentals of R environment, focusing on data manipulation and visualization.				

Course Outcomes (CO):

CO	CO Statement	Cognitive Level*	Knowledge Category#	Evaluation Tools used
CO1	Perform essential data input and manipulation activities within SPSS.	U	C	Instructor-created exams / Seminar Presentation/ Instructor-created exams/ Quiz
CO2	Implement Data analysis using SPSS	C	C	Assignment / Instructor-created exams
CO3	Compute descriptive statistics and conduct parametric and nonparametric tests in SPSS	C	P	Assignment / Instructor-created exams
CO4	Conduct hypothesis testing and regression analysis in R	Ap	P	Hands-on practical sessions

CO5	Create effective visualizations using SPSS and R.	C	P	
* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C) # - Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge (M)				

Detailed Syllabus:

Module	Unit	Content	Hrs	Marks
I	**Introduction to SPSS**		**12**	**19**
	1	Features – Data View – Variable View – Output Viewer Window – Syntax Editor Window -	1	
	2	Open data file , Save , import from other data source ,data entry , labelling for dummy numbers	2	
	3	Recode in to same variable, Recode in to different variable, Transpose of data, Insert variables and cases	2	
	4	Merge variables and cases, Split, Select cases, Compute total scores	2	
	5	Table looks – Changing column - font style and sizes	2	
	6	Diagrammatic representation	2	
II	**Data Analysis Using SPSS**		**10**	**18**
	7	Estimation of mean, median and mode- Standard deviation and coefficient of variation.	2	
	8	Descriptive statistics, Parametric tests t-test (paired or unpaired), ANOVA (one-way- two way)	2	

	9	Pearson rank correlation, Linear regression	3	
	10	Non parametric tests: Mann Whitney U test, Wilcoxon signed rank test .	2	
	11	Kruskall Wallis test ,Chi- Square test	1	
III	**Overview of R Environment**		**11**	**18**
	12	R editor, Workspace	2	
	13	Data type – Importing and Exporting Data	2	
	14	Basic Computational Ideas – Merges in R. Matrix Determinant – Inverse – Transpose,Trace)	3	
	15	Eigen Values and Eigen Vectors	2	
	16	Construction of Bar, Pie, Histogram, Line Chart, Box Plot, Scatter Plot	2	
IV	**Parametric and Non Parametric testing of Statistical Hypothesis**		**12**	**15**
	17	One Sample t test, Two group t tests, Paired t test, one way ANOVA, two way ANOVA	3	
	18	Wilcoxon, Mann Witney, Kruskal Wallis Simple Correlation	3	
	19	Linear Regression, Multiple Linear Regression, Testing for overall significance of Model Coefficients – Testing for Individual Regression Coefficients.	2	
	20	Outliers Detection Control Charts, Variable Control Chart, x, R, S.	2	
	21	Attribute Control Chart - p, np, c, u. CUSUM Control Chart, EWMA Control Chart.	2	

	22	Process Capability Analysis, Process Capability Analysis		
V	**Hands-on Word Processor and Presentation Tool:** **Practical Applications, Case Study and Course Project**		**30**	
		SPSS 1. Descriptive Statistics 2. Paired –Samples T Test 3. One-Way ANOVA 4. Correlation & Linear Regression 5. Chi- Square Test **R PROGRAMMING** 6. Simple Correlation 7. Linear Regression 8. One- Way ANOVA 9. Paired T test 10. Plotting Bar Chart	20	
		Case study(Example): **SPSS and R** 1. **Case Study: Customer Satisfaction Analysis** Analyze factors influencing customer satisfaction using survey data. Employ SPSS for regression analysis to identify significant predictors such as product quality, pricing, and customer service. Use R programming to analyse data and make predictions.	10	

Reference Books:

1. Michael S. Louis – Beck (1995). Data analysis an introduction, Series: quantitative applications in the social sciences. Sage, Publications. London

2. Jeremy J. Foster (2001). Data analysis using SPSS for windows. New edition. Versions 8-10. Sage publications. London.

3. Sprankle , M., Problem Solving & Programming Concepts, Pearson India
4. Learning Statistics using R By Rndall E.Schumacker, Sage Publication
5. R for Everyone By Jared P.Lander, Pearson Education

Mapping of COs with PSOs and POs :

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PO1	PO2	PO3	PO4	PO5	PO6
CO 1	-	-	2	-	2	3						
CO 2	-	-	2	-	2	3						
CO 3	-	-	3	2	3	3						
CO 4	-	-	3	-	3	3						
CO 5	-	-	3	-	3	3						

Correlation Levels:

Level	**Correlation**
-	Nil
1	Slightly / Low
2	Moderate / Medium
3	Substantial / High

Assessment Rubrics:

- Quiz / Assignment/ Quiz/ Discussion / Seminar
- Midterm Exam
- Programming Assignments (20%)
- Final Exam (70%)

Mapping of COs to Assessment Rubrics :

	Internal Exam	Assignment	Project Evaluation	End Semester Examinations
CO 1	✓	✓		✓
CO 2	✓	✓		✓
CO 3	✓	✓		✓
CO 4	✓	✓	✓	✓
CO 5	✓	✓	✓	✓

Programme	B. Sc. Computer Science				
Course Code	CSC3MN203				
Course Title	Data Visualisation using Python				
Type of Course	**Minor**				
Semester	III				
Academic Level	200-299				
Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	4	3	-	2	75
Pre-requisites	Have an understanding about algorithms and flowchart				
Course Summa ry	This course explores the versatility of Python language in programming and teaches the application of various data structures using Python.				

Course Outcomes (CO):

CO	CO Statement	Cognitie Level*	Knowledge	Evaluation Tools used
CO1	Understand the basic concepts of Python programming	U	C	Instructor- created exams / Quiz
CO2	Apply problem-solving skills using different control structures and loops	Ap	P	Coding Assignments/ Code reading and review
CO3	Design simple Python programs to solve basic computational problems and acquire knowledge of Python's error handling mechanisms to effectively debug programs	Ap	P	Coding Assignments/ exams

CO4	Analyze the various data structures and operations on it using Python	An	P	Instructor-created exams / Case studies
CO5	Apply modular programming using functions	U	C	Instructor- created exams / Quiz
CO6	Identify the necessary Python packages in the domain and create simple programs with it	U, Ap	C, P	Coding
* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C) # - Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge (M)				

Detailed Syllabus:

Module	**Unit**	**Content**	**Hrs**	**Marks**
I	**Introduction to Python**		**12**	18
	1	Features of Python, Different methods to run Python,Python IDE	2	
	2	Comments, Indentation, Identifiers, Keywords, Variables	2	
	3	Standard Data Types	2	
	4	Input Output Functions, Import Functions, range function	1	
	5	Operators and Operands, Precedence of Operators, Associativity	2	
	6	Type Conversion, Multiple Assignment	1	
	7	Expressions and Statements, Evaluation of Expressions	1	
	8	Boolean Expressions	1	
	Control Structures		**12**	19
	9	Decision Making- if statement, if…else statement, if…elif…else statement, Nested if statement	5	

II	10	Loops - for loop, for loop with else, while loop, while loop with else, Nested Loops	5	
	11	Using indentation in Python to define code blocks	1	
	12	Control Statements- break, continue, pass	1	
III	**Data Structures in Python**		**12**	19
	13	Working with strings and string manipulation	3	
	14	List - creating list, accessing, updating and deleting elements from a list	2	
	15	Basic list operations	1	
	16	Tuple- creating and accessing tuples in python	2	
	17	Basic tuple operations	1	
	18	Dictionary, built in methods to create, access, and modify key-value pairs	2	
	19	Set and basic operations on a set	1	
	Functions		**9**	18
IV	20	Built-in functions - mathematical functions, date time functions, random numbers	1	
	21	Writing user defined functions - function definition, function call, flow of execution, parameters and arguments, return statement	6	
	22	Recursion. Introduction to basic Python libraries (e.g., math, random)	2	
	Hands-on Data Structures: Practical Applications, Case Study and Course Project		**30**	
Design programs from the concepts listed below. Select the topics and programs suited for your domain				

V	1	• Read input, include casting that input to the appropriate type • Select from one of several alternatives by using an if-elif or if-elif-else statement • Use the range() function in a form loop • Call and use functions residing in the math module		
		Case study: ● Design a basic calculator application in Python that can perform addition, subtraction, multiplication, and division. ● Create a Python program that retrieves weather data from an API (e.g., OpenWeatherMap) and displays it.		

	4	**Data Structures in Python** • String - Create a string , Indexing / Looping / Slicing • Lists - Create a list , Indexing /Looping / Slicing , Adding items / Modifying items / Removing items • Tuples - Create a tuple , Indexing / Looping / Slicing / Adding items to a tuple • Dictionary - Create a dictionary and access values with key / Adding a key- value pair / Adding to an empty dictionary /Modifying values in a dictionary / Removing key-value pair		

	5	**Function** • Call functions residing in the math module • Define a function for later use • Pass one or more values into a function • Return one or more results from a function • Call a function that you have defined previously		

Mapping of COs with PSOs and POs :

	PSO 1	PSO 2	PSO 3	PSO 4	PSO 5	PSO 6	PO 1	PO2	PO3	PO4	PO5	PO6
CO 1	-	1	2	3	1	1						
CO 2	-	1	2	3	1	1						
CO 3	-	2	2	3	1	1						
CO 4	1	1	-	-	1	-						
CO 5	1	1	2	2	1	-						
CO 6	-	1	2	2	2	1						

Correlation Levels:

Level	Correlation
-	Nil
1	Slightly / Low
2	Moderate / Medium
3	Substantial / High

Assessment Rubrics:

- Quiz / Assignment/ Quiz/ Discussion / Seminar
- Midterm Exam
- Programming Assignments (20%)
- Final Exam (70%)

Mapping of COs to Assessment Rubrics :

	Internal Exam	Assignment	Project Evaluation	End Semester Examinations
CO 1	✓			✓
CO 2	✓	✓	✓	✓
CO 3	✓		✓	✓
CO 4	✓	✓	✓	✓
CO 5	✓			✓
CO 6	✓			✓

Reference Books:

1. Jose, Jeeva. Taming Python By Programming. Khanna Book Publishing, 2017. Print.
2. Downey, Allen. Think Python. Green Tea Press, 2nd ed. 2009

Programme	B. Sc. Computer Science				
Course Code	CSC1MN104				
Course Title	Computer Essentials with Word Processing & Presentation				
Type of Course	**Minor**				
Semester	I				
Academic Level	100-199				
Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	4	3	-	2	75
Pre-requisites	1. Fundamental Mathematics Concepts: Number System				
Course Summary	This course serves as an introductory exploration into the foundational concepts of computing. Through a combination of lectures, hands-on exercises, and practical assignments, participants develop a holistic understanding of computer fundamentals. Ultimately, this course serves as a cornerstone for further studies in computer science, information technology, and related disciplines, empowering learners to navigate and contribute to the ever-evolving landscape of computing.				

Course Outcomes (CO):

CO	CO Statement	Cognitive Level*	Knowledge Category#	Evaluation Tools used
CO1	Gain proficiency in understanding and representing data in various forms, including binary, decimal, hexadecimal, and character encodings.	Ap	F	Instructor-created exams / Quiz
CO2	Understand the basic principles of computer architecture and organization	U	C	Assignment / Demonstrations
CO3	Understand the concept of software and its significance in computing and be familiar with various types of software, including system software, application	U	C	Seminar Presentation / Group Tutorial Work

	software, and utility software.			
CO4	Understand the basic principles of document design and layout for enhanced readability and visual appeal	Ap	P	Hands-on practical sessions
CO5	Understand the importance of effective communication and visual aids in presentations.	Ap	P	Hands-on practical sessions
CO6	Acquire practical skills through hands-on exercises and projects, preparing participants to apply their knowledge in academic, professional, and personal contexts.	Ap	P	Hands-on practical sessions
* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C) # - Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge (M)				

Detailed Syllabus:

Module	Unit	Content	Hrs	Marks
I	**Introduction to Computer Science and data representation**		**10**	**20**
	1	Introduction to Computers: Generation, Classification, Characteristics of Computers, Significance	2	
	2	Number Systems :Binary, Decimal, Octal, Hexadecimal.	2	
	3	Conversion from one base to another	3	
	4	Computer Codes: BCD Code, Excess 3 Code, ASCII Code, Unicode, Gray Code	3	
II	**Basic Computer Organization**		**10**	**20**
	5	CPU organisation :Arithmetic and Logic Unit, Control Unit	1	
	6	Memory hierarchy: Registers, Cache, Primary Memory, Secondary Memory	2	
	7	Primary Storage: RAM(SRAM, DRAM), ROM**(Masked ROM,** PROM,EPROM,EEPROM)	2	
	8	Secondary storage: SSD,HDD, Magnetic tapes, Disk Storage	2	

	9	Input/Output Unit:- Input Device: Keyboard, MouseTouchpad, Trackball, Scanner, Graphics Tablet, Microphone, Webcam, Joystick/Gamepad, Biometric Input Devices Output Devices: Monitor/Display, Printer,Projector, Speakers, Headphones, Plotter	3	
III	**Understanding Softwares**		**10**	**20**
	10	**Introduction to Software** (Definition and Importance of Software, Types of Software-System software, Application Software, Prop oratory vs Open source)	2	
	11	**Operating Systems** (Introduction to Operating Systems , Common Operating Systems,User Interfaces)	2	
	12	**Device Drivers and Utilities (Device Drivers** , **System Utilities** ,**Productivity Software** ,**Multimedia Software)**	2	
	13	Computer languages(Machine, Assembly and HighLevel), Language Translator- Assembler, Compiler, Interpreter	2	
	14	**Security Software and Best Practices(Antivirus Programs** ,F**irewalls and Security Suites**,**Best Practices for Software Security)**	2	
IV	**Introduction to Word Processing & Presentation**		**15**	**10**
	15	**Basics of Word Processing**: Creating, Opening, Saving, and Closing Documents, Text Entry and Formatting (Font, Size, Color),Paragraph Formatting (Alignment, Spacing),Copying, Cutting, and Pasting Text,Spell Check and Grammar Check)	2	
	16	**Advanced Word Processing Techniques**(Styles and Templates,Tables and Graphics (Inserting, Formatting),Headers and Footers,Page Layout (Margins, Orientation),Document Views (Print Layout, Draft, Outline	2	
	17	Advanced Graphics and Multimedia(SmartArt and Shapes , Customizing SmartArt and shapes,Embedding and Linking Media,Advanced techniques for embedding and linking images, audio, and video)	2	
	18	),Document Collaboration (Track Changes, Comments),Mail Merge for Personalized Documents)	1	
	19	**Introduction to Presentation Software(**Creating a New Presentation, Slide Basics (Adding, Deleting, Rearranging), Slide Layouts and Choosing Templates,Text Entry and Formatting Inserting and Formatting Images and Shapes)	2	

	20	**Enhancing Presentations with Multimedia**(Inserting and Formatting Media (Audio, Video),Transitions Between Slides,Master Slides for Consistent Formatting,Design and Themes for Visual Appeal)	3	
	21	**Animations for Text and Objects (**Slide Show Setup (Timings, Rehearsal)	2	
	22	**Effective Presentation Delivery (**Tips for Engaging Presentations, Presenter View and Speaker Notes, Handling Q&A Sessions, Dealing with Technical Issues, Customizing Presentations for Different Audiences, Printing and Exporting Slides	1	
V	**Hands-on Word Processor and Presentation Tool:** **Practical Applications, Case Study and Course Project**		**30**	
		Identification and familiarization of Hardware Components (Processor, RAM,ROM, Peripheral devices, SSD, HDD, SMPS, Motherboard, Ports)	5	
		Microsoft Word: 1. Document Formatting: ● Create a new document, set margins to 1 inch, and change the page orientation to landscape. ● Apply a consistent font style, size, and color to the entire document. 2. Paragraph Formatting: ● Create a bulleted or numbered list with at least three items. ● Adjust the indentation and line spacing for a specific paragraph. 3. Headers and Footers: ● Insert a header with the document title and page number on the right. ● Add a footer with the date aligned to the center. 4. Tables and Graphics: ● Create a table with four columns and three rows. ● Insert an image into the document and adjust its position. 5. Styles and Themes: ● Apply a heading style to a section of text. ● Change the document theme to give it a different look. **Microsoft PowerPoint:**	20	

		6. Slide Creation: ● Create a new PowerPoint presentation and add five slides. ● Apply different slide layouts to each slide. 7. Text and Object Formatting: ● Add a title to the first slide and format it with a unique font and color. ● Insert a shape and customize its fill and outline colors. 8. Transitions and Animations: ● Apply a slide transition between the first and second slides. ● Add an entrance animation to a text box on the third slide. 9. Master Slides: ● Customize the master slide with a background color or image. ● Add a placeholder for slide numbers in the master slide. 10. Delivery and Export: ● Set up presenter view for a slideshow. Export the presentation as a PDF document		
		Case Study: Exploring feature of PowerPoint to enhance presentation skill	5	

Reference Books:

1. Tanenbaum, Andrew S. and Herbert Bos. Modern Operating Systems. 4th ed., Pearson, 2014.

2. Brookshear, J. Glenn. Computer Science: An Overview. 13th ed., Pearson, 2014.

3. Norton, Peter. Introduction to Computers. 7th ed., McGraw-Hill, 2016.

4. Patterson, David A. and John L. Hennessy. Computer Organization and Design: The Hardware/Software Interface. 5th ed., Morgan Kaufmann, 2013.

5. Stallings, William. Computer Organization and Architecture: Designing for Performance. 10th ed., Pearson, 2016.

6. Hennessey, John L. and David A. Patterson. Computer Architecture: A Quantitative Approach. 6th ed., Morgan Kaufmann, 2017.

Mapping of COs with PSOs and POs :

	PSO1	PSO2	PSO3	PSO4	PS O5	PSO6	PO1	PO2	PO3	PO4	PO5	PO6
CO 1	3	-	-	-	-	-						
CO 2	2	3	-	-	-	-						

CO 3	-	-	1	1	-	-						
CO 4	-	-	2	3	-	-						
CO 5	-		-	3	-	-						
CO 6	-	-	-	3	-	-						

Correlation Levels:

Level	**Correlation**
-	Nil
1	Slightly / Low
2	Moderate / Medium
3	Substantial / High

Assessment Rubrics:

- Quiz / Assignment/ Quiz/ Discussion / Seminar
- Midterm Exam
- Programming Assignments (20%)
- Final Exam (70%)

Mapping of COs to Assessment Rubrics :

	Internal Exam	Assignment	Project Evaluation	End Semester Examinations
CO 1	✓	✓		✓
CO 2	✓	✓		✓
CO 3	✓	✓		✓
CO 4			✓	✓
CO 5			✓	✓
CO 6			✓	

Programme	B. Sc. Computer Science Minor				
Course Code	CSC2MN104				
Course Title	Web Design Trends and Techniques				
Type of Course	**Minor**				
Semester	II				
Academic Level	100-199				
Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	4	3	-	2	75
Pre-requisites	1. Knowledge in Computers. 2. Basic knowledge in Internet and Basic knowledge Computers and Internet				
Course Summary	The aim of this course is to provide students with an understanding of the basic concepts in web browser and to achieve the basic web designing skills				

Course Outcomes (CO):

CO	CO Statement	Cognitive Level*	Knowledge Category#	Evaluation Tools used
CO1	To get general introduction to internet	U	C	
CO2	To identify and analyse the current trends in web designing	Ap	P	
CO3	To understand basic knowledge in HTML5 and CSS3 for responsive web design	Ap	P	
CO4	To learn how to design a simple web applications	Ap	P	
CO5	To incorporate user experience principle in web design	Ap	P	
CO6	To Enable student to become	Ap	P	

	proficient in web designing through current technologies			

* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C)

- Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge (M)

Detailed Syllabus:

Module	**Unit**	**Content**	**Hrs**
I	**Introduction to Web Design**		**09**
	1	Overview of Internet	1
	2	Over view of Internet Security	1
	3	Client Server System	1
	5	Websites and Digital Communication Tools	1
	6	Collaboration for Website Development	1
	7	Understanding the evolution of web design	2
	8	Exploring current design trends	1
	9	Overview of innovative websites	1
II	**HTML – Building the Foundation**		**12**
	10	Understanding the basic structure of web pages(Role of HTML, basic concept of webpage, html document structure <!DOCTYPE>,<html>,<head>,<body>).	2
	11	Exploring tags, attributes, and their significance (font type, text formatting tag, otrher text related tag, heading, paragraphs, list, link, image, common attributes like class, id, src, alt, href).	2
	12	Creating interactive forms to collect user data(form element tag like <form>,<input>,<textarea>,<select>,<button>,various form controls like text input, button, drodwonbox).	3
	13	Designing and structuring tabular data.(Basic table structure tag, colspan, rowspan)	2
	14	Enhancing the meaning and structure of your content(understanding semantic elements, benifit of semantic	3

		HTML).	
III	**CSS – Styling Your Web Pages**		**12**
	15	Introduction to CSS(understanding the concept of styling, presentation, basic synatx electors, properties and values, inline, internal and external style, different types of selectors)	3
	16	Introduction to common CSS properties(color,font,text,margin, padding, border, background), CSS box model(margin, border, padding and content), positioning elementss (static, relative, absolute, fixed, z-index)	3
	17	Introduction to layouts in web desigining, The role of responsive layouts, Flexbox layout model, creating Grid, Media queries and breakpoints	3
	18	Transition properties(duration, timing function, property), CSS Animation, Adding interactive hover effect. Overview of CSS frameworks and their benefits. Introduction to Bootstrap and its features.	3
IV	**JavaScript Essentials**		**12**
	19	Overview of Javascript, declaring the variables and understanding data types. Object in Javascript, basic operations and control flow in Javascript	3
	20	Understanding the Document Object Model (DOM).Using selectors to target HTML elements. Modifying content, attributes, and styles dynamically. Using selector, content, attributes and styles dynamically. Creating and Deleting elements. Navigating through the DOM hierarchy.	3
	21	Understanding events triggered, common events, writing event handlers, Bubbling and capturing phases of event propagation, controlling event flow, accessing event object	3
	22	Overview of JavaScript libraries and their benefits.Selecting elements, manipulating the DOM, and handling events with jQuery.Applying visual effects and animations with jQuery. Fade, slide, show/hide, and custom animations.Making asynchronous requests with jQuery.ajax(). Handling JSON data and updating the DOM dynamically.	3

V	**Hands-on Programming in Java(Using VSCode, Atom, Aptana Studio):**		**30**
	Practical Applications, Case Study and Course Project		
	1	Implement the following:	
		1. Program for implementing html tags	20
		2. Write program for implementing Style a paragraph (<p>) to have a red color, a font size of 16px, and a bold font weight	
		3. Write a program to implement CSS Box Model	
		4. Create a simple layout using Flexbox, with three div elements aligned horizontally.	
		5. Implement a media query that changes the background color of a webpage when the screen width is less than 600 pixels.	
		6. Write a JavaScript function that changes the text content of an HTML element with the id "demo" to "Hello, World!" when a button is clicked.	
		7. Declare a variable in JavaScript and assign it a string value. Also, mention the data type of the variable.	
		8. Write a JavaScript program for attaching a click event to a button.	
		9. Use JavaScript to make an asynchronous request to a JSON file and display the data on the webpage.	
		10. Use JavaScript to perform AJAX operation	
	2	Case Study	2
	3	**Project:** Build a web application for perform responsive web application.	8

Text Book :

1. HTML5 Black Book, Covers CSS3, JavaScript, XML, XHTML, AJAX, PHP And JQuery (Second Edition), Dreamtech Press,ISBN: 9789351199076

References :

1. Internet and World Wide Web, H.M.Dietel, Pearson.
2. Mastering HTML, CSS & Javascript Web Publishing (English, Paperback, Lemay Laura)
3. Web Designing (English, Paperback, Hirdesh Bhardwaj)

Mapping of COs with PSOs and POs :

	PSO1	PSO2	PSO3	PSO4	PS O5	PSO6	PO1	PO2	PO3	PO4	PO5	PO6
CO 1	1	-	3	3	-	-						
CO 2	1	-	3	3	-	-						
CO 3	-	-	3	3	2	3						
CO 4	-	-	2	3	-	-						
CO 5	-	-	3	3	2	3						
CO 6			3	3	3							

Correlation Levels:

Level	**Correlation**
-	Nil
1	Slightly / Low
2	Moderate / Medium
3	Substantial / High

Assessment Rubrics:

- Quiz / Assignment/ Quiz/ Discussion / Seminar
- Midterm Exam
- Programming Assignments (20%)
- Final Exam (70%)

Mapping of COs to Assessment Rubrics :

	Internal Exam	Assignment	Project Evaluation	End Semester Examinations
CO 1	✓			✓
CO 2	✓			✓
CO 3	✓	✓		✓
CO 4		✓		✓
CO 5		✓		✓

Programme	B. Sc. Computer Science				
Course Code	CSC3MN204				
Course Title	Programming fundamentals using C				
Type of Course	**Minor**				
Semester	III				
Academic Level	200-299				
Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	4	3	-	2	75
Pre-requisites	1. Basic Computer Literacy 2. Basic Problem-Solving Skills				
Course Summary	This course teaches the basics of programming using the C language. C is a powerful and widely used programming language known for its efficiency and flexibility. Through this course, students will learn how to write, understand and debug C code to solve various problems and build simple applications.				

Course Outcomes (CO):

CO	CO Statement	Cognitive Level*	Knowledge Category#	Evaluation Tools used
CO1	Demonstrate a solid understanding of fundamental programming concepts	An	P	Instructor-created lab exams / Quiz
CO2	Develop effective problem-solving skills by applying algorithmic thinking and logical reasoning.	An	P	Problem-solving assessments
CO3	Gain proficiency in writing, compiling, debugging, and executing	Ap	P	Modelling Assignments

	C programs to implement algorithms, solve problems, and create applications.			
CO4	Learn techniques to write efficient and optimized C code, including memory management, algorithm design, and performance tuning, to produce high-quality and scalable software solutions.	Ap	P	Modelling Assignments/ / Case studies
CO5	Understand and apply software development practices such as modular programming, code documentation and debugging techniques to write maintainable and robust C programs.	Ap	P	Modelling Assignments/ / Case studies
CO6	Develop critical thinking skills by analyzing and evaluating C code, identifying errors and inefficiencies, and proposing solutions to improve code quality and performance.	Ap	P	Hands-on exercises
* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C) # - Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge (M)				

Detailed Syllabus:

Module	Unit	Content	Hrs	Marks
I	**Problem solving and logical Thinking**		**10**	**15**
	1	Overview of computational thinking concepts. Definition of algorithm and its characteristics .Understanding the importance	2	

		of algorithms in problem-solving		
	2	Algorithm Development:Steps involved in designing algorithms	2	
	3	Pseudocode as an intermediate step in algorithm development.	1	
	4	Understanding flowchart symbols and their meanings .Learning to represent algorithms using flowcharts.	2	
	5	Raptor as a precursor to text-based programming languages	2	
	6	Drawing simple flowcharts	1	
II	**Introduction to C**		**10**	**20**
	7	Structure of C program	2	
	8	C Character Set, Keywords, Identifiers	1	
	9	Data Types, Variables, Declarations, Symbolic Constants	2	
	10	Operators :Arithmetic, Logical, Relational & Equality, and Unary, Operator Precedence and Associativity	2	
	11	Library Functions, Comments	1	
	12	I/O functions- Formatted scanf() & printf().	2	
III	**Control Statements,Arrays & Strings**		**14**	**20**
	13	Selection Statements: if, if-else, switch	3	
	14	Iteration: while, do while, for	4	
	15	Arrays: One dimensional and Two Dimensional(introduction only)	3	
	16	Strings:Basic string handling functions	2	
	17	Structure :Definition,Processing-period Operator, Union(Concepts only)	2	
1V	**User defined Functions**		**11**	**15**
	18	Definition of function,Advantages, Understanding function prototypes and declarations	3	
	19	Introduction to function definitions and function calls	3	
	20	Exploring function parameters : Actual and Formal parameters	2	
	21	Recursion	2	

	22	Pointers-declarations(Basic concept only)	1	
V	**Hands-on C:** **Practical Applications, Case Study and Course Project**		**30**	
	1	Write a C program using Variables and Data Types Write a C program using Arithmetic Operations Write a C program using Loops Write a C program using Arrays Write a C program using Functions Write a C program using Strings	20	
	2	Case study: 1. Library Management System: 2. Develop a program to manage a library's collection of books. Implement functions for adding, removing, and searching for books. 3. Ticket Booking System: Design a program to manage ticket bookings for a cinema or theater.	5	
	3	Capstone/Course Project: Design a real-time project in C	5	

Reference:

1. Balagurusamy, E. Programming in ANSI C. Tata McGraw-Hill Education, 2019.
2. King, K. N. C Programming: A Modern Approach. 2nd ed., W. W. Norton & Company, 2008.
3. Kernighan, Brian W., and Dennis M. Ritchie. The C Programming Language. 2nd ed., Prentice Hall, 1988.
4. Prata, Stephen. C Primer Plus. 6th ed., Addison-Wesley, 2013.
5. Perry, Greg. Absolute Beginner's Guide to C. 3rd ed., Que Publishing, 2014.
6. Oualline, Steve. Practical C Programming. 3rd ed., O'Reilly Media, 1997.
7. Hanly, Jeri R., and Elliot B. Koffman. Problem Solving and Program Design in C. 8th ed., Pearson, 2016.
8. Gottfried, Byron S. Programming with C. 2nd ed., McGraw-Hill, 1996.
9. Holmes, Dan. C in a Nutshell. 2nd ed., O'Reilly Media, 2015.

Mapping of COs with PSOs and POs :

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PO1	PO2	PO3	PO4	PO5	PO6
CO 1	3	1	-	-	-	1						
CO 2	1	-	2	-	-	-						

CO 3	-	-	2	-	-	-						
CO 4	-	1	3	3	-	3						
CO 5	-	2	3	3	-	3						
CO 6	-	-	-	-	-	3						

Correlation Levels:

Level	**Correlation**
-	Nil
1	Slightly / Low
2	Moderate / Medium
3	Substantial / High

Assessment Rubrics:

- Quiz / Assignment/ Quiz/ Discussion / Seminar
- Midterm Exam
- Programming Assignments (20%)
- Final Exam (70%)

Mapping of COs to Assessment Rubrics :

	Internal Exam	Assignment	Project Evaluation	End Semester Examinations
CO 1		✓		✓
CO 2	✓	✓		✓
CO 3		✓		✓
CO 4	✓			✓
CO 5	✓		✓	✓
CO 6	✓		✓	✓

Programme	B. Sc. Computer Science				
Course Code	CSC1MN105				
Course Title	INTRODUCTION TO IT				
Type of Course	**Minor**				
Semester	I				
Academic Level	100-199				
Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	4	3	-	2	75
Pre-requisites	Basic understanding of computer operation Basic Science fundamentals				
Course Summary					

The course will create an overall generic awareness about scope of the field of IT and to impart basic personal computing skills and will create background knowledge for the various courses in the programme.

Course Outcomes (CO): .

CO	CO Statement	Cognitive Level*	Knowledge Category#	Evaluation Tools used
CO1	Understand basic terminology in the field of IT	U	C	Instructor-created exams / Assignment
CO2	Identify and describe essential computer hardware components.	U	C	Viva Voce
CO3	Comprehend the distinction between system software and application software and their respective roles in computer functionality	U	C	Practical / Group Work
CO4	Produce documents with precision and efficiency using LaTeX	Ap	P	Practical / Group Work

CO5	Understand the basics of networking and internet concepts.	U	C	Practical/Exam/ Assignments

* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C)

\# - Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge (M)

Detailed Syllabus:

Module	Unit	Content	Hrs	Marks
I	**Characteristics of Computers**		**10**	**15**
	1	Characteristics: Automatic, speed, accuracy, memory, diligence; Digital signals, Instruction set	**2**	
	2	Evolution of computers and generation of computers	**2**	
	3	Classification of computers: Microcomputer, Minicomputer, mainframes, Supercomputers; Personal computers: Desktop,Laptops	**2**	
	4	Binary System and data representation (BCD,ASCII,Unicode)	**3**	
	5	Hardware & Software,Von Neumann model.	**1**	
II	**Hardware**		**13**	**19**
	6	CPU- CU, ALU, Registers	**2**	
	7	Memory units: RAM(SDRAM, DRAM)- feature wise comparison only); ROM-(PROM,EPROM,EEPROM)	**2**	
	8	Auxiliary storage: Flash memory ,Magnetic devices, HDD SSD,	**2**	
	9	Input devices - keyboard, mouse, scanner, speech input devices, digital camera, Touch screen, Joystick, Optical readers, bar code reader	**3**	
	10	Output devices: Display device-Mouse, Joystick, Touchscreens, LCD, Plasma, LED, Printers: Dot-matrix, Inkjet, Laser; Plotters, speaker.	**4**	
III	**Software**		**12**	**20**
	11	System software, Application software ,examples	**1**	

	12	Operating systems: Single user, Multitasking, Time-sharing ,multi-user;	**1**	
	13	Basic features of OS: Process management, Memory management, Device Management,	**2**	
	14	Booting, POST	**1**	
	15	Computer Viruses & Protection	**2**	
	16	Free software, Open source	**1**	
	17	LaTeX : Introduction, installation, and basic document creation,Text styling, sectioning, and lists, Citations and references, Inserting images and creating tables.	**4**	
IV	**Computer Networks**		**10**	**16**
	18	Requirements for a network	**1**	
	19	Server, Workstation, switch, router, network operating systems	**2**	
	20	Internet: brief history, World Wide ,Web, Websites, URL, Browsers, Search engines	**2**	
	21	Internet connections: ISP, Dial-up, cable modem	**2**	
	22	Characteristics of web-based systems, Web pages, introduction to HTML.	**3**	
V	**Practical Applications, Case Study and Course Project**		**30**	
	1	1. Document Basics: Create a document with a title, author, and date. 2. Sections and Headings: Add sections and subsections with headings. 3. Lists: Insert bulleted and numbered lists. 4. Graphics: Insert images and adjust their placement. 5. Tables: Create tables with rows and columns. 6. References: Add citations and create a bibliography. **7.** Formatting: Apply styles like bold, italics, and underline.	**20**	

	2	**Case study:** 1. Academic Essay:Write an essay on a chosen topic, formatting headings, paragraphs, and citations using LaTeX commands. 2. Research Paper: Write a structured paper on a chosen topic, incorporating sections, citations, and formatting using LaTeX.	**10**	

References

1. P. K Sinha, Fundamentals of Computers
2. Behrouz A Forouzan, Data Communication & Networking, MC Graw Hill

Reference Books:

3. Tanenbaum, Andrew S. and Herbert Bos. Modern Operating Systems. 4th ed., Pearson, 2014.

4. Norton, Peter. Introduction to Computers. 7th ed., McGraw-Hill, 2016.

5. Harel, David. Algorithmics: The Spirit of Computing. 3rd ed., Addison-Wesley, 2004.

6. LaTeX Beginner's Guide - Second Edition. Author(s): Stefan Kottwitz.Publisher(s): Packt Publishing.

Mapping of COs with PSOs and POs:

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PO1	PO2	PO3	PO4	PO5	PO6
CO 1	1	1	-	-	-	-						

CO 2	2	2	-	-	-	-						
CO 3	2	2	-	-	1	-						
CO 4	-	-	-	-	2	-						
CO 5	-	-	-	-	1	-						

Correlation Levels:

Level	**Correlation**
-	Nil
1	Slightly / Low
2	Moderate / Medium
3	Substantial / High

Assessment Rubrics:

- Quiz / Assignment/ Quiz/ Discussion / Seminar
- Midterm Exam
- Programming Assignments (20%)
- Final Exam (70%)

Mapping of COs to Assessment Rubrics:

	Internal Exam	Assignment	Project Evaluation	End Semester Examinations
CO 1	✓	✓		✓
CO 2	✓	✓		✓
CO 3	✓	✓		✓
CO 4	✓	✓		✓
CO 5	✓	✓		✓
CO 6	✓	✓		✓

Programme	B. Sc. Computer Science				
Course Code	CSC2MN105				
Course Title	Efficient Office Dynamics				
Type of Course	Minor				
Semester	II				
Academic Level	100-199				
Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	4	3	-	2	75
Pre-requisites	Basic understanding of computer operation Basic Science fundamentals				
Course Summary					

This course provides students with ample training in office automation tools, focusing on Microsoft Word, Excel, and Presentation software, along with internet-based applications.

Course Outcomes (CO): .

CO	CO Statement	Cognitive Level*	Knowledge Category#	Evaluation Tools used
CO1	Understand basic concepts of office automation and the need for technology in the workplace.	U	P	
CO2	Develop proficiency in using a variety of office automation tools, including word processing software, spreadsheet applications, presentation software	C	P	
CO3	Understand the importance of maintaining an organized and accessible document repository.	U	C	
CO4	Develop basics of office automation tools integrated with internet technologies like cloud-based productivity suites, collaboration	U	P	

	platforms, communication tools.			
CO5	develop enhanced through hands-on practice and practical exercises, to increase productivity skills using office automation tools.	Ap,C	P	
* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C) # - Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge (M)				

.

Detailed Syllabus:

Module	Unit	Content	Hrs	Marks
I	**Documentation Using a Word Processor (Oppen Office or MS Office)**		**12**	**20**
	1	Introduction to Office Automation: Definition and types	**1**	
	2	Word Processor: Definition, Use, Options, Ribbon Menu	**1**	
	3	Creating and Editing Document: New, Open, Save, Working with Text (Insert, Selecting, Deleting, Copy, Cut, Paste, Drag and Drop)	**2**	
	4	Formatting the Document : Font Size, Font Style, Margin, Header and Footer, Page Number, Numbering, Bullets, Tables, Image, Hyperlink, Autocorrect, Proofing Tools, Dictionary, Book mark, Find and Replace	**3**	
	5	Advanced Features: Inserting Pictures, Shapes, Smart Art, Charts, Orientation, Page Size, Symbols and Special Characters, Equations,	**2**	
	6	Intending Tabs, Line and Paragraph Spacing, Textbox, Word Arts, Auto Recover, Print Options	**1**	
	7	Mail Merge and Macros	**2**	
II	**Electronic Spread Sheet(OpenOffice Calc/MS-Excel)**		**15**	**20**
	8	Spreadsheet: Definition, Advantage, Use, Workbook, Worksheet	**1**	

	9	Creating and Editing Spreadsheet: File Tab, Cell, Tabs, Groups, Commands, Help	**1**	
	10	Spreadsheet Essential: View Button, Sheet Area, Row Bar, Column Bar, Status Bar, Autofill, Range, Saving Worksheet and Workbook, Hiding and Unhiding	**3**	
	11	Formatting the Spreadsheet: Spell Check, Find and Replace, Insert, Cell Formatting, Font, Rotating Cell, Alignment, Merge Cell, Boarder, Freezing and Unfreezing, Margin,	**3**	
	12	Formulas and Functions: Basic Maths Functions, AutoSum, Roman, Round, Basic Statistical Functions, Basic Financial Functions	**4**	
	13	Advanced Features: Macro, Pivot Table, Preparing Graphs and Charts	**3**	
III	**Working with Presentation (OpenOffice Impress/MS-Power Point)**		**10**	**15**
	14	Presentation: Definition, Use, Advantage,	**1**	
	15	Creating Presentation: Create, Open, Save, Add Slide, Insert Picture, Insert Clip Arts	**2**	
	16	Manipulating Presentation: Style, Theme, Font, Header and Footer, Hyperlink, Inserting Tables and Charts, Slide Transition	**3**	
	17	Organisational Chart and Layered Objects,	**2**	
	18	Manage Animation and Effects	**2**	
IV	**Internet and World Wide Web**		**8**	**15**
	19	Internet: Definition, What is Network (LAN, WAN, MAN), Internet Service Provider HTTP, FTP, Email, World Wide Web and its evolution, URL.	**2**	
	20	Internet Protocols (Concept Only), Domain Name Server, Internet Address, Wi-Fi	**2**	
	21	Search Engine(Google, Bing, Yahoo, DuckDuckgo, Ask.com) and Browsers (Internet Explorer, Firefox,	**2**	

		Chrome, Opera, Safari, Netscape)		
	22	Cloud-based platforms and applications. SaaS ,Cloud storage and file sharing services	**2**	
V	**Hands-on Training:** **Practical Applications, Case Study and Course Project (Use any Office Software)**		**30**	**30**
	1	**Word Processing** 1. Perform Paragraph formatting. 2. Perform Newspaper style Document. 3. Perform Table creation. 4. Perform Mail merge. 5. Perform Page formatting & printing.	**30**	
	2	**Spreadsheet** 6. Perform Worksheet entries. 7. Perform Cell Forming. 8. Chart creation. 9. Perform Basic Mathematical Functions. 10. Performa Basic Statistical Functions. 11. Perform any 3 Financial Functions.		
	3	**Presentation Software** 12. Creating presentations and performing basic formatting. 13. Perform Animations like adding pictures, slide orientation, and slide theme. 14. Add Sound to Slideshow. 15. Create Organizational Charts and Layered Objects.		
	4	**Internet**		

		16. Crimping and Connecting LAN CableIP address configuring. 17. Assign Static I/P Address. 18. Setup a Wired LAN with more than two systems and share the documents. 19. Setup a Wireless LAN with more than two systems and share the documents 20. Installing any Brower and assign default serach engine as Google		

References

1. Russell A. Stultz, *Learn Microsoft Office*, BPB Publication.
2. Winston, Microsoft Excel 2013: Data Analysis and Business Modeling, Prentice Hall India Learning Private Limited (2013), ISBN: 9788120349605
3. H. M.Deitel, P. J. Deitel, et al., *Internet & World Wide Web - How to program*, Prentice

Mapping of COs with PSOs and POs:

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PO1	PO2	PO3	PO4	PO5	PO6
CO 1	1	2	-	-	-	-						
CO 2	-	2	-	-	-	-						
CO 3	-	2	-	-	1	-						
CO 4	-	2	-	-	-	-						
CO 5	-	2	-	-	1	-						
CO 6	-	2	-	-	1	-						

Correlation Levels:

Level	**Correlation**
-	Nil
1	Slightly / Low
2	Moderate / Medium

3	Substantial / High

Assessment Rubrics:

- Quiz / Assignment/ Quiz/ Discussion / Seminar
- Midterm Exam
- Programming Assignments (20%)
- Final Exam (70%)

Mapping of COs to Assessment Rubrics:

	Internal Exam	Assignment	Project Evaluation	End Semester Examinations
CO 1	✓	✓		✓
CO 2	✓	✓		✓
CO 3	✓	✓		✓
CO 4	✓	✓		✓
CO 5	✓	✓		✓
CO 6	✓	✓		✓

Programme	B.Sc Computer Science				
Course Code	CSC3MN205				
Course Title	Mastering Content Management Systems				
Type of Course	**Minor**				
Semester	III				
Academic Level	200-299				
Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	4	3	-	2	45
Pre-requisites	1. Familiarity with web content management systems (CMS). 2. Basic knowledge of internet technologies provides a foundation for learning web design.				
Course Summary	The course covers fundamental web design concepts, including HTML and CMS principles, and focuses on Drupal as a robust content management system. Students will learn to create and customise websites using Drupal, exploring its features, such as content types, themes, and modules to build dynamic and interactive web pages.				

Course Outcomes (CO):

CO	CO Statement	Cognitive Level*	Knowledge Category#	Evaluation Tools used
CO1	Cultivate a robust understanding of web design fundamentals, laying a strong foundation for their journey into the dynamic world of digital design and development.	U	C	Assignment / Instructor-created exams / Quiz
CO2	Attain comprehensive knowledge and practical proficiency in Content Management Systems (CMS), empowering to navigate and excel in the ever-evolving landscape of digital content creation and management.	U	C	Assignment / Instructor-created exams / Quiz

CO3	Develop expertise in Drupal, a widely used CMS platform, gaining comprehensive understanding of its features, configuration, and installation processes, thus preparing them for proficient and innovative web development endeavors.	Ap	P	Practical Assignment / Instructor-created exams / Quiz
CO4	Impart a comprehensive understanding of website development using Drupal and facilitate the acquisition of expertise across various options within the Drupal ecosystem.	Ap	P	Practical Assignment / Instructor-created exams / Quiz
CO5	Gain an understanding of how to apply web design concepts to real-world scenarios, effectively designing and developing functional and aesthetically pleasing websites utilizing the Drupal CMS.	C	P	Practical Assignment / Instructor-created exams / Quiz
CO6	Develop proficiency in advanced website management skills, including installing and configuring modules, managing menus, and more, to effectively navigate and optimize the functionality of websites built on the Drupal platform.	C	P	Practical Assignment / Instructor-created exams / Quiz
* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C) # - Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge (M)				

Detailed Syllabus:

Module	Unit	Content	Hrs	Marks
I	**Introduction to Web Designing**		**08**	**15**
	1	Basics of Web Designing -World Wide Web (WWW), W3C, Web Browser	2	
	2	Web Server, Web Hosting, Web Pages	2	
	3	Domain Name System, Uniform Resource Locator	1	
	4	Overview of HTML: Definition and Basic structure	3	

II	**Introduction to CMS**		**09**	**15**
	5	Introduction to Content Management Systems (CMS) - Features of CMS	2	
	6	Web Content Management System	2	
	7	Components of Content Management System	2	
	8	Enterprise Content Management System	3	
III	**Introduction to Drupal**		**13**	**20**
	9	Drupal - Features, Advantages and Disadvantages,	2	
	10	Comparison of Wordpress and Drupal	1	
	11	Installation and Configuration	2	
	12	Content types and Field	2	
	13	Drupal Architecture	2	
	14	User Management, Managing Comments	2	
	15	Creating and Customizing Themes	2	
IV	**Building Website**		**15**	**20**
	16	Website Development - Working with Templates and Template files	2	
	17	Articles, Creating Web Forms	2	
	18	Managing blocks, Add Links to Blocks, Moving Elements within Block	3	
	19	Blocks and Regions	2	
	20	Creating and Customizing Views	2	
	21	Installing and Configuring Modules	2	
	22	Static Pages, Creating Pages, Menu Management.	2	
V	**Hands-on Programming**		**30**	**30**
		1. Install Drupal on your local server and configure it to run. 2. Create a new content type called "Blog Post" with fields for title, body, and image. 3. Add a new field to the user profile for "Job Title" using Drupal's field management system. 4. Customise the default theme by changing the colours and		

		fonts. 5. Create a new custom theme from scratch and apply it to your Drupal site. 6. Add a new block to the sidebar displaying recent blog posts. 7. Create a custom view that displays a list of all users with their job titles. 8. Configure Drupal's built-in caching system to improve performance. 9. Install and configure a contributed module from Drupal.org to extend the functionality of your site. 10. Implement a custom module that adds a new feature to your Drupal site, such as a contact form or slideshow. 11. Set up user permissions to restrict access to certain parts of the site based on user roles. 12. Use the Drupal Views module to create a dynamic page that displays a grid of images from a specific content type. 13. Implement a responsive design for your Drupal site using CSS media queries. 14. Use Drupal's taxonomy system to categorise content and create a navigation menu based on taxonomy terms. 15. Test your site's accessibility using automated tools and make any necessary adjustments to improve accessibility for users with disabilities.		

Mapping of COs with PSOs and POs:

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PO1	PO2	PO3	PO4	PO5	PO6
CO 1	1	3	1	1	3	1						
CO 2	1	3	2	1	3	1						
CO 3	1	3	1	1	3	2						
CO 4	1	3	3	1	3	2						
CO 5	3	3	3	1	3	2						

CO 6	1	3	3	1	3	2						

Correlation Levels:

Level	Correlation
-	Nil
1	Slightly / Low
2	Moderate / Medium
3	Substantial / High

Assessment Rubrics:

- Quiz / Assignment/ Quiz/ Discussion / Seminar
- Midterm Exam
- Programming Assignments (20%)
- Final Exam (70%)

Mapping of COs to Assessment Rubrics:

	Internal Exam	Assignment	Practical Evaluation	End Semester Examinations
CO 1	✓	✓		✓
CO 2	✓	✓		✓
CO 3	✓	✓		✓
CO 4	✓	✓		✓
CO 5	✓	✓		✓
CO 6	✓	✓		✓

References:

1. Jennifer Campbell, Jennifer T Campbell, Web Design: Introductory, Course Technology.

2. Jason Beaird and Alex Walker, The Principles of Beautiful Web Design, SitePoint.

3. Bob Boiko, Content Management Bible, Wiley.

4. Daniel Sipos, Drupal 9 Module Development, Packt Publishing Limited

Programme	B. Sc. Computer Science				
Course Code	CSC1MN106				
Course Title	Computer Fundamentals with MS Excel,SPSS				
Type of Course	**Minor**				
Semester	I				
Academic Level	100-199				
Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	4	3	-	2	75
Pre-requisites	1. Fundamental Mathematics Concepts 2. Basic computer knowledge				
Course Summary	The course provides enough understanding of computer fundamentals, MS Excel, and SPSS. Students learn basic computing concepts, data entry, manipulation, and analysis in Excel and statistical analysis techniques using SPSS.				

Course Outcomes (CO):

CO	CO Statement	Cognitive Level*	Knowledge Category#	Evaluation Tools used
CO1	Understand fundamental concepts and skills essential for understanding and operating a computer system	U	C	Instructor-created exams / Seminar Presentation/ Instructor-created exams/ Quiz
CO2	Execute fundamental data input and manipulation tasks in MS Excel	C	P	Assignment / Instructor-created exams
CO3	Perform essential data input and manipulation activities within SPSS.	C	P	Assignment / Instructor-created exams

CO4	Implement Data analysis using SPSS	Ap	P	Hands-on practical sessions
CO5	Implement Data analysis using MS EXCEL	Ap	P	Hands-on practical sessions
* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C) # - Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge (M)				

Detailed Syllabus:

Module	Unit	Content	Hrs	Marks
I	**Introduction to computer system**		**12**	**19**
	1	Features, Limitations, Types	1	
	2	Number systems and character representation, Binary arithmetic	2	
	3	Basic components of computer -	2	
	4	Computer software types,Utility Program, Operating systems functions and types	2	
	5	Input and output devices ,Primary memory and secondary storage	2	
	6	Overview of Emerging Technologies: cloud computing, big data, data mining, mobile computing and embedded systems	2	
	7	Use of Computers in Education and Research: Data analysis, Heterogeneous storage, e-Library, Google Scholar, Domain specific packages such as SPSS, MATLAB, Mathematica etc	1	
II	**Introduction to Spread Sheet**		**10**	**18**
	7	MS Excel - Creating & Editing Worksheet, Formatting and Essential Operations	2	
	8	Formulas and Functions, Charts	2	
	9	Advanced features: Vlookup,Hlookup, Index, Address, Match, Offset, Transpose, Conditional Formatting, Data Sorting and Filtering	3	

	10	Pivot table & Pivot Chart	2	
	11	Linking and Consolidation	1	
III	**Introduction to SPSS**		**12**	**18**
	12	Features – Data View – Variable View – Output Viewer Window – Syntax Editor Window -	2	
	13	Open data file , Save , import from other data source ,data entry , labelling for dummy numbers	2	
	14	Recode in to same variable, Recode in to different variable, Transpose of data, Insert variables and cases	2	
	15	Merge variables and cases, Split, Select cases, Compute total scores	2	
	16	Table looks – Changing column - font style and sizes	2	
	17	Diagrammatic representation	2	
IV	**Data Analysis Using Ms Excel & SPSS**		**12**	**15**
	18	Estimation of mean, median and mode- Standard deviation and coefficient of variation.	3	
	19	Descriptive statistics, Parametric tests t-test (paired or unpaired), ANOVA (one-way- two way)	3	
	20	Pearson rank correlation, Linear regression	2	
	21	Non parametric tests: Mann Whitney U test, Wilcoxon signed rank test .	2	
	22	Kruskall Wallis test ,Chi- Square test5x	2	

V	**Hands-on Word Processor and Presentation Tool:** **Practical Applications, Case Study and Course Project**		**30**	
		EXCEL 1. Create a chart 2. Measures of Central Tendency & Descriptive Statistics 3. Parametric Tests T-Test 4. Correlation & Linear Regression 5. Chi- Square Test SPSS 6. Descriptive Statistics 7. Paired –Samples T Test 8. One-Way ANOVA 9. Correlation & Linear Regression 10. Chi- Square Test	20	
		Case study: Data analysis study on publically available biological data like bacterial growth analysis.		

Reference Books:

1. A. Goel, Computer Fundamentals, Pearson Education, 2010.
2. P. Aksoy, L. DeNardis, Introduction to Information Technology, Cengage Learning, 2006
3. P. K.Sinha, P. Sinha, Fundamentals of Computers, BPB Publishers, 2007
4. Excel Functions and Formulas Paperback by Bernd Held
5. Microsoft Excel 2010 Data Analysis and Business Modeling Paperback by Winsto
6. Jeremy J. Foster (2001). Data analysis using SPSS for windows. New edition. Versions 8-10. Sage publications. London.
7. Michael S. Louis – Beck (1995). Data analysis an introduction, Series: quantitative applications in 1

8. the social sciences. Sage, Publications. London.

Mapping of COs with PSOs and POs :

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PO1	PO2	PO3	PO4	PO5	PO6
CO 1	2	2	1	-	2	3						
CO 2	-	-	2	-	2	3						
CO 3	-	-	2	-	2	3						
CO 4	-	-	2	-	2	3						
CO 5	-	-	2	-	2	3						

Correlation Levels:

Level	**Correlation**
-	Nil
1	Slightly / Low
2	Moderate / Medium
3	Substantial / High

Assessment Rubrics:

- Quiz / Assignment/ Quiz/ Discussion / Seminar
- Midterm Exam
- Programming Assignments (20%)
- Final Exam (70%)

Mapping of COs to Assessment Rubrics :

	Internal Exam	Assignment	Project Evaluation	End Semester Examinations
CO 1	✓	✓		✓
CO 2	✓	✓		✓
CO 3	✓	✓		✓
CO 4	✓	✓	✓	✓
CO 5	✓	✓	✓	✓

Programme	B. Sc. Computer Science				
Course Code	CSC2MN106				
Course Title	Fundamentals of System Software, Networks and DBMS				
Type of Course	**Minor**				
Semester	II				
Academic Level	100-199				
Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	4	3	-	2	75
Pre-requisites	Basic understanding of computer operation Basic Science fundamentals				
Course Summary					

The course covers essential concepts in operating systems, network protocols, and database management systems, providing foundational knowledge for computer science and IT careers.

Course Outcomes (CO):

CO	CO Statement	Cognitive Level*	Knowledge Category#	Evaluation Tools used
CO1	Understand System Software principles	U	C	Instructor-created Exams / Assignment/ Viva Voce
CO2	Understand basic concepts of operating systems functions	U	C	Instructor-created Exams / Assignment/ Viva Voce
CO3	Interpret the concepts of data communications system and its components	An	C	Instructor-created Exams / Assignment/ Viva Voce

CO4	Acquire a good understanding of the architecture and functioning of Database Management Systems.	U	C	Instructor-created Exams / Assignment/ Viva Voce
CO5	Construct basic SQL queries to retrieve and manipulate data as required.	C	P	Practical/Exam/ Assignments
* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C) # - Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge (M)				

.

Detailed Syllabus:

Module	**Unit**	**Content**	**Hrs**	**Marks**
I			**11**	18
	1	Overview of System software and Application Software	1	
	2	System Software Components: operating systems, compilers, and device drivers.	1	
	3	Compilers: Classification of programming languages and language processors	2	
	4	Types of Operating System	3	
	5	Functions of Operating System	**3**	
II	**Computer networks**		**12**	18
	6	Goals of networking	1	
	7	network topologies	1	
	8	types of networks (LAN, MAN and WAN)	1	
	9	Communication Media-Guided (Twisted Pair, Coaxial Cable and Fiber Optic) and	2	
	10	Communication Media -Unguided (microwave, satellite)	2	
	11	Network OSI model- 7 layers	3	

	12	Internet Layer- 5 layers	2	
III	Database Management Systems		**12**	18
	13	Introduction to DB and DBMS:Definition	1	
	14	Advantages of DBMS	1	
	15	Three schema architecture of DBMS(External,Conceptual and internal)	1	
	16	Data Independence: Logical data independence and Physical data independence	2	
	17	Structure of Database Management System	2	
	18	Data models (Relational Model,Network Model. c. The Hierarchical Model. Object-Oriented Model)	5	
IV	**Structured query language - Create, insert, select, update, delete, alter, drop commands**		**10**	16
	19	DML	2	
	20	DDL	3	
	21	Constraints	2	
	22	Operators and functions	3	
V	**Practical Applications, Case Study and Course Project**		**30**	
	1	1. **Create Database**: ● Write a SQL query to create a new database in MySQL. 2. **Create Table**: ● Create tables with various data types for columns such as INT, VARCHAR, DATE, etc. ● Include constraints such as PRIMARY KEY, FOREIGN KEY, UNIQUE, NOT NULL, etc. 3. **Insert Data**: ● Insert records into tables using the INSERT INTO statement. ● Practice inserting data into tables with	20	

different data types.

4. **Retrieve Data**:
    - Write SELECT queries to retrieve data from tables.
    - Retrieve specific columns using SELECT.
    - Filter rows using the WHERE clause.
5. **Update Data**:
    - Update existing records in a table using the UPDATE statement.
    - Modify records based on specific conditions using the WHERE clause.
6. **Delete Data**:
    - Delete records from a table using the DELETE statement.
    - Remove records based on specific conditions using the WHERE clause.
7. **Sorting and Filtering**:
    - Sort the result set using ORDER BY clause.
    - Filter records using various conditions such as equality, comparison operators, and logical operators.
8. **Grouping and Aggregation**:
    - Group rows using GROUP BY clause.
    - Use aggregate functions like COUNT(), SUM(), AVG(), MIN(), and MAX().
9. **String Functions**:
    - Use string functions like CONCAT(), SUBSTRING(), UPPER(), LOWER(), etc.
    - Manipulate string data in SELECT queries.
10. **Date and Time Functions**:
    - Use date and time functions like DATE(),

		NOW(), YEAR(), MONTH(), DAY(), etc. ● Work with date and time data in SELECT queries. 11. **Mathematical Functions**: ● Use mathematical functions like ROUND(), CEIL(), FLOOR(), ABS(), etc. ● Perform mathematical operations on numeric data in SELECT queries. 12. **Conditional Functions**: ● Use conditional functions like IF(), CASE statement, etc. ● Implement conditional logic in SELECT queries.		
		Case study: 1. **Library Management System**: Track books, borrowers, and transactions, facilitating library operations efficiently. 2. **Student Information System**: Manage student records, courses, grades, and attendance for academic institutions. 3. **Employee Database System**: Store employee details, salaries, and performance evaluations, streamlining HR processes for companies.	10	

References

1 P. K Sinha, Fundamentals of Computers

2. D. M Dhamdhere,Operating System: A concept based Approach

3. Behrouz A Forouzan, Data Communication & Networking, MC Graw Hill

4. "Learning MySQL: Get a Handle on Your Data" by Seyed M.M. (Saied) Tahaghoghi and Hugh E. Williams.

Mapping of COs with PSOs and POs:

	PSO 1	PSO 2	PSO 3	PSO 4	PS O5	PS O6	PO 1	PO 2	PO 3	PO 4	PO 5	P O 6
CO 1	-	2	1	-	1	3						
CO 2	-	2	1	-	1	3						
CO 3	-	2	1	-	1	3						
CO 4	-	2	1	-	1	3						
CO 5	-	2	1	-	1	3						
CO 6	-	-	2	-	1	3						

Correlation Levels:

Level	**Correlation**
-	Nil
1	Slightly / Low
2	Moderate / Medium
3	Substantial / High

Assessment Rubrics:

- Quiz / Assignment/ Quiz/ Discussion / Seminar
- Midterm Exam
- Programming Assignments (20%)
- Final Exam (70%)

Mapping of COs to Assessment Rubrics:

	Internal Exam	Assignme nt	Project Evaluation	End Semester Examinations
CO 1	✓	✓		✓
CO 2	✓	✓		✓
CO 3	✓	✓		✓
CO 4	✓	✓		✓
CO 5	✓	✓		✓
CO 6	✓	✓		✓

Programme	B. Sc. Computer Science				
Course Code	CSC3MN206				
Course Title	Python Programming				
Type of Course	**Minor**				
Semester	III				
Academic Level	200-299				
Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	4	3	-	2	75
Pre-requisites	Have an understanding about algorithms and flowchart				
Course Summary	This course covers fundamentals of Python programming and teaches essential tools for data manipulation and analysis				

Course Outcomes (CO):

CO	CO Statement	Cognitiv e Level*	Knowledg e	Evaluation Tools used
CO1	Understand the basic concepts of Python programming	U	C	Instructor- created exams / Quiz
CO2	Apply problem- solving skills using different control structures and loops	Ap	P	Coding Assignments/ Code reading and review
CO3	Implement simple Python programs to solve basic computational problems and GUI	Ap	P	Coding Assignments/ exams

	applications			
CO4	Analyze the various data structures and operations on it using Python	An	P	Instructor-created exams / Case studies
CO5	Apply modular programming using functions	Ap	C	Instructor- created exams / Quiz
CO6	Identify the necessary Python packages in the domain and create simple programs with it	U, Ap	C, P	Coding
* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C) # - Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge (M)				

Detailed Syllabus:

Module	**Unit**	**Content**	**Hrs**	**Marks**
I	Problem solving strategies		**12**	15
	1	Problem analysis – formal definition of problem	1	
	2	Top- down design – breaking a problem into sub problems	2	
	3	Overview of the solution to the sub problems by writing step by step procedure (algorithm)	2	
	4	Repesentation of procedure by flowchart	1	
	5	Implementation of algorithms – use of procedures to achieve modularity.	2	

	6	Examples for algorithms and flow charts - At least ten problems Starting with non-numerical examples, and numeric problems like factorial, largest	4	
	Introduction to Python		**13**	19
	7	Variables, Data types	2	
	8	Expressions and Statements, Evaluation of Expressions	2	
II	9	Operators and Operands, Order of precedence, Boolean Expressions and logical operators, String Operations	2	
	10	Control statements,Conditional and alternative executions, Nested Conditionals, Recursion	2	
	11	Iteration - Multiple Assignment, While Statement	2	
	12	Tables, Two Dimensional Tables	2	
	13	Encapsulation and generalization, Local Variables	1	
	Introduction to NumPy		**12**	18
	14	The Basics of NumPy Arrays, Computation on NumPy Arrays: Universal Functions	3	
	15	Aggregations: Min, Max, and Everything in Between	2	
	16	Computation on Arrays:Broadcasting, Comparisons, Masks, and Boolean Logic.	2	
	17	Fancy Indexing, Sorting Arrays	2	
III	18	Structured Data: NumPy's Structured Arrays.	2	
	Functions		**8**	18
		Functions, Calling functions, Type conversion and coercion, composition of functions	2	

IV	19			
	20	Mathematical functions, User-defined Functions, Parameters and Arguments.	2	
	21	Strings and Lists – string traversal and comparison with List operations with Examples,	2	
	22	Tuples and dictionaries – Operations and Examples.	2	
	Hands-on Data Structures: **Practical Applications, Case Study and Course Project**		**30**	
Design programs from the concepts listed below. Select the topics and programs suited for your domain				
V	1	• Program to demonstrate basic data types in python • Program to demonstrate operators in python. • A cashier has currency notes of denominations 10, 50, and 100. If the amount to be withdrawn is input through the keyboard using input () function in hundreds, find the total number of currency notes of each denomination the cashier will have to give to the withdrawer. • Program to demonstrate list and tuple in python. • A library charges a fine for every book returned late. For first 5 days the fine is 50 paisa, for 6-10 days fine is one rupee and above 10 days fine is 5 rupees. If you return the book after 30 days your membership will be cancelled. • Write a program to accept the number of days the member is late to return the		

		book and display the fine or the appropriate message • Write a Program for checking whether the given number is an even number or not. • Write a Python program to print Fibonacci series. • Write function to compute gcd and lcm of two numbers. • Using a for loop, write a program that prints out the decimal equivalents of 1/2, 1/3, ¼. • Write a program to calculate overtime pay of 10 employees. Overtime is paid at the rate of • Rs.12.00 per hour for every hour worked above 40 hours. Assume that employee do not work for fractional part of an hour • Write a function reverse to reverse a list without using the reverse function.		
	2	**Case study(Examples):** ● Design a basic calculator application in Python that can perform addition, subtraction, multiplication, and division. ● Analysis of Antibiotic Resistance - Utilize publicly available datasets on antibiotic resistance in bacteria. Use NumPy to perform basic statistical analysis, such as calculating mean, median, and standard deviation of minimum inhibitory concentrations (MICs) for different antibioticss		

Reference Books:

1. Downey, A. et al., How to think like a Computer Scientist: Learning with Python, John Wiley,2015
2. Lambert K. A., Fundamentals of Python - First Programs, Cengage Learning India, 2015
3. Sprankle , M., Problem Solving & Programming Concepts, Pearson India

Mapping of COs with PSOs and POs :

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PO 1	PO2	PO3	PO4	PO5	PO6
CO 1	-	1	2	3	2	2						
CO 2	-	1	2	2	-	-						
CO 3	-	-	2	3	2	2						
CO 4	-	-	2	2	3	3						
CO 5	-	-	3	3	3	3						
CO 6	-	-	2	3	3	3						

Correlation Levels:

Level	**Correlation**
-	Nil
1	Slightly / Low
2	Moderate / Medium
3	Substantial / High

Assessment Rubrics:

- Quiz / Assignment/ Quiz/ Discussion / Seminar
- Midterm Exam
- Programming Assignments (20%)

- Final Exam (70%)

Mapping of COs to Assessment Rubrics :

	Internal Exam	Assignment	Project Evaluation	End Semester Examinations
CO 1	✓			✓
CO 2	✓	✓	✓	✓
CO 3	✓		✓	✓
CO 4	✓	✓	✓	✓
CO 5	✓			✓
CO 6	✓			✓

Programme	B. Sc. Computer Science				
Course Code	CSC1MN107				
Course Title	**Computer Hardware Assembly**				
Type of Course	**Minor**				
Semester	I				
Academic Level	100 - 199				
Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	4	3	-	2	75
Pre-requisites	Basic understanding of computer operation No previous experience in hardware assembly required				
Course Summary	Students will learn about the different components of a computer system, how they work together, and the skills necessary to assemble and maintain computer hardware effectively				

Course Outcomes (CO):

CO	CO Statement	Cognitive Level*	Knowledge Category#	Evaluation Tools used
CO1	A comprehensive understanding of fundamental concepts in Computer Organization and Hardware	U	C	Instructor-created exams / Assignment
CO2	Students will be able to understand and identify computer hardware components	Ap	P	Viva Voce
CO3	Students will be able to proficiently assemble computer hardware components adhering to industry standards and best practices.	C	P	Practical / Group Work
CO4	Students will learn to install and configure various operating systems (e.g., Windows, Linux) and drivers on	C	C	Practical / Group Work

	newly assembled computer systems			
CO5	Students will acquire the skills to diagnose and troubleshoot common hardware issues encountered during computer assembly	E	P	Practical/Exam/ Assignments
CO6	Students will develop the skills to perform hardware upgrades	C	P	Practical / Group Work
* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C) # - Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge (M)				

Detailed Syllabus:

Module	Unit	Content	Hrs	Marks
I	**Basic Computer Organization and Concept of Hardware**		**11**	**17**
	1	Basic Computer Organization: Input Unit, Storage Unit, Processing Unit, Control Unit, Output Unit	1	
	2	CPU Architecture: Arithmetic and Logic Unit, Control unit, Registers	1	
	3	Memory: Primary Memory, Secondary Memory	1	
	4	Access Time, Storage Capacity-bit, byte, nibble	1	
	5	Cache memory, Primary Memory- RAM (Static , Dynamic), ROM	2	
	6	Secondary Memory, storage devices (Magnetic tape, Hard disk , SSD and CD drive). Memory hierarchy.	**3**	
	7	Input and Output Devices	2	
II	**Hardware Components**		**12**	**18**
	8	Concept of Hardware and Software	1	
	9	Microprocessor, Clock Speed and Performance, Types of processors (Single core, dual core, multi core), GPU	2	
	10	Inside CPU: SMPS, Motherboard, Processor, Storage Devices (HDD, SSD), RAM (DDR2, DDR3, DDR4), ROM	2	
	11	Motherboard Components: Processor Slot, Cooling Fan, RAM, Expansion Slots (PCIe), Mouse and Keyboard Ports, Chipset, BIOS/UEFI Chip, SATA/NVMe Slots, Network Interface, Ports- Ethernet, VGA port, HDMI port, USB port	3	

	12	Cables and Connectors,	2	
	12	Expansion Cards: Graphics card, Sound Card, Network Interface Card	1	
III	**Hardware Assembling**		**10**	**17**
	14	**Safety and Tools**: Introduction to ESD (Electrostatic Discharge) safety, use of antistatic wrist straps, and essential tools for assembling hardware	1	
	15	**Assembling a PC**: Step-by-step guide on assembling a PC, including installing processor onto the motherboard, Setup Cooling Fan, Install RAM, Install other expansion cards, Mounting the motherboard into the case and Install storage devices (HDD, SSD) into drive bays, Install the GPU into the appropriate PCIe slot (if not integrated into the CPU)	4	
	16	**Cable Management**: Best practices for managing cables within a PC case to ensure optimal airflow and aesthetics. Connect power supply cables to the motherboard, CPU, GPU, and storage devices, Connect case cables (power switch, reset switch, LEDs, USB ports) to the appropriate headers on the motherboard,	3	
IV	**System Configuration, OS Installation, trouble Shooting**		**12**	**18**
	17	**BIOS and UEFI**: Understanding the roles of BIOS and UEFI navigating BIOS settings, and configuring hardware.	2	
	18	**Installing and Configuring Operating Systems**: Guidelines for installing operating systems (Windows, Linux)	2	
	19	**Installing Drivers**: install drivers for motherboard components (Chipset, LAN, Audio), GPU, and other peripherals.	2	
	20	**Hardware Upgrades**: How to upgrade components such as RAM, storage, and GPUs, including compatibility considerations.	2	
	21	**Troubleshooting Common Assembly Issues**: Identifying and resolving common issues encountered during PC assembly	2	
	22	**Diagnostics and Maintenance:** Introduction to methods, tools and software used for diagnosing hardware issues.	2	
V	**Hands-on Hardware Assembling** **Practical Applications, Case Study and Course Project**		**30**	

	1	**1: Identifying Computer Components** ● Identify and describe the function of the CPU, RAM, motherboard, PSU, storage devices, and peripheral connectors. ● Disassemble and reassemble a desktop computer, identifying each component as it is removed and replaced. **2: Building a PC from Scratch** ● Use appropriate tools and safety equipment to assemble a computer, including installing the motherboard, CPU, CPU cooler, RAM, and storage. ● Practice cable management to ensure a neat and efficient build. ● Document each step of the assembly process for future reference and learning. **3: Operating System Installation and Configuration** ● Install a chosen operating system (e.g., Windows, Linux) from a bootable USB drive or DVD. ● Install essential drivers and software updates. ● Configure basic settings (user accounts, network settings, display resolution). **Case Study: Trouble Shooting and Maintenance**	25 5	

References

1. Pradeep K. Sinha and Priti Sinha, Computer Fundamentals: Concepts, Systems & Applications. BPB Publications.
2. Bigelow's Troubleshooting, Maintaining & Repairing PCs Hardcover – by Stephen Bigelow
3. Kevin Wilson, Computer Hardware: The Illustrated Guide to Understanding Computer Hardware. Amazon Digital Services LLC – KDP, 2018.

Mapping of COs with PSOs and POs:

	PSO1	PSO2	PSO3	PSO4	PS O5	PSO6	PO1	PO2	PO3	PO4	PO5	PO6
CO 1	1	2	-	-	-	-						
CO 2	-	2	-	-	-	-						
CO 3	-	2	-	-	1	-						
CO 4	-	2	-	-	-	-						
CO 5	-	2	-	-	1	-						
CO 6	-	2	-	-	1	-						

Correlation Levels:

Leve l	Correlation
-	Nil
1	Slightly / Low
2	Moderate / Medium
3	Substantial / High

Assessment Rubrics:

- Quiz / Assignment/ Quiz/ Discussion / Seminar
- Midterm Exam
- Programming Assignments (20%)
- Final Exam (70%)

Mapping of COs to Assessment Rubrics:

	Internal Exam	Assignment	Project Evaluation	End Semester Examinations
CO 1	✓	✓		✓

CO 2	✓	✓		✓
CO 3	✓	✓		✓
CO 4	✓	✓		✓
CO 5	✓	✓		✓
CO 6	✓	✓		✓

Programme	B. Sc. Computer Science				
Course Code	CSC2MN107				
Course Title	**Exploring Cyber security in social media**				
Type of Course	**Minor**				
Semester	II				
Academic Level	100-199				
Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	4	3	-	2	75
Pre-requisites	1. Fundamentals of Computer science				
Course Summary	Students can investigate the complex interplay between social media and cyber security with this minor programme. The course will explore the different risks, vulnerabilities, and dangers related to social media platforms, providing participants with valuable knowledge on how to safeguard both individuals and organisations. Students will get a thorough grasp of cyber security principles as they relate to social media through a combination of academic study, hands-on activities, and case analysis.				

Course Outcomes (CO):

CO	CO Statement	Cognitive Level*	Knowledge Category#	Evaluation Tools used
CO1	Understand the idea of cyber security as well as the problems and difficulties that surround it.	U	F	Instructor-created exams / Quiz
CO2	Understand the cyber crimes, their nature, legal remedies and as to how report the crimes through available platforms and procedures	U	C	Practical Assignment / Observation of Practical Skills
CO3	Understand the privacy and security issues associated with using online social media. They should also be aware of the best practices for using social media platforms, the legal ramifications, and how to report incorrect content.	U	F	Seminar Presentation / Group Tutorial Work/ Viva Voce

CO4	Understand ethical standards related to usage of social media and apply those ethical standards in their day today life usage.	U	C	Instructor-created exams / Home Assignments
CO5	Comprehend the fundamentals of computer and mobile security and will be able to safeguard their gadgets with simple tools and technology.	Ap	P	Writing assignments/ Instructor-created exams/ practicals
CO6	Develop a cybersecurity plan for a hypothetical social media.	Ap	P	Case Study/ mini Project/ practicals
* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C) # - Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge (M)				

Detailed Syllabus:

Module	**Unit**	**Content**	**Hrs (45+30)**	**Marks (70)**
I	**Introduction to Cyber security & History of Internet and social media**		**9**	**12**
	1	Historical overview of social media development	2	
	2	Impact of social media	1	
	3	Internet, World wide web, Introduction of the internet	2	
	4	Internet infrastructure for data transfer and governance	2	
	5	Terminologies like anti-virus, firewall, Wi-Fi network	2	
II	**Introduction Cyber Security & reporting of cyber crimes**		**12**	**15**
	6	Concept of cyber security, Issues and challenges of cyber security Terminologies: Cyber Security, Cyber Crime, Cyber Attack, Cyber Espionage, Cyber Warfare	2	
	7	Classification of cyber crimes : Financial crimes: Online fraud, phishing, identity theft (basic concepts only)	2	
	8	Cyber crime targeting computers and mobiles	2	
	9	Cyber crime against women and children, social engineering attacks, malware and ransomware attacks	2	
	10	Reporting of cyber crimes,	2	

	11	Legal perspective of cyber crime, IT Act 2000 and its amendments, Cyber crime and offences	2	
III		**Introduction to Social Media**	**12**	**15**
	12	Introduction to Social networks. Types of Social media, Social media platforms	2	
	13	Social media monitoring, Hashtag, Viral content	3	
	14	Social media marketing	2	
	15	Social media privacy, Challenges, opportunities and pitfalls in online social network	2	
	16	Security issues related to social media: Phishing Attacks, Account take over, Data breeches, Fake Accounts and Impersonation, Credential Stuffing, Doxing (concepts only)	3	
IV		**Cyber Security in social media**	**12**	**20**
	17	End Point device and Mobile phone security, Password policy	1	
	18	Data backup, Downloading and management of third party software	2	
	19	Cyber Security best practices, Significance of host firewall and Ant-virus, Management of host firewall and Anti-virus,	2	
	20	Wi-Fi security, Configuration of basic security policy and permissions.	1	
	21	Terminologies like- strong password, Two-Factor Authentication, Login Activity Monitoring, Authorized Devices	3	
	22	Ethical dilemmas in social media usage: Privacy vs. Transparency, Authenticity vs. Self-Presentation, Misinformation vs. Truthfulness, Cyberbullying and Online Harassment, Data Privacy and User Consent, Influence and Manipulation, Addiction & Mental Health	3	
V		**Practical Implementations of Cyber security in social media**	**30**	**20**
	1	● Setting, configuring and managing three password policy in the computer (BIOS, Administrator and Standard User). ● Setting and configuring two factor authentication in the Mobile phone. ● Security patch management and updates in Computer and Mobiles. ● Managing Application permissions in Mobile phone. ● Installation and configuration of computer Anti-virus. ● Installation and configuration of Computer Host Firewall.	20	

		● Wi-Fi security management in computer and mobile. ● Hands-on exercises with social media monitoring tools		
	2	Develop a cybersecurity plan for a hypothetical social media scenario (Capstone) Organisations dealing with Cyber crime and Cyber security in India, Case studies.	10	

References

- "Social Media Security: Leveraging Social Networking While Mitigating Risk" by Michael Cross
- "The Social Media Security Playbook: Your Guide to Stopping Threats, Plugging Gaps, and Responding to Emergencies" by Christopher Hadnagy and Michele Fincher
- Cyber Security Understanding Cyber Crimes, Computer Forensics and Legal Perspectives by Sumit Belapure and Nina Godbole, Wiley India Pvt. Ltd. (First Edition, 2011)
- Security in the Digital Age: Social Media Security Threats and Vulnerabilities by Henry A. Oliver, Create Space Independent Publishing Platform. (Pearson , 13th November, 2001)
- Fundamentals of Network Security by E. Maiwald, McGraw Hill.

Mapping of COs with PSOs and POs :

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PO1	PO2	PO3	PO4	PO5	PO6
CO 1	-	2	-	-	-	-						
CO 2	-	1	-	-	-	-						
CO 3	-	2	-	-	-	-						
CO 4	-	2	-	-	-	-						
CO 5	-	1	-	-	-	-						
CO 6	-	1	1	-	-	-						

Correlation Levels:

Level	Correlation
-	Nil
1	Slightly / Low
2	Moderate / Medium
3	Substantial / High

Assessment Rubrics:

- Quiz / Assignment/ Quiz/ Discussion / Seminar
- Midterm Exam
- Programming Assignments (20%)
- Final Exam (70%)

Mapping of COs to Assessment Rubrics :

	Internal Exam	Assignment	Practical Evaluation	End Semester Examinations
CO 1	✓	✓	✓	✓
CO 2	✓	✓	✓	✓
CO 3	✓	✓	✓	✓
CO 4	✓	✓	✓	✓
CO 5	✓	✓	✓	✓
CO 6	✓	✓	✓	

Programme	B. Sc. Computer Science Minor				
Course Code	CSC3MN107				
Course Title	**Emerging Trends in Computer Science**				
Type of Course	**Minor**				
Semester	III				
Academic Level	200-299				
Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	4	3	-	2	75
Pre-requisites	1. Knowledge in Computers. 2. Basic knowledge in Internet				
Course Summary	This course provides an overview of the latest trends and advancements in the field of computer science. Students will explore emerging technologies, methodologies, and research areas shaping the future of computing.				

Course Outcomes (CO):

CO	CO Statement	Cognitive Level*	Knowledge Category#	Evaluation Tools used
		U	C	
CO1	Analyze real-world use cases and applications of emerging technologies, identifying opportunities and challenges for innovation and problem-solving in areas such as healthcare, finance, smart cities, and industry.	An	C	
CO2	Understand the fundamental concepts of artificial intelligence (AI), and applications across various domains.	U	C	
CO3	Identify the key components of a block chain network, such as nodes, blocks, transactions, and smart contracts.	Ap	C	

CO4	Understand the fundamental concepts of computer networks and popular applications	U	C	
CO5	Identify the key components of a block chain network and applications	Ap	C	
CO6	Understand the evolution of database management systems (DBMS) from traditional modern .	U	C	
CO7	Describe the features of NoSQL databases and their advantages over traditional relational databases.	Ap	C	
* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C) # - Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge (M)				

Detailed Syllabus:

References:

Module	Unit	Content	Hrs	Marks(70)
I	**Basic concepts of Artificial Intelligence and Machine Learning**		12	**20**
	1	Concept of Machine Learning and Artificial Intelligence: Definition, Evolution	2	
	2	Types Of Machine Learning: Supervised learning, Unsupervised learning, Reinforcement learning, Evolutionary learning	3	
	3	Common ML algorithms: Regression, Classification, Clustering. (Concepts)	3	
	4	The Machine Learning Process: Data Collection and Preparation, Feature Selection , Algorithm Choice , Parameter and Model Selection , Training , Evaluation	2	
	5	Application of Machine Learning: Healthcare, Finance, Self Driving Cars, Robotics	2	
II	**Introduction to Block chain Technology:**		**12**	**20**

	6	Cryptography Overview: Definition, Types of Cryptography -Public and Private Keys, Application -Digital Signature	3	
	7	Introduction to Block chain Technology: History of Block chain, Generic Elements of Block chain, Features of Block chain(Decentralization, Transparency , Immutability and Security)	2	
	8	Types of Block chain:	2	
	9	Applications of Bock chain Technology : Financial Services, Supply Chain Management, Smart Contracts	2	
	10	Crypto currencies: Definition, Bit coin, Ethereum	2	
	11	Challenges in Block chain Adoption: Scalability, interoperability, and regulatory concerns, Security considerations.	1	
III		**IOT and Cloud Technology**	**10**	**15**
	12	Overview of Computer Networks: Definition, types, and importance.	2	
	13	Network Design Concepts: ISO/OSI and TCP/IP	2	
	14	Networking Devices and Protocols: Routers, switches, Hub, Modems, TCP, UDP, IP.	2	
	15	Cloud Computing and Services: Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS).	2	
	16	Internet of Things: Definition, Key characteristic, Architecture and components, Challenges and Security in IOT.	2	
IV		**Unstructured Database**	**10**	**15**
	17	Overview of traditional DBMS: Relational, Object-Oriented Database	3	
	18	Structured and Unstructured Database	1	
	19	Introduction to NoSQL databases	2	
	20	Types of NoSQL databases (Document-oriented, Key-value stores, Column-family stores, Graph databases)		
	21	Cloud-Based Database Services: Database as a Service (DBaaS) overview, Benefits of cloud-based solutions	2	

	22	Block chain Databases (2 hours): Understanding graph databases and their applications, Overview of Block chain databases and their role in data integrity.	2	
		Practical Applications, Case Study	**30**	
	1	1. Identify the various software platforms for AI programming.		
		2. Identify the various platforms used for No-Code AI.(Google Cloud Auto ML, Microsoft Azure Al Buildeer, IBM Watson Studio etc..)		
		3. Use chatbot platforms like ChatGPT or any other to engage in conversational interactions and understand how natural language processing works.		
		4.Use online tools or any applications that demonstrate image recognition capabilities.		
		5.Use online platforms or software that provide interactive AI demos and simulations, such as neural network visualisers or AI-powered character generators.		
		6.Use an online tool like CyberChef or an online RSA key generator to generate a pair of RSA public and private keys.		
		7. Create a digital signature for a given document using an online service like DocuSign or HelloSign, and verify it using the service's verification features.		
		8. Use online resources to create a comparison table for public, private, and consortium blockchains, including real-world examples.		
		9. Use MySQL or PostgreSQL to create a database, define tables, and perform CRUD operations.		
		10.Use db4o (Database for Objects) or an equivalent tool to create and manipulate an object-oriented database.		
		11. Use MongoDB to store and query unstructured data.		
		12. Use MongoDB to create a collection and perform CRUD operations.		
		13. Deploy and interact with a cloud-based database service.		

		14. Explore and implement a basic blockchain database using an appropriate platform.		
		15. Use BigchainDB or a similar blockchain database platform to create a blockchain database.		

1. "Explorations in Artificial Intelligence and Machine Learning" - By Roberto Zicari
2. "Blockchain Fundamentals"- Dr. Ravindhar Vadapalli
3. "Cloud Computing: Concepts, Technology & Architecture" by Thomas Erl, Ricardo Puttini, and Zaigham Mahmood
4. Michael Miller, "The Internet of Things: How Smart TVs, Smart Cars, Smart Homes, and Smart Cities Are Changing the World", Pearson Education 2015
5. "A Brief Guide to the Emerging World of Polyglot Persistence:- By Pramod J. Sadalage, Pramod Sadalage, Martin Fowler

Mapping of COs with PSOs and POs :

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PO1	PO2	PO3	PO4	PO5	PO6
CO 1	-	-	1	-	-	3						
CO 2	-	2	1	-	2	3						
CO 3	-	2	1	-	2	3						
CO 4	-	2	1	-	2	3						
CO 5	-	2	1	-	2	3						
CO 6	-	2	1	-	2	3						
CO7	-	-	1	-	-	3						

Correlation Levels:

Level	Correlation
-	Nil
1	Slightly / Low
2	Moderate /

	Medium
3	Substantial / High

Assessment Rubrics:

- Quiz / Assignment/ Quiz/ Discussion / Seminar
- Midterm Exam
- Programming Assignments (20%)
- Final Exam (70%)

Mapping of COs to Assessment Rubrics :

	Internal Exam	Assignment	Project Evaluation	End Semester Examinations
CO 1	✓			✓
CO 2	✓			✓
CO 3	✓	✓		✓
CO 4		✓		✓
CO 5		✓		✓

VOCATIONAL MINOR

Programme	BSc Computer Science				
Course Code	CSC1VN101				
Course Title	Computational Mathematics in Data Science				
Type of Course	Vocational Minor				
Semester	I				
Academic Level	100-199				
Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	4	3	-	2	75
Pre-requisites	Basic Mathematics is required (Algebra, Arithmetic)				
Course Summary	This course provides a fundamental exploration of mathematical concepts essential for computer science. Students will explore into key topics including Linear Algebra, Differential and Integral Calculus. The course aims to equip students with the mathematical tools and reasoning skills necessary for creating and analyzing algorithms, understanding and solving computational problems in various areas of computer science Data science, Artificial Intelligence.				

Course Outcomes (CO):

CO	CO Statement	Cognitive Level*	Knowledge Category#	Evaluation Tools used
CO1	Reflect the concept of matrices and determinants as a way to depict and streamline mathematical ideas to perform basic operations.	U	C	Instructor- created exams / Quiz/Assignment/ Seminar
CO2	Able to find the inverse of square matrices using different methods and demonstrate a solid understanding of eigen values.	U	C	Instructor-created exams/ Quiz/Assignment/ Seminar
CO3	Proficiency in solving linear equations using different techniques and understanding the geometric interpretation of solutions.	U	C	Instructor-created exams/ Quiz/Assignment/ Seminar
CO4	Gain proficiency in representing vectors geometrically and algebraically, understanding vector addition, dot and cross products.	U	C	Instructor-created exams/ Quiz/Assignment/ Seminar

CO5	Able to apply differential and integral calculus to various functions encountered in data science such as polynomials, exponentials and logarithmic functions.	U	C	Instructor-created exams/ Quiz/Assignment/ Seminar
CO6	Represent various mathematical problems using algorithmic approaches and enhance problem-solving skills by visualizing solutions through the utilization of software tools.	U, Ap	C, P	Practical Assignment / Observation of Practical Skills
* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C) # - Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge (M)				

Detailed Syllabus

Module	Unit	Contents	Hrs (45+30)	Mark
I		**Matrices and Determinants**	**13**	**18**
	1	Matrices: Definition, Order of a matrix, Types of matrices	1	
	2	Operations on matrices: Addition, Subtraction, Multiplication	3	
	3	Properties of matrix: Various kind of Matrices, Transpose of a matrix	2	
	4	Elementary Transformations of Matrices and Rank of Matrices	2	
	5	Symmetric and Skew Symmetric Matrices	2	
	6	Determinants, Minors, Cofactors, Inverse of a matrix	3	
II		**Linear Algebra and Vector Calculus**	**11**	**18**
	7	Linear Independence: Characteristic equations,	1	
	8	Eigen values, Eigen Vector	2	
	9	Solving system of linear equations: Gauss Elimination Method, Gauss Jordan method, Gauss Siedel Methods	3	
	10	Vectors: Definition Magnitude of a vector, Types of Vectors, Vector addition	2	
	11	Dot products and Cross products	2	
	12	Vectors in 2- and 3-space	1	
III		**Differentiation**	**10**	**17**
	13	Limits; Definition (concept only), Derivative of a Point, Derivative at Function	2	

	14	Differentiation: Definition, Differentiation from first principle, Differentiation of important function	2	
	15	Product rule, Quotient rule	3	
	16	Derivative of function of a function	2	
	17	Logarithmic differentiation	1	
IV		**Integration**	**11**	**17**
	18	Integration: Integral as Anti-derivative, Indefinite integral & constant of integration	2	
	19	Fundamental theorems, Elementary Standard results	2	
	20	Integral of different functions, Integration by Substitution	3	
	21	Definite Integrals, Properties of definite integrals	2	
	22	Evaluation of Definite Integrals by Substitution	2	
V		**Lab Activities (Use Sci Lab or any other Alternative tools)**	**30**	
		1. Create and display a m x n order matrix. 2. Perform addition of two matrices. 3. Perform multiplication of two matrices. 4. Find the Determinant of a n x n matrix. 5. Read and display a polynomial of degree n. 6. Find the dot product of two given vectors. 7. Find the cross product of two given vectors. 8. Find the eigen values of a n x n matrix. 9. Find the derivative of a polynomial with degree n. 10. Find the integral of a polynomial with degree n having limits a and b.	30	

Mapping of COs with PSOs and POs:

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PO1	PO2	PO3	PO4	PO5	PO6
CO 1	1	-	1	1	-	-						
CO 2	2	-	2	2	-	-						
CO 3	2	-	2	2	-	-						
CO 4	2	-	2	2	-	-						
CO 5	2	-	2	2	-	-						
CO 6	2	-	2	2	-	-						

Correlation Levels:

Level	Correlation
-	Nil
1	Slightly / Low
2	Moderate / Medium
3	Substantial / High

Assessment Rubrics:

- Quiz / Assignment/ Discussion / Seminar
- Midterm Exam
- Programming Assignments (20%)
- Final Exam (70%)

Mapping of COs to Assessment Rubrics:

	Internal Exam	Assignment	Practical Evaluation	End Semester Examinations
CO 1	✓	✓		✓
CO 2	✓	✓		✓
CO 3	✓	✓		✓
CO 4	✓	✓		✓
CO 5	✓	✓		✓
CO 6	✓	✓	✓	✓

References:

1. Advanced Engineering Mathematics, Erwin Kreyszig, Wiley
2. Higher Engineering Mathematics, John Bird, Elsevier Direct
3. Skills in Mathematics: Algebra, S.K.Goyal
4. Higher Engineering Mathematics, B S Grewal, Khanna Publishers
5. Higher Engineering Mathematics, Ramana, Tata McGraw Hill
6. Engineering Mathematics, P Kandasamy, S. Chand Group
7. Gilbert Strang, “Introduction to Linear Algebra”, Wellesley-Cambridge Press, 2023.

.

Programme	BSc Computer Science				
Course Code	CSC2VN101				
Course Title	Introduction to Data Science				
Type of Course	Vocational Minor				
Semester	II				
Academic Level	100-199				
Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	4	3	-	2	75
Pre-requisites	1. Basic understanding of computer science concepts. 2. Familiarity with data handling. 3. simple mathematical analysis.				
Course Summary	Data science is the domain of study that deals with vast volumes of data using modern tools and techniques to find unseen patterns, derive meaningful information, and make business decisions.				

Course Outcomes (CO):

CO	CO Statement	Cognitive Level*	Knowledge Category#	Evaluation Tools used
CO1	Identify the relevance and applications of computers in other disciplines with various data science applications.	R	C	Assignment / Instructor-created exams / Quiz
CO2	understanding of data science concepts and be capable of applying data science skills and interpret data science results	U	C	Assignment / Instructor-created exams / Quiz
CO3	Acquire logical thinking about evolution of data science	U	C	Assignment / Instructor-created exams / Quiz
CO4	How to use tools for acquiring, cleaning, analyzing, exploring, and visualizing data	Ap	P	Assignment / Instructor-created exams / Quiz
CO5	Learn to make data-driven inferences and decisions	Ap	P	Assignment / Instructor-created exams / Quiz
CO6	Able to perform data science processing, such as data import, data analysis, data visualization, and data modelling	Ap	P	Assignment / Instructor-created exams / Quiz

* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C) # - Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge (M)

Detailed Syllabus

Module	**Unit**	**Content**	**Hrs (45+30)**	**Mark**
I	**Introduction to Data Science**		**10**	**15**
	1	Introduction to Data Science-Definition	2	
	2	Evolution of Data Science	2	
	3	Data Science Roles	3	
	4	Application of data sciences.	3	
II	**Data Collection and Data Pre-Processing**		**10**	**15**
	5	Data Collection Strategies	1	
	6	Data Pre-Processing Overview	2	
	7	Data Cleaning	2	
	8	Data Integration and Transformation	3	
	9	Data Reduction and Descretization	2	
III	**Data Analytics**		**12**	**20**
	10	Descriptive Statistics	2	
	11	Mean, Standard Deviation	2	
	12	Skewness and Kurtosis	2	
	13	Box Plots	2	
	14	Pivot Table	2	
	15	Correlation Statistics	2	
IV	**Data Model Devolopment and Evaluation**		**13**	**20**
	16	Simple and Multiple Regression	2	
	17	Model Evaluation using Visualization	2	
	18	Residual plot and distributional plot	2	
	19	Prediction and Decision Making	2	
	20	Model Evaluation techniques-	3	
	21	Supervised learning techniques	1	
	22	unsupervised learning techniques	1	
V	**Practical: Introduction to data analysis tools in Python** • Working with Pandas data frames • Basic plots using Matplotlib • Frequency distributions • Averages • Correlation and scatter plots • Correlation coefficient • Regression		**30**	

Mapping of COs with PSOs and POs:

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PO1	PO2	PO3	PO4	PO5	PO6
CO 1	3	-	1	-	2	-						
CO 2	3	-	1	-	1	-						
CO 3	3	-	2	-	1	-						
CO 4	2	-	2	-	2	-						
CO 5	1	-	2	-	2	-						
CO 6	1	-	2	1	2	-						

Correlation Levels:

Level	Correlation
-	Nil
1	Slightly / Low
2	Moderate / Medium
3	Substantial / High

Assessment Rubrics:

- Quiz / Assignment/ Quiz/ Discussion / Seminar
- Midterm Exam
- Programming Assignments (20%)
- Final Exam (70%)

Mapping of COs to Assessment Rubrics:

	Internal Exam	Assignment	Practical Evaluation	End Semester Examinations
CO 1	✓	✓		✓
CO 2	✓	✓		✓

CO 3		✓	✓	✓
CO 4		✓	✓	✓
CO 5		✓	✓	✓
CO 6	✓	✓		✓

Text books:

1. Jojo Moolayil, “Smarter Decisions : The Intersection of IoT and Data Science”, PACKT, 2016.
2. Cathy O’Neil and Rachel Schutt , “Doing Data Science”, O'Reilly, 2015.
3. David Dietrich, Barry Heller, Beibei Yang, “Data Science and Big data Analytics”, EMC 2013
4. Introduction to Data Science a Python approach to concepts, Techniques and Applications, Igual, L;Seghi’, S. Springer, ISBN:978-3-319-50016-4 2.
5. Data Analysis with Python A Modern Approach, David Taieb, Packt Publishing, ISBN-9781789950069

Programme	B. Sc. Computer Science				
Course Code	CSC3VN201				
Course Title	Data Analysis and Visualisation Using Spreadsheets				
Type of Course	Vocational Minor				
Semester	III				
Academic Level	200-299				
Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	4	3	-	2	75
Pre-requisites	• Basic understanding of computers • Familiarity with basic mathematical operations				
Course Summary	This course provides a comprehensive introduction to Spreadsheets, focusing on understanding formulas, functions, data organization, analysis techniques, and data visualization. Participants will gain skills in spreadsheet management, data cleansing, analysis, and visualization using Excel's various tools and features.				

Course Outcomes (CO):

CO	CO Statement	Cognitive Level*	Knowledge Category#	Evaluation Tools used
CO1	Students will demonstrate proficiency in managing spreadsheets, including creating, formatting, and manipulating data within Excel workbooks. They will be able to effectively navigate Excel's interface and utilize toolbars.	U	P	Instructor-created exams / Quiz
CO2	Learners will understand the importance of data organization and cleansing in Excel. They will be able to import, export, filter, sort, validate, and remove duplicates from datasets. Students will develop skills to ensure data integrity and consistency, enhancing their ability to work with clean and organized data sets.	U	P	Instructor-created exams/ Home Assignments
CO3	Participants will acquire advanced data analysis skills like pivot tables, what-if analysis, and goal seek. They will be able to apply various Excel functions and tools to perform complex calculations, analyze trends, and make informed	Ap	P	Instructor-created exams

	decisions based on data analysis.			
CO4	Students will gain proficiency in data visualization techniques using Excel. They will be able to create a variety of charts, design pivot charts, dashboards for effective data analysis. Additionally, learners will be able to implement form controls for interactive data manipulation in their visualizations.	Ap	P	Instructor-created exams
CO5	Learners will develop skills in advanced features of Excel like macros, protect data sheets and workbooks, utilize split, freeze, and hide options effectively, incorporate add-ins for extended functionalities, and manage printing options in Excel for professional presentation of data.	Ap	P	Instructor-created exams
* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C) # - Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge (M)				

Detailed Syllabus:

Module	Unit	Content	Hrs (45)	Marks (70)
I		**Introduction to Excel & Understanding Formulas, Functions**	**10**	**15**
	1	Features of Spreadsheet	1	
	2	Parts of Excel Window, Toolbars, Worksheet and Workbook, Insertion and Deletion of cells, columns, rows	2	
	3	Formatting in Excel (Merge, Warp, Font Formatting, Number Formatting, Borders and Shading, Colouring)	2	
	4	Range, Autofill, Autosum, Relative, Absolute and Mixed Referencing in Excel, Linking data between worksheets	2	
	5	Formulas and Functions in Excel: Use of Formula Bar, Functions: SUM, ROUND, CEIL, FLOOR, IF, AND, OR, AVERAGE, MIN, MAX ,COUNT, COUNTIF, SUMIF, VLOOKUP,HLOOKUP	3	
II		**Cleansing and Organising Data in Excel**	**10**	**10**
	6	Importance of Data Cleansing and Organisation	1	
	7	Data Import and Export	2	
	8	Filtering and Sorting	2	
	9	Data Validation and removal of duplicates	2	
	10	Group, Ungroup, Subtotal	2	
	11	Conditional Formatting – Highlight Cell Rules, Top/Bottom Rules	1	
III		**Advanced Techniques for Data Analysis**	**14**	**10**
	12	Features of Pivot table	1	

	13	Pivot Table creation	2	
	14	Fitting Linear regression in Excel	3	
	15	Linear regression using Excel formulas	3	
	16	Interpreting regression results	2	
IV	**Data Visualisation Techniques**		**14**	**15**
	17	Creating Charts, Different types of charts	2	
	18	Formatting Chart Objects, Changing the Chart Type, Showing and Hiding the Legend, Showing and Hiding the Data Table	2	
	19	Creating charts from regression	2	
	20	Pivot Chart	2	
	21	Dashboards	2	
	22	Form Controls	4	
V	**Hands-on Spreadsheets**		**30**	**30**
	Spreadsheet Basics: 1. Create a new workbook in Excel. 2. Identify and label different parts of the Excel window, such as the Ribbon, Formula Bar, Name Box, and Worksheet Tabs. 3. Insert and delete cells, columns, and rows within a worksheet.			
	Formatting: 4. Merge cells and wrap text within merged cells. 5. Experiment with different font styles, sizes, and colors for text formatting. 6. Apply various number formatting options (e.g., currency, percentage, date) to cells. 7. Add borders and shading to cells or ranges.			
	Range Operations: 8. Use Autofill to quickly populate a series of cells with data (e.g., numbers, dates, text). 9. Utilize Autosum to calculate the sum of a range of numbers automatically. 10. Practice relative, absolute, and mixed referencing in formulas to understand their impact on cell references. 11. Link data between different worksheets within the same workbook.			
	Formulas and Functions: 12. Experiment with different mathematical formulas (e.g., addition, subtraction, multiplication, division) using the Formula Bar. 13. Apply common functions such as SUM, ROUND, CEIL, FLOOR, IF, AND, OR, AVERAGE, MIN, MAX, COUNT, COUNTIF, SUMIF, VLOOKUP, and HLOOKUP to solve specific problems or analyze data sets. 14. Combine functions within formulas to perform more complex calculations.			
	Data Import and Export: 15. Import external data from sources such as CSV files, text files, or databases into Excel. 16. Export Excel data to different formats (e.g., CSV, PDF) for sharing or further analysis.			
	Filtering and Sorting: 17. Filter data to display specific records based on criteria (e.g., dates,			

	categories, numerical ranges). 18. Sort data alphabetically, numerically, or chronologically to analyze trends or identify patterns.		
	Data Validation and Removal of Duplicates: 19. Implement data validation rules to restrict input values within specified criteria (e.g., date ranges, numerical limits, list selections). 20. Identify and remove duplicate records from a dataset while preserving unique data entries.		
	Grouping, Ungrouping, and Subtotal: 21. Group related rows or columns together to organize data hierarchically. 22. Perform subtotal calculations within grouped data to summarize information at different levels of detail.		
	Conditional Formatting: 23. Apply conditional formatting using highlight cell rules to visually identify data outliers, trends, or exceptions. 24. Utilize top/bottom rules to highlight top or bottom values within a dataset for quick analysis.		
	Pivot Table Creation: 25. Import a dataset into Excel and create a pivot table summarizing key metrics (e.g., sales revenue, product quantities) by different dimensions (e.g., region, product category). 26. Experiment with different pivot table configurations (e.g., adding calculated fields, grouping data, creating hierarchical rows/columns) to gain insights into the dataset.		
	Fitting Linear Regression in Excel: 27. Import a dataset containing variables for linear regression analysis (e.g., independent and dependent variables). 28. Use Excel's built-in regression analysis tool to fit a linear regression model to the data and calculate coefficients, standard errors, and goodness-of-fit measures.		
	Creating Charts from Regression Analysis: 29. Perform linear regression analysis on a dataset containing independent and dependent variables. 30. Create a scatter plot chart to visualize the relationship between the variables, including the regression line and confidence intervals.		
	Pivot Chart: 31. Create a pivot chart based on a pivot table summarizing key metrics from a dataset. 32. Customize the pivot chart to display data trends and patterns dynamically as the underlying pivot table data is updated.		
	Dashboards: 33. Design a dashboard incorporating multiple charts and pivot tables to provide a comprehensive overview of business metrics or performance indicators. 34. Use interactive features such as slicers and timeline controls to enable users to filter and analyze data dynamically.		
	Form Controls: 35. Add form controls such as checkboxes, dropdown lists, and option buttons to interact with charts and pivot tables.		

	36. Create interactive features allowing users to customize chart views or update data dynamically based on user inputs.		

References

1. "Excel 2019 Bible" by Michael Alexander and Richard Kusleika
2. "Excel Formulas & Functions For Dummies" by Ken Bluttman and Peter Aitken
3. “Excel with Microsoft Excel: Comprehensive & Easy Guide to Learn Advanced MS Excel” by Naveen Mishra

Assessment Rubrics:

- Quiz / Assignment/ Quiz/ Discussion / Seminar
- Midterm Exam
- Final Exam

Programme	BSc. Computer Science				
Course Code	CSC8VN401				
Course Title	Predictive Modelling				
Type of Course	Vocational Minor				
Semester	III				
Academic Level	400-499				
Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	4	4	-	-	60
Pre-requisites	1. Basic Mathematical Concepts 2. Basic Statistics				
Course Summary	Predictive Modeling gives undergraduate students a solid foundation in predictive analytics techniques essential for data-driven decision-making. The course covers key topics such as correlation, covariance, linear regression, multiple regression, polynomial regression, logistic regression, and time series analysis and forecasting.				

Course Outcomes (CO):

CO	CO Statement	Cognitive Level*	Knowledge Category#	Evaluation Tools used
CO1	Apply correlation and covariance analysis to assess relationships between variables.	Ap	P	Problems/ Projects
CO2	Implement linear, multiple, and polynomial regression models to predict outcomes from numerical data.	Ap	P	Problems/ Projects
CO3	Employ time series analysis techniques to identify trends, and seasonal patterns, and make accurate forecasts.	An	P	Problems/ Projects
CO4	Evaluate model performance and interpret results to inform business decisions.	An	P	Analysis of reports and case studies
CO5	Utilize logistic regression to classify categorical outcomes and make data-driven decisions.	Ap	P	Projects
CO6	Acquire proficiency in building predictive models using real-world datasets	U	C	Assignments/ Quiz
* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C) # - Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge (M)				

Detailed Syllabus:

Module	Unit	Content	Hrs (48)	Marks (70)
I	**Correlation & Covariance**		**8**	**12**
	1	Data types or levels of measurement- Nominal, ordinal, interval and ratio	2	
	2	Covariance sample and population, sign and magnitude of covariance,	1	
	3	The covariance matrix, Covariance vs Correlation	2	
	4	Measures of Correlation, Simple correlation	1	
	5	Partial correlation and Multiple correlations	2	
II	**Regression Techniques**		**12**	**16**
	6	Simple linear regression	2	
	7	Basics of fitting and residual analysis	2	
	8	Multiple linear regression	2	
	9	Gauss Markov theorem	2	
	10	Least Squares Method, ordinary least squares, weighted least squares	2	
	11	Polynomial regression	2	
III	**Logistics Regression**		**11**	**14**
	12	Basics of Logistic regression	2	
	13	Logistic regression with binary predictor	2	
	14	Odds ratio, z-statistic, p-values	3	
	15	Confidence intervals	2	
	16	Logistic regression with categorical predictors	2	
IV	**Time Series analysis and forecasting**		**16**	**28**
	17	Components of time-series, additive and multiplicative models	3	
	18	Methods for measurement of trends	2	
	19	Methods for measurement of seasonal fluctuations	3	
	20	Forecasting, Autocorrelation	2	
	21	ARIMA Model	3	
	22	ARMA Model	3	
V	**Open Ended Module: Assignments, Case study**		**12**	
	1. Provide real-world examples to understand the relationships between variables in data analysis using covariance, correlation 2. Using real examples, understand the difference between different types of correlation. 3. Provide examples for nominal, ordinal, interval, and ratio data types 4. Provide examples of how linear regression is used in various fields such as economics, finance, healthcare, and engineering to analyze relationships between variables and make predictions.		4	
	5. Analyze the relationship between different variables using multiple linear regression. Eg: Health Care analytics: patients' demographics, lifestyle factors, and medical history using multiple linear regression. 6. Analyze the relationship between different variables using logistic regression. Eg: Predict the risk of developing a certain disease (e.g.,		8	

	diabetes, cancer) using logistic regression with binary predictors such as genetic markers, lifestyle factors, and medical history. 7. Analyze the time series model using ARIMA/ ARMA model		

Mapping of COs with PSOs and POs:

	PS O1	PS O 2	PS O 3	PSO 4	PS O 5	PS O6	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6
CO 1	3	-	3	-	-	-						
CO 2	3	-	3	2	-	2						
CO 3	3	-	3	2	-	2						
CO 4	-	-	2	-	-	2						
CO 5	3	-	3	2	-	2						
CO 6	-	-	2	-	-	2						

Correlation Levels:

Level	Correlation
-	Nil
1	Slightly / Low
2	Moderate / Medium
3	Substantial / High

Mapping of COs to Assessment Rubrics:

	Internal Exam	Assignment	Project Evaluation	End Semester Examinations
CO 1	✓	✓	✓	✓
CO 2	✓	✓	✓	✓
CO 3	✓	✓	✓	✓
CO 4	✓		✓	✓

CO 5	✓			✓
CO 6	✓			✓

Assessment Rubrics:

- Quiz / Assignment/ Quiz/ Discussion / Seminar
- Midterm Exam
- Final Exam

References:

1. Fan, Jianqing, et al. Statistical Foundations of Data Science. United States, CRC Press, 2020.
2. Hilbe, Joseph M. Practical Guide to Logistic Regression. United States, CRC Press, 2016.
3. Nielsen, Aileen. Practical Time Series Analysis: Prediction with Statistics and Machine Learning. United States, O'Reilly Media, 2019.
4. Fundamentals of Mathematical Statistics. United Kingdom, Sultan Chand & Sons, 2020.

Programme	BSc. Computer Science				
Course Code	CSC1VN102				
Course Title	Statistical Foundations for Artificial Intelligence				
Type of Course	Vocational Minor				
Semester	I				
Academic Level	100 - 199				
Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	4	3	-	2	75
Pre-requisites	1. A strong foundation in algebra 2. Fundamentals of Set theory and logic				
Course Summary	The course on probability and statistics covers fundamental topics including descriptive statistics (measures of central tendency and dispersion), probability theory (events, sample spaces, probability laws, random variables, and distributions), inferential statistics (regression analysis), and applications in various fields such as science, engineering, economics, and social sciences, emphasizing critical thinking, data analysis, and problem-solving skills.				

Course Outcomes (CO):

CO	CO Statement	Cognitive Level*	Knowledge Category#	Evaluation Tools used
CO1	Apply fundamental statistics concepts	Ap	C	Quizzes, Homework, Exams
CO2	Analyze data using descriptive statistics	An	P	Projects, Midterm, Exams
CO3	Perform regression analysis	An	P	Projects, Exams
CO4	Apply probability and statistics in real-world situations	Ap	C	Projects, Exams
CO5	Develop critical thinking and problem-solving skills	E	M	Homework, Projects
CO6	Communicate statistical findings effectively	E	M	Presentations, Reports

* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C)
# - Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge (M)

Detailed Syllabus

Module	Unit	Content	Hrs (45+30)	Mark
I		**DESCRIPTIVE STATISTICS**	**9**	**15**
	1	Concept of primary and secondary data, Methods of collection	1	
	2	Measures of central tendencies (Mean, Median, Mode, HM, GM)	4	
	3	Measures of dispersion, Relative Measures and Absolute Measures	2	
	4	Range, Quartile deviation, Mean deviation, standard deviation, Variance	2	
II		**STATISTICAL INFERENCE AND REGRESSION ANALYSIS**	**11**	**15**
	5	Principles of Least Squares and Fitting of Stright Line	2	
	6	Point estimation: maximum likelihood estimation (MLE), method of moments. Confidence intervals for population parameters.	3	
	7	Pearson's Coefficient of Correlation and Rank Correlation	3	
	8	Simple linear regression and multiple linear regression. Logistic regression for classification problems.	3	
III		**PROBABILITY THEORY**	**11**	**20**
	9	Random experiment, Sample point, Sample Space	1	
	10	Events, Operation of events (Union, Intersection, Complement of Events)	2	
	11	Exclusive and exhaustive events, equally likely events with examples	1	
	12	Classical approach to probability	1	
	13	Axiomatic definitions of probability, simple problems	2	
	14	Conditional probability	1	
	15	Inverse probability	1	
	16	Baye's Theorem	2	
IV		**ADVANCED PROBABILITY DISTRIBUTION**	**14**	**20**
	17	Discrete and continuous random variables and probability distribution	2	
	18	Binomial distribution: Definition, Expectation, Variance, Moment Generating Function and Problems	2	
	19	Poisson distribution: Definition, Expectation, Variance, Moment Generating Function and Problems	2	
	20	Normal distribution: Definition, Expectation, Variance, Moment Generating Function, Standard normal curve and Problems	3	

	21	Testing of Hypothesis: General principles of testing, Two types of errors	3	
	22	Type of Testing: T-Test, ANOVA-Test, Chi-square test (Concept Only)	2	
V		**Lab Activities (Use Sci Lab)**	**30**	
	1	Implements mean, median and mode hight of then students	20	
		Determine the standard deviation and variance		
		Plot a histogram to visualize their distribution		
		Use SciLab to perform simple linear regression on a dataset with two variables.		
		Implement SciLab code to plot box plots, scatter plots, and density plots for the dataset to explore its characteristics.		
		Use SciLab to perform Least Square		
		Implement algorithms for multiple linear regression and logistic regression in SciLab to predict outcomes based on input features.		
		You have a deck of 52 playing cards. Calculate the probability of drawing a face card (jack, queen, or king) from the deck.		
		Simulate random experiments and calculate probabilities of events using Scilab.		
		Write functions in SciLab to calculate probabilities for events based on given probability distributions (e.g., binomial, normal).		
	2	Case Study	2	
	3	Develop a predictive model using statistical techniques and tools for identifying a real-world problem in Artificial Intelligence.	8	

Mapping of COs with PSOs and POs:

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PO1	PO2	PO3	PO4	PO5	PO6
CO 1	1	3	-	1	1	1						
CO 2	1	3	-	-	1	-						
CO 3	1	3	-	-	2	2						
CO 4	1	3	-	-	2	2						
CO 5	2	1	-	1	1	-						
CO 6	2	1	1	2	2	1						

Correlation Levels:

Level	Correlation
-	Nil
1	Slightly / Low
2	Moderate / Medium
3	Substantial / High

Assessment Rubrics:

- Quiz / Assignment/ Quiz/ Discussion / Seminar
- Midterm Exam
- Programming Assignments (20%)
- Final Exam (70%)

Mapping of COs to Assessment Rubrics:

	Internal Exam	Assignment	Practical Evaluation	End Semester Examinations
CO 1	✓	✓	✓	✓
CO 2	✓	✓		✓
CO 3	✓	✓		✓
CO 4	✓	✓		✓
CO 5	✓	✓		✓
CO 6	✓	✓		✓

References:

1. Introduction to Mathematical Statistics, Hogg R V Craig A T, Macmillan
2. Mathematical Statistics, Freund J E, Waple R E, Prentice Hall of India.
3. Probability and Statistics for Engineers, Miller I Freund J E, Prentice Hall of India.
4. Statistics for Management, Levin R I, Prentice Hall of India
5. Introduction to Mathematical Statistics, Hogg R V Craig A T, Macmillan
6. Mathematical Statistics, Freund J E, Waple R E, Prentice Hall of India.

Programme	BSc. Computer Science				
Course Code	CSC2VN102				
Course Title	Foundations Artificial Intelligence				
Type of Course	Vocational Minor				
Semester	II				
Academic Level	100-199				
Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	4	3	-	2	75
Pre-requisites	A course on Discrete Mathematics is recommended				
Course Summary	This course provides an introduction to the field of Artificial Intelligence covering fundamental concepts, problem solving methods such as search algorithms and heuristics approaches and different knowledge representation techniques. The course addresses the ethical dimensions of AI and their societal impacts.				

Course Outcomes (CO):

CO	CO Statement	Cognitive Level*	Knowledge Category#	Evaluation Tools used
CO1	Able to gain insight into the evolution of key ideas and technologies by exploring the Artificial Intelligence history and its foundational concepts.	U	C	Instructor-created exams / Quiz/Assignment/ Seminar
CO2	Able to acquire knowledge and skills to understand, design, implement intelligent agents to perceive, reason and act within their environments.	U	C	Instructor-created exams/ Quiz/Assignment/ Seminar
CO3	Proficiency in various uninformed and informed search strategies along with constraint satisfaction problem solving methods.	U	C	Instructor-created exams/ Quiz/Assignment/ Seminar
CO4	Ability to design and implement logical agents and construct ontologies that capture the semantics of a domain, facilitating knowledge representation.	U	C	Instructor-created exams/ Quiz/Assignment/ Seminar

CO5	Understand the ethical considerations of AI and their societal impacts and gain insights into the future trajectory of AI by analysing the emerging trends.	U	C	Instructor-created exams/ Quiz/Assignment/ Seminar
CO6	Represent various AI problems using algorithmic approaches and enhance problem-solving skills by visualizing solutions through the utilization of software tools.	U, Ap	C, P	Practical Assignment / Observation of Practical Skills
* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C) # - Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge (M)				

Detailed Syllabus

Module	Unit	Contents	Hrs (45+30)	Marks
I		**Introduction to AI**	**11**	**18**
	1	Artificial Intelligence: Definition and Applications	2	
	2	Foundations of Artificial Intelligence	1	
	3	History of Artificial Intelligence, State of the Art	2	
	4	Intelligent Agents: Agents and Environments	1	
	5	The Concept of Rationality, Nature of Environments: Specifying the Task Environment, Properties of Task Environment	3	
	6	Structure of Agents: Agent Programs, Simple Reflex Agent, Model Based Reflex Agent, Goal Based Agent, Utility Based Agent, Learning Agent (Concept Only, No Algorithm required)	2	
II		**AI Problem Solving**	**14**	**20**
	7	Problem Solving Agents (Concept Only), Examples Problems: Toy problems, Real world problems	3	
	8	Solutions for searching: Tree Search and Graph Search and Measuring Problem Solving Performance (Concept Only)	1	
	9	Uninformed Search Strategies: Breadth First Search, Uniform Cost Search, Depth First Search,	4	
	10	Informed search strategies: Greedy Best First search, A* Search, Heuristic Search (Concept Only)	2	

	11	Constrain Satisfaction Problems: Definition, Examples: Map colouring, Job-Shop scheduling	2	
	12	Constraint Propagation: Node Consistency, Arc Consistency, Path Consistency and K-Consistency	2	
III		**Knowledge Representation**	**13**	**20**
	13	Logical agents: Knowledge based agents, The Wumpus world	2	
	14	Logic: Definition, Propositional logic, Syntax and Semantics, Simple Knowledge Base	3	
	15	First Order Logic: Definition, Syntax and Semantic (Models, Symbols and Interpretations, Terms, Atomic Sentences, Complex Sentences, Quantifiers, Equality)	3	
	16	Ontological Engineering: Definition	1	
	17	Categories and Objects: Physical Composition, Measurements, Objects: Things and Stuff, Process, Time Intervals, Fluent and Objects Quantifying Uncertainty (Concept Only)	4	
IV		**AI: Philosophical Foundations and Future**	**7**	**12**
	18	Weak AI: Can machines act intelligently?	1	
	19	Strong AI: Can machines really think?	2	
	20	Ethics and risks of developing Artificial Intelligence	2	
	21	Agent components and architectures	1	
	22	Are we going in the right direction? What if AI succeed?	1	
V		**Lab Activities**	**30**	
		1. Identify the various software platforms for AI programming. 2. Identify the various platforms used for No-Code AI (Google Cloud Auto ML, Microsoft Azure Al Buildeer, IBM Watson Studio etc.) 3. Use chatbot platforms like ChatGPT or any other to engage in conversational interactions and understand how natural language processing works. 4. Use online tools or any applications that demonstrate image recognition capabilities. 5. Use online platforms or software that provide		

		interactive AI demos and simulations, such as neural network visualisers or AI-powered character generators. 6. Demonstrate the use of AI-based image editing tools (Actions: Remove objects from images, enhance details, or perform automated retouching.) 7. Utilize AI-powered text summarisation tools like SummarizeBot or Resoomer to generate summaries of lengthy articles or research papers. 8. Use any presentation software like Microsoft PowerPoint or Google Slides to demonstrate AI-driven design suggestions and layout recommendations. 9. Explore AI-based translation tools such as Google Translate or DeepL for translating text between different languages. 10. Assign students to analyse news articles, advertisements, or social media posts using AI technologies. 11. Introduce Students to data visualisation using Tableau Public, an accessible data visualisation tool. 12. Use Google's Teachable Machine platform to create a simple image classification model.		
		Case Study: Provide students with case studies or examples of AI applications in different domains (e.g., healthcare, finance, marketing).		
		Organize demos of AI technologies and applications, such as virtual assistants, autonomous vehicles, facial recognition systems, and recommendation engines.		

Mapping of COs with PSOs and POs:

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PO1	PO2	PO3	PO4	PO5	PO6
CO 1	1	-	-	-	-	-						
CO 2	2	-	2	2	2	2						
CO 3	2	-	2	2	2	2						
CO 4	2	-	2	2	2	2						
CO 5	2	-	2	2	-	-						

CO 6	1	-	1	1	-	-						

Correlation Levels:

Level	Correlation
-	Nil
1	Slightly / Low
2	Moderate / Medium
3	Substantial / High

Assessment Rubrics:

- Quiz / Assignment/ Discussion / Seminar
- Midterm Exam
- Programming Assignments (20%)
- Final Exam (70%)

Mapping of COs to Assessment Rubrics :

	Internal Exam	Assignment	Practical Evaluation	End Semester Examinations
CO 1	✓	✓		✓
CO 2	✓	✓		✓
CO 3	✓	✓		✓
CO 4	✓	✓		✓
CO 5	✓	✓		✓
CO 6	✓	✓	✓	✓

References:

1. Stuart Russell, Peter Norvig, "Artificial Intelligence: A Modern Approach", 3rd Edition, Prentice Hall, 2010.
2. Deepak Khemani, "A First Course in Artificial Intelligence", McGraw Hill Education, 2017.
3. Elaine Rich, Kevin Knight, & Shivashankar B Nair, "Artificial Intelligence", McGraw Hill, 3rd Edition, 2009.

Programme	BSc. Computer Science				
Course Code	CSC3VN202				
Course Title	Automation and Robotics				
Type of Course	Vocational Minor				
Semester	III				
Academic Level	200-299				
Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	4	3	-	2	75
Pre-requisites	No pre-requisites required				
Course Summary	This course provides a comprehensive overview of automation which includes their production systems, elements, automation functions and usage of discrete and continuous control system. The course also explores the fundamentals of robotics, including anatomy, process control and how these functions could be improved by the integration of Artificial Intelligence.				

Course Outcomes (CO):

CO	CO Statement	Cognitive Level*	Knowledge Category#	Evaluation Tools used
CO1	Understand the production systems and automation, enabling them to analyse, optimize and evaluate the different levels of automation.	U	C	Instructor- created exams / Quiz/Assignment/ Seminar
CO2	Able to recognize the difference between the process industries, manufacturing industries, continuous and discrete control system.	U	C	Instructor- created exams/ Quiz/Assignment/ Seminar
CO3	Proficiency in understanding the various forms of process control which includes the direct digital control, programmable logic control, distributable control systems etc.	U	C	Instructor- created exams/ Quiz/Assignment/ Seminar
CO4	Familiarize with the various hardware components used for automation and process control such as sensors, actuators analog-digital converters etc.	U	C	Instructor- created exams/ Quiz/Assignment/ Seminar

CO5	Understand the present developments in the field of automation and robotics and how integrating artificial intelligence can contribute to the future of these systems.	U	C	Instructor-created exams/ Quiz/Assignment/ Seminar
CO6	Represent various problems using algorithmic approaches and enhance problem-solving skills by visualizing solutions through the utilization of software tools.	U, Ap	C, P	Practical Assignment / Observation of Practical Skills
* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C) # - Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge (M)				

Detailed Syllabus:

Module	Unit	Contents	Hrs (45+30)	Mark
I		**Introduction to Automation**	**11**	**15**
	1	Production systems - Facilities, Manufacturing support systems	2	
	2	Automation in production systems – Automated manufacturing system, Computerized manufacturing support systems, Reasons for automating	3	
	3	Manual labour in production systems	1	
	4	Elements of automation - power to accomplish the process, Program of instructions, control system	3	
	5	Advanced automation functions – safety monitoring, maintenance and repair diagnostics, error detection and recovery	1	
	6	Levels of automation	1	
II		**Control Systems**	**13**	**15**
	7	Process industries versus Discrete manufacturing industries, Continuous versus Discrete control	1	
	8	Continuous control system	3	
	9	Discrete control system	1	
	10	Computer process control, Control requirements, Capabilities of computer control	2	

	11	Forms of computer process control - Computer process monitoring, Direct digital control, Computer numerical control and robotics, Programmable logic controllers, Supervisory control and data acquisition, Distributed control systems	3	
	12	Hardware for automation and process control (Concept only) - Sensors, Actuators, Analog to Digital converters Digital to Analog converters, Input/output devices for discrete data.	3	
III	**Industrial Robotics**		**15**	**25**
	13	Robot anatomy – Joints and links, Common robot configurations, Joint drive systems, Sensors in robotics	4	
	14	Robot control systems – Limited sequence control, Playback with point-to-point control, Playback with continuous path control, Intelligent control	2	
	15	End effectors – Grippers, Tools	1	
	16	Robot Programming – Lead through programming, Powered lead through, Motion programming, Advantages and disadvantages	2	
	17	Discrete process control – logic control, sequence control	4	
	18	Programmable Logic Controllers, Components of PLC	2	
IV	**Automation and Robotics: Present and Future**		**6**	**15**
	19	Machine Intelligence, Computer and Robotics	1	
	20	Flexible automation vs Robotics technology	1	
	21	Artificial Intelligence and Automated Manufacturing, AI and Robotics	2	
	22	Robotics in India, Future of Robotics	2	
V	**Lab Activities**		**30**	
	1	Set up a simulation of a production system using any software tools.	28	
	2	Utilise online simulation tools and platforms that allow students to simulate robot control.		
	3	Utilise online simulation tools and platforms that allow students to simulate automation systems.		
	4	Assign online projects or challenges that require participants to design, program, or simulate automation systems and robotic applications.		

	5	Explore any online virtual reality (VR) applications that simulate manufacturing environments, robotic operations, and automation scenarios.		
	6	Analyze publicly available datasets on platforms like Kaggle, UCI Machine Learning Repository, or Data.gov.		
	7	Experiment with virtual robotics simulations using platforms like V-REP (Virtual Robot Experimentation Platform) or Gazebo.		
	8	Designing and building a simple chatbot using no-code platforms like ChatGPT or Google's Dialogflow.		
	9	Allow students to customize their chatbots by defining conversational flows.		
	10	Provide Programmable Logic Controllers (PLCs) and challenge them to program various control sequences.		
	11	Host a discussion session on the intersection of Artificial Intelligence (AI) and Robotics in automated manufacturing.	2	

Mapping of COs with PSOs and POs:

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PO1	PO2	PO3	PO4	PO5	PO6
CO 1	1	-	-	-	-	-						
CO 2	2	2	-	-	2	-						
CO 3	2	2	-	-	2	-						
CO 4	2	2	-	-	2	-						
CO 5	1	-	-	-	-	1						
CO 6	-	-	2	2	-	-						

Correlation Levels:

Level	Correlation
-	Nil
1	Slightly / Low
2	Moderate / Medium
3	Substantial / High

Assessment Rubrics:

- Quiz / Assignment/ Discussion / Seminar
- Midterm Exam
- Programming Assignments (20%)
- Final Exam (70%)

Mapping of COs to Assessment Rubrics:

	Internal Exam	Assignment	Practical Evaluation	End Semester Examinations
CO 1	✓	✓		✓
CO 2	✓	✓		✓
CO 3	✓	✓		✓
CO 4	✓	✓		✓
CO 5	✓	✓		✓
CO 6	✓	✓	✓	✓

References:

1. Mikell P. Groover, "Automation, Production Systems and Computer Integrated Manufacturing", 4th edition, Pearson Education, 2017.

2. S.R. Deb, S. Deb " Robotics Technology and flexible automation," Tata McGraw-Hill Education, 2017.

3 Mikell P. Groover, " "Industrial Robots - Technology, Programming and Applications", McGraw-Hill Education, 2017

Programme	BSc. Computer Science				
Course Code	CSC8VN402				
Course Title	Expert Systems and Fuzzy Logic				
Type of Course	Vocational Minor				
Semester	VIII				
Academic Level	400 - 499				
Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	4	4	-	-	60
Pre-requisites	1. Familiarity with basic logic and set theories. 2. Understanding the fundamentals of computer science, such as algorithms and data structures, can be beneficial for the implementation aspects of expert systems. 3. A basic understanding of probability and statistics is often required.				
Course Summary	The Fuzzy logic and expert systems course introduce two interconnected fields in artificial intelligence: fuzzy logic and expert systems. Fuzzy logic deals with reasoning under uncertainty and imprecision, while expert systems involve the development of computer-based systems that emulate human expertise in specific domains.				

Course Outcomes (CO):

CO	CO Statement	Cognitive Level*	Knowledge Category#	Evaluation Tools used
CO1	Explain the fundamental concepts of fuzzy set theory and interpret membership functions and linguistic variables.	U	F	Instructor-created exams / Quiz
CO2	Design and implement fuzzy controllers for decision-making. Develop fuzzy inference systems (FIS) for various applications and apply fuzzy clustering techniques for pattern recognition.	U	C	Practical Assignment / Observation of Practical Skills

CO3	Describe the role of expert systems in artificial intelligence and Understand knowledge representation techniques in expert systems.	Ap	P	Practical Assignment / Observation of Practical Skills
CO4	Explain the functioning of inference engines in rule-based systems.	Ap	P	Practical Assignment / Observation of Practical Skills
CO5	Acquire domain knowledge for expert system development.	An	C	Instructor-created exams / Quiz
CO6	Construct a knowledge base and define rules for an expert system and implement validation and refinement techniques for expert systems.	Ap	P	Practical Assignment / Observation of Practical Skills
* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C) # - Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge (M)				

Detailed Syllabus

Module	Unit	Content	Hrs (48+12)	Mark
I		**Introduction to Fuzzy Logic**	**8**	**10**
	1	Overview of Fuzzy Logic	1	
	2	Fuzzy Sets and Membership Functions	2	
	3	Fuzzy Operations (Union, Intersection, Complement)	2	
	4	Basic principles of fuzzy logic. Fuzzification and defuzzification.	2	
	5	Linguistic variables and terms.	1	
II		**Fuzzy Inference Systems (FIS) and Fuzzy Logic Applications**	**12**	**20**
	6	Mamdani FIS-Rule-based systems in fuzzy logic, Rule base and implication methods.	2	
	7	Sugeno FIS-Structure and operation of Sugeno FIS. Comparison with Mamdani FIS.	2	
	8	Basic structure of fuzzy logic controllers (FLCs)	3	
	9	Rule-based systems and fuzzy inference	3	
	10	Applications of fuzzy logic controllers	2	
III		**Introduction to Expert Systems and Rule-Based Systems**	**12**	**20**
	11	Definition and characteristics of expert systems.	2	
	12	Knowledge representation and reasoning.	3	
	13	Expert system components: knowledge base, inference engine, user interface. Examples and applications of expert systems	3	
	14	Rule-based systems and production rules, Forward and backward chaining.	2	

	15	Inference mechanisms in expert systems, Examples of rule-based expert systems.	2	
IV		**Introduction to SCILAB/MATLAB Programming**	**16**	**20**
	16	SCILAB/MATLAB environment and basic navigation, Variables, data types, and basic operations, Script files and running SCILAB/MATLAB code. Introduction to functions and function files.	3	
	17	Introduction to functions and function files, Conditional statements (if, else, elseif), Loop structures (for, while).	2	
	18	Logical operators and relational expressions, Vectorized operations and element-wise operations.	2	
	19	Introduction to arrays, matrices, and vectors, Cell arrays and structures, Indexing and slicing in SCILAB/MATLAB, Working with multidimensional arrays.	2	
	20	Basic file input/output operations, Reading and writing data files (text, CSV, Excel), Data visualization using plotting functions.	2	
	21	Statistical analysis and plotting techniques, Fuzzy logic toolbox in SCILAB/MATLAB.	2	
	22	Expert system development tools in SCILAB/MATLAB, Building expert systems using SCILAB/MATLAB.	3	
V		**Open end**	**12**	
		Case Studies: Real-world applications and their impact. Technological Challenges: Addressing the limitations and exploring new solutions. Future Prospects: Predictions and potential advancements in the field.		

Mapping of COs with PSOs and POs:

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PO1	PO2	PO3	PO4	PO5	PO6
CO 1	1	3	-	-	-	1						
CO 2	1	3	-	-	1	-						
CO 3	1	3	-	-	2	2						
CO 4	1	3	-	-	2	2						
CO 5	2	1	3	1	1	-						
CO 6	2	1	3	2	2	1						

Correlation Levels:

Level	**Correlation**

-	Nil
1	Slightly / Low
2	Moderate / Medium
3	Substantial / High

Assessment Rubrics:

- Quiz / Assignment/ Quiz/ Discussion / Seminar
- Midterm Exam
- Programming Assignments (20%)
- Final Exam (70%)

Mapping of COs to Assessment Rubrics:

	Internal Exam	Assignment	Practical Evaluation	End Semester Examinations
CO 1	✓	✓		✓
CO 2	✓	✓	✓	✓
CO 3	✓	✓	✓	✓
CO 4		✓	✓	✓
CO 5		✓	✓	✓
CO 6	✓	✓	✓	✓

References:

1. "Fuzzy Logic with Engineering Applications" by Timothy J. Ross
2. "Expert Systems: Principles and Programming" by Joseph C. Giarratano and Gary D. Riley
3. "Fuzzy Sets and Fuzzy Logic: Theory and Applications" by George J. Klir and Bo Yuan
4. "Expert Systems: Principles and Case Studies" by Efraim Turban, Jay E. Aronson, and Ting-Peng Liang
5. “Introduction to Fuzzy Logic using MATLAB” by S.N. Sivanandam, S. Sumathi, and S. N. Deepa.
6. Nagar, S. (2017). Introduction to Scilab: For Engineers and Scientists. Apress.

General Foundation Papers

Programme	B. Sc. Computer Science				
Course Code	CSC1FM105				
Course Title	Data Analysis and Visualisation Through Spread sheets				
Type of Course	**MDC**				
Semester	I				
Academic Level	100-199				
Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	3	3	-	-	45
Pre-requisites	• Basic understanding of computers • Familiarity with basic mathematical operations				
Course Summary	This course provides a comprehensive introduction to Spreadsheets, focusing on understanding formulas, functions, data organization, analysis techniques, and data visualization. Participants will gain skills in spreadsheet management, data cleansing, analysis, and visualization using Excel's various tools and features.				

Course Outcomes (CO):

CO	CO Statement	Cognitive Level*	Knowledge Category#	Evaluation Tools used
CO1	Students will demonstrate proficiency in managing spreadsheets, including creating, formatting, and manipulating data within Excel workbooks. They will be able to effectively navigate Excel's interface and utilize toolbars.	U	P	Instructor-created exams / Quiz
CO2	Learners will understand the importance of data organization and cleansing in Excel. They will be able to import, export, filter, sort, validate, and remove duplicates from datasets. Students will develop skills to ensure data integrity and consistency, enhancing their ability to work with clean and organized data sets.	U	P	Instructor-created exams/ Home Assignments
CO3	Participants will acquire advanced data analysis skills like pivot	Ap	P	Instructor-created exams

	tables, what-if analysis, and goal seek. They will be able to apply various Excel functions and tools to perform complex calculations, analyze trends, and make informed decisions based on data analysis.			
CO4	Students will gain proficiency in data visualization techniques using Excel. They will be able to create a variety of charts, design pivot charts, dashboards for effective data analysis. Additionally, learners will be able to implement form controls for interactive data manipulation in their visualizations.	Ap	P	Instructor-created exams
CO5	Learners will develop skills in advanced features of Excel like macros, protect data sheets and workbooks, utilize split, freeze, and hide options effectively, incorporate add-ins for extended functionalities, and manage printing options in Excel for professional presentation of data.	Ap	P	Instructor-created exams
* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C) # - Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge (M)				

Detailed Syllabus:

Module	Unit	Content	Hrs (36+9)	Marks (50)
I		**Introduction to Excel & Understanding Formulas, Functions**	**9**	**15**
	1	Features of Spreadsheet	1	
	2	Parts of Excel Window, Tool bars, Work sheet and Work book, Insertion and Deletion of cells, columns, rows	2	
	3	Formatting in Excel (Merge, Warp, Font Formatting, Number Formatting, Borders and Shading, Colouring)	2	
	4	Range, Autofill, Autosum, Relative, Absolute and Mixed Referencing in Excel, Linking data between worksheets	2	
	5	Formulas and Functions in Excel: Use of Formula Bar, Functions: SUM,ROUND, CEIL, FLOOR,IF, AND,	2	

		OR,AVERAGE, MIN, MAX ,COUNT, COUNTIF, SUMIF, VLOOKUP,HLOOKUP		
II		**Cleansing and Organising Data in Excel**	**9**	**10**
	6	Importance of Data Cleansing and Organisation	1	
	7	Data Import and Export	2	
	8	Filtering and Sorting	2	
	9	Data Validation and remove Duplicates	1	
	10	Group, Ungroup, Subtotal	2	
	11	Conditional Formatting – Highlight Cell Rules, Top/Bottom Rules	1	
III		**Advanced Techniques for Data Analysis**	**8**	**10**
	12	Features of Pivot table	1	
	13	Pivot Table creation	2	
	14	What-if Analysis	2	
	15	Goal Seek	2	
	16	Watch Window	1	
IV		**Data Visualisation Techniques**	**10**	**15**
	17	Creating Charts, Different types of charts	2	
	18	Formatting Chart Objects, Changing the Chart Type, Showing and Hiding the Legend, Showing and Hiding the Data Table	2	
	19	Pivot Chart	2	
	20	Dashboards	1	
	21	Form Controls	3	
V		**Open Ended Module: More about Excel**	**9**	
		1. Recording and Running Macros 2. Protecting Data Sheets and Workbooks 3. Split, Freeze and Hide options 4. Add-ins 5. Printing options in Excel		

References

1. "Excel 2019 Bible" by Michael Alexander and Richard Kusleika
2. "Excel Formulas & Functions For Dummies" by Ken Bluttman and Peter Aitken

3. “Excel with Microsoft Excel: Comprehensive & Easy Guide to Learn Advanced MS Excel” by Naveen Mishra

Assessment Rubrics:

- Quiz / Assignment/ Quiz/ Discussion / Seminar
- Midterm Exam
- Final Exam

Programme	B. Sc. Computer Science				
Course Code	CSC2FM106				
Course Title	Digital Empowerment through Ethical Standards				
Type of Course	**MDC**				
Semester	II				
Academic Level	100 – 199				
Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	3	3	-	-	45
Pre-requisites	Basic understanding of computers				
Course Summary	This course explores the evolution from pre-digital challenges to the current digital landscape, covering historical milestones, key technologies, and the vision of Digital India. It emphasizes the benefits and importance of digital revolution while addressing ethical and security considerations. Participants engage with digital tools for personal and professional growth and examine case studies on digital infrastructure, missions, and services to understand real-world applications.				

Course Outcomes (CO):

CO	CO Statement	Cognitive Level*	Knowledge Category#	Evaluation Tools used
CO1	Students will be able to analyze the challenges of the pre-digital age and comprehend the importance and benefits of digital revolution, facilitating a deeper understanding of technological evolution.	An	F	Instructor-created exams / Quiz
CO2	Participants will gain familiarity with key digital technologies like Cloud Computing, IoT, AI, and Blockchain, equipping them with the knowledge to identify their applications and potential benefits in different sectors.	U	C	Instructor-created exams/ Home Assignments
CO3	Students will develop insights into Digital India initiatives and emergence of Kerala as Digital Society	U	C	Instructor-created exams
CO4	Through exploration of digital tools	Ap	P	Instructor-

	for personal and professional growth, students will enhance their digital literacy and ability in utilizing tools for data sharing, online learning, networking, and content creation, empowering them to thrive in the digital age.			created exams
CO5	Learners will become aware of ethical and security considerations in the digital age, including privacy concerns, Intellectual Property Rights, key terminologies related to cyber security, and an introduction to cyber laws in India, fostering responsible digital citizenship.	U	C	Instructor-created exams
CO6	Students will analyze real-world case studies of digital infrastructure projects, digital missions, and digital services to demonstrate a comprehensive understanding of the practical applications and implications of digital technologies in various contexts, fostering critical thinking and strategic decision-making skills in digital transformation initiatives.	An	C	Instructor-created exams
* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C) # - Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge (M)				

Detailed Syllabus:

Module	**Unit**	**Content**	**Hrs** **36+9**	**Marks** **(50)**
I		**Transition to Digital World**	**7**	**8**
	1	Challenges of Pre-Digital Age	1	
	2	Importance and Benefits of Digital Revolution	2	
	3	Key concepts: digitization, digitalization, digital transformation	1	
	4	Introduction to Key Digital Technologies: Cloud Computing, IoT, AI, Block Chain	3	

II		**Perspective of Digital India & Digital Innovations in Kerala**	**11**	**15**
	5	Understanding Digital India: Concept, Objectives, and Evolution	1	
	6	Overview of Digital Infrastructure: Broadband Connectivity, Digital Literacy, and Access to Information	2	
	7	Vision of Digital India: DigiLocker, E-Hospitals, e-Pathshala, BHIM, , e-Health Campaigns	3	
	8	Kerala-Emergence as Digital Society : Internet & Mobile Penetration in Kerala, 4 Pillars of Digital Emergence in Kerala (Akshaya Project, IT@School Project, Digital Infrastructure Availability, State Data Centre & allied Applications),	2	
	9	Role of K-DISC in Digital Empowerment	1	
	10	Kerala State IT Mission: Core IT Infrastructure, e-Governance Applications, Service Delivery Platforms,	2	
III		**Digital Tools for Personal and Professional Growth**	**9**	**12**
	11	Digital Tools for Data Sharing: Google Drive, Google Sheets	2	
	12	Digital Tools for Data Sharing: Google Docs, Google Classroom	3	
	13	Online learning platforms and resources (e.g., Coursera, Khan Academy, MOOCs, Duolingo)	2	
	14	Networking Tools: LinkedIn	1	
	15	Content Creation and Management: Canva	1	
IV		**Ethical and Security Considerations in the Digital Age**	**9**	**15**
	16	Understanding privacy in the digital age	1	
	17	Legal and ethical considerations in data collection and processing: Intellectual Property Rights (IPR)	2	
	18	Key Terminologies: Cyber Security, Cyber Crime, Cyber Attack, Cyber Espionage, Cyber Warfare	2	
	19	Authentication, Authorisation	1	
	20	Cyber Crimes and Classification	2	
	21	Introduction to Cyber Laws in India	1	
V		**Open Ended Module: Case Study (One from each set)**	**9**	
	1	Case Study on Digital Infrastructure Projects: (Bharat Broadband Network (BBNL) , Submarine Cable Project, Google Data Center)	3	
	2	Case Study on Digital Mission:	3	

		(Digital Literacy Missions in Kerala, SmartDubai Project, China's Digital Silk Road)		
	3	Case Study on Digital Services: (MyGov.in , Moodle LMS, Digital Payment Services)	3	

References

1. "Digital India Importance Needs and Values" by S K Kaushal
2. "Cyber Security in India: Government, Law Enforcement and Corporate Sector" by Vipin M. Chaturvedi and Shivani Kapoor
3. "Information Security: Principles and Practices in Indian Context" by R.S. Pressman, G. Sharma, and G. Sridhar
4. "Introduction to Computer Security" by Michael Goodrich and Roberto Tamassia
5. https://kdisc.kerala.gov.in/
6. https://itmission.kerala.gov.in/

Assessment Rubrics:

- Quiz / Assignment/ Quiz/ Discussion / Seminar
- Midterm Exam
- Final Exam

Programme	B. Sc. Computer Science				
Course Code	CSC3FV108(1)				
Course Title	Introduction to Cyber laws				
Type of Course	VAC				
Semester	III				
Academic Level	100 - 199				
Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	3	3	-	3	45
Pre-requisites	1. Basic Computer Literacy 2. Familiarity with Online Platforms 3. Willingness to Learn				
Course Summary	Introduction to Cyber laws provides students with a foundational understanding of various concepts Cyber Crimes and Cyber laws against them.				

Course Code		Course Title Introduction to Cyber Laws		
Credit 3		Duration 45 hrs		
Sl. NO:	Course Outcome	Cognitive level *	Know ledge catego ry #	Evaluation Tools used
CO1	To understand the concept of Cyber Space ,Cyber Crimes and cyber laws	U	C	Instructor-Create Exams or Quiz
CO2	To understand details of cyber crimes and criminals	A	P	Discussions and Quizzes
CO3	To examine various provisions in IT Act 2000	U	F	Instructor created exams or Home assignments
CO4	To Identify Intellectual Property right and E-commerce related issues.	A ,E	P	Discussions, Quizzes
CO5	To get overall idea of cyber laws and its	Ap	P	Viva Voce

	enforcement mechanisms in India			Observation of practical skills
CO6	To get to know about Penalties and legal implications associated with cyber crimes under Indian law	U	M	Instructor Created - Exams, Assignments
* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C) # - Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge (M)				

Module	Unit	Content	Hrs	Marks
I		Introduction to cyber space	**9**	**12**
	1	Cyber Space- Fundamental definitions	2	
	2	Jurisprudence and-Jurisdiction in Cyber Space	2	
	3	Need for IT act - Enforcement agencies	3	
	4	Introduction to cyber law and its relevance in the Indian context	2	
II		**Cyber Crimes and Criminals**	**9**	**12**
	5	Cyber crimes	2	
	6	Cyber Criminals and their Objectives	2	
	7	Cyber stalking; cyber pornography	2	
	8	Forgery and fraud; crime related to IPRs;	2	
	9	Phishing and Identity Theft	1	
III		**Indian Cyber law**	**9**	**14**
	10	Introduction to Indian Cyber Law	2	
	11	Cyber Crime vs Conventional Crime	2	
	12	Electronic Commerce and related issues	2	
	13	Overview of Intellectual Property rights	2	
	14	Computer Software and related IPR Issues	1	
IV		**Basics of IT law and its regulatory mechanisms**	**9**	**12**
	13	Key provisions of the Information Technology Act, 2000 related to cyber crimes and offenses	2	
	14	Regulatory Mechanisms and Enforcement	2	
	15	Overview of the Cyber Crime Investigation Cell (CCIC)	2	
	16	Understanding the process of reporting cyber crimes	2	
	17	Penalties and legal implications associated with cyber crimes under Indian law (basics only)	1	
V		**Hands-on :** **Practical Applications, Case Study and Course Project**	**9**	
	1	Social Media based Cyber crimes	2	
	2	Discussion on Emerging issues	2	
	3	Recent trends in digital marketing	3	
	4	Demonstrate how to use google web masters Indexing Using API	2	

References:

1. Cyber law –The Indian perspective by Pavan Duggal
2. Justice Yatindra Singh: Cyber Laws, Universal Law Publishing Co., New Delhi
3. Farouq Ahmed, Cyber Law in India, New Era publications, New Delhi

Correlation Levels:

Level	Correlation
-	Nil
1	Slightly / Low
2	Moderate / Medium
3	Substantial / High

Assessment Rubrics:

- Quiz / Assignment/ Quiz/ Discussion / Seminar
- Midterm Exam
- Programming Assignments (20%)
- Final Exam (70%)

Mapping of COs to Assessment Rubrics :

	Internal Exam	Assignment	Practical Evaluation	End Semester Examinations
CO 1	✓			✓
CO 2	✓	✓		✓
CO 3	✓			✓
CO 4	✓			✓
CO 5		✓		✓
CO6				✓

Programme	B. Sc. Computer Science				
Course Code	CSC4FV109(2)				
Course Title	Introduction to Content Management System				
Type of Course	VAC				
Semester	IV				
Academic Level	100-199				
Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	3	3	-	-	45
Pre-requisites	1. Familiarity with web content management systems (CMS). 2. Basic knowledge of internet technologies provides a foundation for learning web design.				
Course Summary	The course covers fundamental web design concepts including HTML and CMS principles, focusing on Drupal as a robust Content Management System. Students will learn to create and customize websites using Drupal, exploring its features such as content types, themes, and modules to build dynamic and interactive web pages.				

Course Outcomes (CO):

CO	CO Statement	Cognitive Level*	Knowledge Category#	Evaluation Tools used
CO1	Cultivate a robust understanding of web design fundamentals, laying a strong foundation for their journey into the dynamic world of digital design and development.	U	C	Assignment / Instructor-created exams / Quiz
CO2	Attain comprehensive knowledge and practical proficiency in Content Management Systems (CMS), empowering to navigate and excel in the ever-evolving landscape of digital content creation and management.	U	C	Assignment / Instructor-created exams / Quiz
CO3	Develop expertise in Drupal, a widely used CMS platform, gaining comprehensive understanding of its features, configuration, and installation processes, thus preparing them for proficient and innovative web development endeavors.	Ap	P	Practical Assignment / Instructor-created exams / Quiz
CO4	Impart a comprehensive understanding of website development using Drupal and facilitate the acquisition of expertise across various options within the Drupal ecosystem.	Ap	P	Practical Assignment / Instructor-created exams / Quiz
CO5	Gain an understanding of how to apply web design concepts to real-world scenarios, effectively designing and developing functional and aesthetically	C	P	Practical Assignment / Instructor-created exams /

	pleasing websites utilizing the Drupal CMS.			Quiz
CO6	Develop proficiency in advanced website management skills, including installing and configuring modules, managing menus, and more, to effectively navigate and optimize the functionality of websites built on the Drupal platform.	C	P	Practical Assignment / Instructor-created exams / Quiz
* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C) # - Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge (M)				

Detailed Syllabus

Module	Unit	Content	Hrs	Marks
I		**Introduction to Web Designing**	**8**	**10**
	1	Basics of Web Designing -World Wide Web (WWW), W3C, Web Browser	1	
	2	Web Server, Web Hosting, Web Pages	1	
	3	DNS, URL	2	
	4	Overview of HTML (Concept only) and its role in Web Development	2	
	5	Open Source S/W, Open Source vs Closed Source Software, Open Source Licenses (Concept only)	2	
II		**Introduction to CMS**	**6**	**10**
	6	Introduction to Content Management Systems (CMS) - Features of CMS	2	
	7	Web Content Management System	2	
	8	Components of Content Management System	2	
III		**Introduction to Drupal**	**10**	**15**
	10	Drupal - Features, Advantages and Disadvantages	1	
	11	Installation and Configuration	1	
	12	Content types and Field	2	
	13	Drupal Architecture	1	
	14	User Management, Managing Comments	2	
	15	Creating and Customizing Themes	3	
IV		**Building Website**	**12**	**15**
	16	Website Development - Working with Templates and Template files	2	
	17	Articles, Creating Web Forms	2	
	18	Managing blocks, Add Links to Blocks, Moving Elements within Block	2	
	19	Blocks and Regions	2	
	20	Creating and Customizing Views	1	
	21	Installing and Configuring Modules	1	
	22	Static Pages, Creating Pages, Menu Management.	2	
V		**Open Ended Module – Website Development**	**9**	
	23	Develop a simple Website using Drupal.	9	

Mapping of COs with PSOs and POs:

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PO1	PO2	PO3	PO4	PO5	PO6
CO 1	1	3	1	1	3	1						
CO 2	1	3	2	1	3	1						
CO 3	1	3	1	1	3	2						
CO 4	1	3	3	1	3	2						
CO 5	3	3	3	1	3	2						
CO 6	1	3	3	1	3	2						

Correlation Levels:

Level	**Correlation**
-	Nil
1	Slightly / Low
2	Moderate / Medium
3	Substantial / High

Assessment Rubrics:

- Quiz / Assignment/ Quiz/ Discussion / Seminar
- Midterm Exam
- Programming Assignments (20%)
- Final Exam (70%)

Mapping of COs to Assessment Rubrics:

	Internal Exam	Assignment	Practical Evaluation	End Semester Examinations
CO 1	✓	✓		✓
CO 2	✓	✓		✓
CO 3	✓	✓		✓
CO 4	✓	✓		✓
CO 5	✓	✓		✓
CO 6	✓	✓		✓

References:

1. Jennifer Campbell, Jennifer T Campbell, Web Design: Introductory, Course Technology.

2. Jason Beaird and Alex Walker, The Principles of Beautiful Web Design, SitePoint.

3. Bob Boiko, Content Management Bible, Wiley.

4. Daniel Sipos, Drupal 9 Module Development, Packt Publishing Limited.

Programme	B. Sc. Computer Science				
Course Code	CSC5FS112				
Course Title	Introduction to Digital Marketing				
Type of Course	SEC				
Semester	V				
Academic Level	100 - 199				
Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	3	3	-	3	45
Pre-requisites	1. Basic Computer Literacy 2. Familiarity with Online Platforms 3. Willingness to Learn				
Course Summary	Introduction to Digital Marketing" provides students with a foundational understanding of key concepts and techniques in the rapidly evolving field of digital marketing. Through engaging lectures. Students will explore various digital marketing channels, including search engine optimization (SEO), social media marketing, email marketing, and content marketing				

Course Code		Course Title Introduction to Digital Marketing		
Credit 3		Duration 45 hrs		
Sl. NO:	Course Outcome	Cognitive level *	Knowledge category #	Evaluation Tools used
CO1	To understand the concept of digital marketing and its integration with traditional marketing	U	C	Instructor-Create Exams or Quiz
CO2	To understand customer value journey in digital context and behaviour of online consumers	A	P	Discussions and Quizzes

CO3	To examine various tactics for enhancing a website's position and ranking with search engines	U	F	Instructor created exams or Home assignments
CO4	To Identify and differentiate between various digital marketing channels, including SEO, social media, email, and content marketing.	A ,E	P	Discussions, Quizzes
CO5	To get overall idea in implementing basic digital marketing strategies to enhance online visibility and engagement.	Ap	P	Viva Voce Observation of practical skills
CO6	To get to know about ethical considerations and best practices in digital marketing, including privacy, data protection, and consumer trust	U	M	Instructor Created - Exams, Assignments
* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C) # - Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge (M)				

Detailed Syllabus

Module	Unit	Content	Hrs	Marks
I		Digital Marketing Basics	**9**	**12**
	1	Overview of digital marketing	2	
	2	Importance of digital marketing for businesses	2	
	3	Introduction to key digital marketing channels (SEO, social media, email marketing)	3	
	4	Basics of creating a digital marketing strategy	2	
II		**Content Marketing & Social Media**	**9**	**12**
	5	Content Marketing Fundamentals	2	
	6	Content Strategy Development	2	
	7	Content Creation for Different Platforms	2	
	8	Introduction to Social Media Marketing & keyword Optimization	2	
	9	Social Media Strategy & Community Management	1	
III		**Search Engine Optimization (SEO) & Paid Advertising**	**9**	**14**
	10	Introduction to Search Engine Optimization	2	

	11	On-page and Off-page SEO Techniques	2	
	12	Search Engine Marketing (SEM) Fundamentals	2	
	13	Pay-Per-Click (PPC) Advertising with Google Ads	2	
	14	Social Media Advertising Platforms	1	
IV		**Web Analytics & Emerging Trends**	**9**	**12**
	13	Introduction to Web Analytics & Key Metrics	2	
	14	Using Analytics Tools for Data-Driven Decision Making	2	
	15	Conversion Tracking & Optimization	2	
	16	Emerging Trends in Digital Marketing	2	
	17	The Future of Marketing	1	
V		**Hands-on :** **Practical Applications, Case Study and Course Project**	**9**	
	1	Social Media Marketing-Social media Channels	2	
	2	Leveraging social media for brand conversions and buzz	2	
	3	Recent trends in digital marketing	3	
	4	Demonstrate how to use google web masters Indexing Using API	2	

References:

1. Digital Marketing for Dummies by Ryan DeWald
2. MARKETING 4.0 Moving from Traditional to Digital PHILIP KOTLER HERMAWAN KARTAJAYA IWAN SETIAWAN
3. Ryan, D. (2014). Understanding Digital Marketing: Marketing Strategies for Engaging the Digital Generation, Kogan Page Limited
4. Taxmanns - Digital Marketing - Satinder Kumar, Supereet Kaur
5. Social Media Marketing 2024 - Mastering New Trends & Strategies for Online Success - Robert Hill

Correlation Levels:

Level	Correlation
-	Nil
1	Slightly / Low
2	Moderate / Medium
3	Substantial / High

Assessment Rubrics:

- Quiz / Assignment/ Quiz/ Discussion / Seminar
- Midterm Exam
- Programming Assignments (20%)
- Final Exam (70%)

Mapping of COs to Assessment Rubrics :

	Internal Exam	Assignment	Practical Evaluation	End Semester Examinations
CO 1	✓			✓
CO 2	✓	✓		✓
CO 3	✓			✓
CO 4	✓			✓
CO 5		✓		✓
CO6				✓

Model Question Papers/ Major

FIRST SEMESTER (CUFYUGP) DEGREE EXAMINATIONS, OCTOBER 2024

Computer Science

CSC1CJ101 - Fundamentals of Computers and Computational Thinking

(2024 Admissions)

Time: Two Hours **Maximum: 70 Marks**

Section A

[Answer All. Each question carries 3 marks] (Ceiling 24 Marks)

1. Briefly describe the historical development of computers mentioning two key figures and their contributions.
2. Explain the concept of the Von Neumann architecture.
3. Convert the following numbers from decimal to binary: (a) 25, (b) 100.
4. Differentiate between active and passive electronic components. Provide an example of each.
5. What is the function of a motherboard? List four key components on a motherboard.
6. Distinguish between application software and system software. Give an example of each.
7. Briefly explain the role of an operating system in a computer system.
8. Define the term "computational thinking."
9. What are the steps involved in problem decomposition?
10. Explain the difference between inductive and deductive reasoning.

Section B

[Answer All. Each question carries 6 marks] (Ceiling 36 Marks)

11. Describe the evolution of computers from first generation to present day, highlighting the key features of each generation.
12. Explain the concept of digital codes with reference to Gray code and BCD.
13. Briefly explain the working principle of a transistor.
14. Differentiate between RAM and ROM. Explain the different types of RAM.
15. Discuss the different types of operating systems and their characteristics.
16. Explain the concept of booting with reference to POST and UEFI/Legacy BIOS.
17. Describe the four key pillars of computational thinking.
18. Explain the importance of algorithms in solving problems. Discuss the qualities of a good algorithm.

Section C

[Answer any one. Each question carries 10 marks] (1 x 1 = 10 Marks)

20. a) Discuss the contributions of John von Neumann to the field of computing.

    b) Briefly explain the concept of Quantum Processing Units (QPU) and their potential applications.

21. a) Describe the various components of a computer system and their functionalities.

    b) Explain the need for device drivers in a computer system.

***

FIRST SEMESTER (CUFYUGP) DEGREE EXAMINATIONS, OCTOBER 2024

Computer Science

CSC1FM103 - Data Analysis and Visualization using Spreadsheets

(2024 Admissions)

Time: Two Hours **Maximum: 70 Marks**

Section A

[Answer All. Each question carries 3 marks] (Ceiling 24 Marks)

1. Define the terms "worksheet" and "workbook" in the context of Excel.
2. Explain three formatting options available for cells in Excel and briefly describe their applications.
3. How can you insert a new row and a new column within an Excel sheet?
4. Write a formula to calculate the average of a range of cells (A1:A10) in Excel.
5. Explain the purpose and benefits of data validation in Excel.
6. Differentiate between absolute and relative cell referencing with an example for each.
7. Describe the concept of Autofill and give an example of how it can be used in Excel.
8. Explain the steps involved in filtering data based on a specific criterion in Excel.
9. Describe the process of importing data from a text file into an Excel spreadsheet.
10. List two commonly used functions for applying conditional formatting in Excel.

Section B

[Answer All. Each question carries 6 marks] (Ceiling 36 Marks)

11. A dataset contains duplicate entries. Describe the steps involved in removing these duplicate rows in Excel.
12. You are given a dataset with sales figures for different regions. Explain how you would create a pivot table to analyse trends in sales across these regions.
13. Explain the concept of "What-If Analysis" in Excel and provide an example of how it can be used to support decision-making.
14. Write a formula using the VLOOKUP function to find the product price based on a product code in another table.
15. Describe three different chart types suitable for visualizing data in Excel and explain when you might use each type.
16. Explain the steps involved in creating a chart from a selected data range in Excel.
17. How can you format chart elements like titles, labels, and data points in Excel to improve clarity and presentation?

18. What is a dashboard in Excel, and what are the benefits of using dashboards for data analysis and communication?

Section C

[Answer any one. Each question carries 10 marks] (1 x 1- = 10 Marks)

19. You are provided with a large dataset containing customer information and sales data.
    a. Describe how you would utilize advanced features like data filtering, sorting, and pivot tables to identify the top 5 customers by sales in a specific region for the past year.
    b. Create a visually appealing dashboard in Excel that summarizes key customer and sales data, including a chart to represent the top-selling products.

(or)

20. Explain the concept of macros in Excel and discuss their potential benefits and drawbacks. Briefly describe the steps involved in creating a simple macro to automate a repetitive task.

***

SECOND SEMESTER (CUFYUGP) DEGREE EXAMINATIONS, OCTOBER 2024

Computer Science

CSC2FM106- Digital Empowerment Through Ethical Standards

(2024 Admissions)

Time: Two Hours **Maximum: 70 Marks**

Section A

[Answer All. Each question carries 3 marks] (Ceiling 24 Marks)

1. Briefly discuss the challenges faced in the pre-digital age.
2. Explain the concept of digital transformation and its significance.
3. Briefly describe two key digital technologies and their potential benefits.
4. Explain the importance of Digital India initiatives in empowering citizens.
5. Describe the role of Akshaya Project in Kerala's digital emergence.
6. List four digital tools for data sharing and collaboration.
7. Briefly explain how online learning platforms like Coursera can enhance your professional growth.
8. What are the ethical considerations one should keep in mind while creating content online?
9. Define the term "Intellectual Property Rights" (IPR).
10. Briefly explain the concept of cybercrime.

Section B

[Answer All. Each question carries 6 marks] (Ceiling 36 Marks)

11. Compare and contrast the concepts of digitization and digitalization.
12. Explain the working principle of Cloud Computing with its advantages and limitations.
13. Describe the Internet of Things (IoT) and its potential applications in different sectors like healthcare or agriculture.
14. Discuss the four pillars of Digital Emergence in Kerala.
15. Explain the role of K-DISC (Kerala Development and Innovation Strategic Council) in digital empowerment.
16. Describe how online collaboration tools like Google Sheets can be used for data analysis and visualization.
17. Explain the concept of cyber security and differentiate between authentication and authorization.
18. Briefly discuss the different types of cyber attacks and how to protect yourself online.

Section C

[Answer any one. Each question carries 10 marks] (1 x 1 = 10 Marks)

19. a) Analyze the importance of digital literacy in the 21st century. Discuss how ethical considerations shape responsible digital citizenship.

b) Using a real-world example, critically evaluate the impact of a digital mission or service on a specific community.

20. a) Explain the concept of Blockchain technology and its potential applications in various fields.

b) Discuss the legal framework for cyber security in India. Briefly explain some key cyber laws.

***

SECOND SEMESTER (CUFYUGP) DEGREE EXAMINATIONS, OCTOBER 2024

Computer Science

CSC2CJ101- Fundamentals Of Programming (C Language)

(2024 Admissions)

Time: Two Hours **Maximum: 70 Marks**

Section A

[Answer All. Each question carries 3 marks] (Ceiling 24 Marks)

1. Explain the importance of algorithms in problem-solving. Provide an example to illustrate your answer.
2. Define pseudocode. How does it aid in algorithm development? Give an example of a simple algorithm represented using pseudocode.
3. Discuss the significance of flowchart symbols in algorithm representation. Draw a flowchart to represent the algorithm for finding the factorial of a number.
4. What is Raptor, and how does it relate to programming languages? Explain with an example.
5. Explain the structure of a C program. Give a brief description of each component.
6. Discuss the concept of operators in C programming. Provide examples of arithmetic, logical, and relational operators.
7. What are selection statements in C? Differentiate between 'if', 'if-else', and 'switch' statements with suitable examples.
8. Define arrays in C. Explain the difference between one-dimensional and two-dimensional arrays with examples.
9. Briefly explain the basic string handling functions in C.
10. Define structures in C. Discuss the concept of processing-period operator with an example.

Section B

[Answer All. Each question carries 6 marks] (Ceiling 36 Marks)

11. Write a C program to find the sum of all elements in an array.
12. Explain the difference between recursion and iteration. Provide examples to illustrate both concepts.
13. What are function prototypes in C? Why are they necessary? Provide an example.
14. Discuss the advantages of using functions in C programming.
15. How are pointers declared in C? Give examples to illustrate pointer declarations.
16. Write a C function to reverse a given string.
17. Discuss the concept of formal and actual parameters in C functions. Provide examples to explain each.
18. Explain the concept of unions in C. How are they different from structures? Provide an example.

Section C

[Answer any one. Each question carries 10 marks] (1 x 1 = 10 Marks)

19. Develop an algorithm to find the largest element in an array. Implement the algorithm in C and explain each step of your solution.
20. Write a C program to implement a simple calculator that can perform addition, subtraction, multiplication, and division operations. Ensure proper error handling for division by zero.

***

SECOND SEMESTER (CUFYUGP) DEGREE EXAMINATIONS, OCTOBER 2024

Computer Science

CSC3CJ201- Software Project Management

(2024 Admissions)

Time: Two Hours **Maximum: 70 Marks**

Section A

[Answer All. Each question carries 3 marks] (Ceiling 24 Marks)

1. Define the term "Software Project Management" and explain its significance in software development.
2. Differentiate between the Waterfall model and the Incremental model of the Software Development Life Cycle (SDLC).
3. Briefly explain the concept of Agile Development and its core principles.
4. What is the purpose of the requirement engineering process?
5. Describe the key elements of data design in software development.
6. Explain the concept of UML and its role in software design.
7. Briefly discuss the importance of project planning in software project management.
8. Explain the Work Breakdown Structure (WBS) and its role in project management.
9. Define the term "Critical Path" in project scheduling.
10. Briefly explain the difference between proactive and reactive risk management strategies.

Section B

[Answer All. Each question carries 6 marks] (Ceiling 36 Marks)

11. Compare and contrast the Agile and Waterfall methodologies for software development. Discuss the advantages and disadvantages of each approach.
12. Describe the different phases involved in the design process for software development.
13. Explain the concept of architectural design using Data Flow Diagrams (DFDs).
14. Discuss the various techniques used for software estimation during project planning.
15. Explain the concept of network diagrams in project scheduling and their advantages over Gantt Charts.
16. Briefly describe the Program Evaluation and Review Technique (PERT) and its applications in project management.
17. Discuss different levels of software testing used to ensure software quality.
18. Explain the concept of white-box testing and black-box testing, providing examples for each.

Section C

[Answer any one. Each question carries 10 marks] (1 x 1 = 10 Marks)

19. a) Discuss the different phases of the Agile development methodology. Explain how an iterative approach contributes to successful project delivery.

    b) Consider a real-world software project (e.g., library management system, e-commerce platform) and identify potential risks associated with the project. Explain how you would develop a risk management plan (RMMM) to mitigate these risks.

20. a) Describe the various quality assurance processes and methodologies used in software development.

    b) Create a Gantt chart for a simple software development project outlining key tasks, durations, and dependencies. Briefly explain your reasoning behind the scheduling decisions.

***

SECOND SEMESTER (CUFYUGP) DEGREE EXAMINATIONS, OCTOBER 2024

Computer Science

CSC3CJ202- Data Structures and Algorithm

(2024 Admissions)

Time: Two Hours **Maximum: 70 Marks**

Section A

[Answer All. Each question carries 3 marks] (Ceiling 24 Marks)

1. Define data structure. Explain the difference between a data type and a data structure.
2. Discuss the applications of stacks in computer science, with relevant examples.
3. What are the different types of linked lists? Explain the concept of a circular linked list.
4. Explain the process of converting an infix expression to postfix using a stack with an example.
5. Define a queue. Discuss the applications of queues in real-world scenarios.
6. Compare and contrast a binary tree and a binary search tree.
7. How does a graph differ from a tree? Provide examples of both directed and undirected graphs.
8. Discuss the importance of hashing in data structures. Explain collision resolution techniques with examples.
9. Explain the difference between a data type and a data structure. Provide examples to illustrate each.
10. Describe the characteristics and real-world applications of a stack data structure

Section B

[Answer All. Each question carries 6 marks] (Ceiling 36 Marks)

11. Implement a stack using an array in C++. Include functions for push, pop, and display.
12. Write an algorithm to perform a depth-first traversal of a binary tree. Illustrate with an example.
13. Explain the concept of a doubly linked list. Discuss its advantages over a singly linked list.
14. Describe the process of performing a merge sort algorithm on an array. Provide a step-by-step explanation.
15. Discuss the working principle of the quick sort algorithm. Provide an example for better understanding.
16. Implement a queue using a linked list in Java. Include functions for enqueue, dequeue, and display.

17. Explain the process of binary search and its time complexity. Provide an example demonstrating its application.
18. Discuss the concept of hash tables and their role in efficient data retrieval. Illustrate with a suitable example.

Section C

[Answer any one. Each question carries 10 marks] (1 x 1 = 10 Marks)

19. Analyze the time complexity of the quicksort algorithm. Discuss its advantages and limitations compared to other sorting algorithms.
20. Investigate the collision resolution techniques used in hashing. Compare and contrast open hashing (chaining) and closed hashing (probing) methods

***

SECOND SEMESTER (CUFYUGP) DEGREE EXAMINATIONS, OCTOBER 2024

Computer Science

CSC4CJ203- Database Management System

(2024 Admissions)

Time: Two Hours **Maximum: 70 Marks**

Section A

[Answer All. Each question carries 3 marks] (Ceiling 24 Marks)

1. Define the term "Database Management System" (DBMS) and explain its key characteristics.
2. Differentiate between a file system and a database management system.
3. Briefly describe the three-schema architecture in a DBMS.
4. Explain the concept of entity-relationship modeling (ER Model) in database design.
5. What are the different cardinalities in an entity-relationship diagram?
6. Define the term "normalization" in the context of relational databases.
7. Briefly explain the concept of Data Definition Language (DDL) in SQL.
8. What are the functionalities of Data Manipulation Language (DML) in SQL?
9. Explain the concept of transactions in a database system.
10. Briefly describe the ACID properties of transactions.

Section B

[Answer All. Each question carries 6 marks] (Ceiling 36 Marks)

11. Discuss the advantages of using a DBMS approach for data management compared to a traditional file system approach.
12. Explain the concept of relational data model with examples of domains, attributes, tuples, and relations.
13. Describe the different normalization forms (1NF, 2NF, 3NF) used in relational database design.
14. Write an SQL query to create a table named "Students" with attributes "StudentID" (integer, primary key), "Name" (varchar), and "Program" (varchar).
15. Explain how to retrieve data from multiple tables using JOIN operations in SQL.
16. Briefly describe the concept of views in SQL and their benefits.
17. Discuss the concept of concurrency control in a database system and its importance.
18. Explain the concept of NoSQL databases and differentiate them from relational databases. Briefly describe two main types of NoSQL databases (e.g., Key-value, Document).

Section C

[Answer any one. Each question carries 10 marks] (1 x 1 = 10 Marks)

19. a) Design an Entity-Relationship (ER) diagram for a library management system considering entities like Books, Authors, Members, and Loans. Include relevant attributes and relationships between entities.

    b) Write SQL queries to perform the following operations in a library database:

    i) Insert a new book record.

    ii) Search for books by a specific author name.

    iii) Update the availability status of a borrowed book upon return.

20. a) Discuss the ACID properties (Atomicity, Consistency, Isolation, Durability of transactions and their role in maintaining data integrity in a database system.

    b) Briefly explain the concept of transaction logs and their significance in transaction recovery.

    c) Describe two-phase locking as a concurrency control technique in databases.

***

FOURTH SEMESTER (CUFYUGP) DEGREE EXAMINATIONS, OCTOBER 2024

Computer Science

CSC4CJ204- Python Programming

(2024 Admissions)

Time: Two Hours **Maximum: 70 Marks**

Section A

[Answer All. Each question carries 3 marks] (Ceiling 24 Marks)

1. Explain the importance of indentation in Python programming. Provide an example to illustrate its significance.
2. Define the following terms: Identifiers, Keywords, and Variables in Python. Provide examples for each.
3. Describe the different types of function arguments in Python. Give examples to illustrate each type.
4. What is the scope and lifetime of variables in Python? Explain with suitable examples.
5. Discuss the decision-making structures available in Python with examples.
6. Explain the concept of looping structures in Python. Provide examples of for and while loops.
7. How are strings indexed and sliced in Python? Provide examples to demonstrate string indexing and slicing.
8. Describe the operations and methods available for manipulating lists in Python. Provide examples for each operation.
9. Explain the concept of operator precedence and associativity in Python. Provide examples to illustrate their significance.
10. Discuss the importance of indentation in Python programming. How does it affect the execution of code? Provide an example..

Section B

[Answer All. Each question carries 6 marks] (Ceiling 36 Marks)

11. Define a function in Python and explain its components. Write a Python function to calculate the factorial of a given number.
12. Discuss the advantages of using modules in Python. Write a Python program to demonstrate the use of a user-defined module.
13. Explain the operations and methods available for dictionaries in Python. Provide examples for each operation.
14. Describe the creation and operations of sets in Python. Write a Python program to perform set operations such as union, intersection, and difference.
15. Explain the advantages of NumPy arrays over traditional Python lists. Create a NumPy array and perform arithmetic operations on it.

16. Discuss the basic plotting techniques available in Matplotlib. Provide examples of at least two types of plots.
17. Define Pandas Series and Pandas DataFrames. Write a Python program to create a Pandas Series from a dictionary.
18. Explain the concept of fancy indexing in NumPy. Provide an example demonstrating its usage.

Section C

[Answer any one. Each question carries 10 marks] (1 x 1 = 10 Marks)

19. Discuss the importance of broadcasting in NumPy arrays. Provide examples to illustrate how broadcasting works.
20. Explain the process of creating histograms and pie charts using Matplotlib. Provide examples demonstrating the creation of both types of plots.

***

SECOND SEMESTER (CUFYUGP) DEGREE EXAMINATIONS, OCTOBER 2024

Computer Science

CSC4CJ205- Computer Networks

(2024 Admissions)

Time: Two Hours **Maximum: 70 Marks**

Section A

[Answer All. Each question carries 3 marks] (Ceiling 24 Marks)

1. Define the term "computer network" and differentiate between the Internet and an Intranet.
2. Briefly describe three common network topologies.
3. Explain the concept of a layered network architecture model using an example.
4. Differentiate between analog and digital signals in data transmission.
5. Briefly explain the concept of multiplexing and its different types.
6. What are the different types of errors that can occur during data transmission?
7. Briefly describe the concept of Cyclic Redundancy Check (CRC) for error detection.
8. Define the term "MAC address" and its significance in data link layer communication.
9. Differentiate between CSMA/CD and CSMA/CA protocols used in multiple access networks.
10. Briefly explain the function of a router and its role in internetworking.

Section B

[Answer All. Each question carries 6 marks] (Ceiling 36 Marks)

11. Discuss the different layers of the OSI reference model and their functionalities.
12. Explain the concept of data encapsulation in layered network communication protocols.
13. Briefly describe the characteristics of twisted-pair cable and coaxial cable as transmission media.
14. Explain the concept of error correction techniques and differentiate them from error detection techniques.
15. Briefly describe the stop-and-wait and go-back-n error recovery protocols used in data link layer communication.
16. Explain the functionalities of network devices like repeaters, bridges, and gateways.
17. Discuss the difference between IPv4 and IPv6 addressing schemes.
18. Briefly explain the concept of subnet masks and their role in network addressing.

Section C

[Answer any one. Each question carries 10 marks] (1 x 1 = 10 Marks)

19. a) Explain the functionalities of TCP and UDP protocols in the transport layer with their key differences.

    b) Discuss various congestion control mechanisms used in TCP to avoid network congestion.

20. a) Describe the concept of the Domain Name System (DNS) and its role in internetworking.

    b) Briefly explain the functionalities of common application layer protocols like HTTP, FTP, and SMTP.

***

FIFTH SEMESTER (CUFYUGP) DEGREE EXAMINATIONS, OCTOBER 2024

Computer Science

CSC5CJ301- Data Mining

(2024 Admissions)

Time: Two Hours **Maximum: 70 Marks**

Section A

[Answer All. Each question carries 3 marks] (Ceiling 24 Marks)

1. Define the term "data mining" and differentiate it from the Knowledge Discovery in Databases (KDD) process.
2. Briefly describe different types of data sources used in data mining tasks.
3. What kinds of patterns can be mined from data using data mining techniques?
4. Explain the role of statistics and machine learning in data mining.
5. What is the significance of data cleaning in data preprocessing?
6. Describe how missing values are handled during data cleaning.
7. Briefly explain the concept of data warehousing and its relation to data mining.
8. What is the purpose of data reduction techniques in data mining?
9. Define the term "association rule mining" and provide an example.
10. Briefly explain the concept of decision tree learning in classification tasks.

Section B

[Answer All. Each question carries 6 marks] (Ceiling 36 Marks)

11. Discuss the various phases involved in the KDD process, highlighting the importance of each phase.
12. Explain the concept of data integration and address potential challenges associated with it.
13. Briefly describe the Principal Component Analysis (PCA) technique for data dimensionality reduction.
14. Explain the Apriori algorithm used for frequent itemset mining in association rule learning.
15. Discuss the concept of attribute selection measures used in decision tree learning algorithms.
16. Explain the working principle of K-Means clustering algorithm with its advantages and limitations.
17. Differentiate between hierarchical clustering and density-based clustering techniques. Briefly explain one example of each.
18. Explain the concept of outlier detection and its importance in data analysis.

Section C

[Answer any one. Each question carries 10 marks] (1 x 1 = 10 Marks)

19. a) Consider a real-world scenario like customer purchase data from an online retail store. Explain how data mining techniques can be applied to extract valuable insights for improved marketing strategies.

    b) Describe the process of data mining using a sample dataset (e.g., movie ratings, weather data). Explain the steps involved in data pre-processing, choosing a suitable data mining algorithm, and interpreting the results.

20. a) Discuss the ethical considerations and challenges associated with data mining in the context of privacy and security.

    b) Briefly explain the concept of recommender systems and their applications.

    c) Describe the role of data mining in different domains like healthcare, finance, or social media.

***

FIFTH SEMESTER (CUFYUGP) DEGREE EXAMINATIONS, OCTOBER 2024

Computer Science

CSC5CJ302- Object Oriented Programming (Java)

(2024 Admissions)

Time: Two Hours **Maximum: 70 Marks**

Section A

[Answer All. Each question carries 3 marks] (Ceiling 24 Marks)

1. Define the key concepts of object-oriented programming (OOP) and explain its advantages over procedural programming.
2. Differentiate between primitive data types and reference data types in Java.
3. Briefly explain the concept of operators in Java and provide examples of different types of operators.
4. What is the purpose of access modifiers in Java?
5. Differentiate between constructor overloading and method overloading in Java.
6. Briefly explain the concept of inheritance in Java and its types.
7. What are exceptions in Java?
8. Briefly describe the use of try...catch...finally blocks for exception handling.
9. Explain the concept of threads in Java and their different states.
10. Briefly describe the Model-View-Controller (MVC) pattern used in GUI applications.

Section B

[Answer All. Each question carries 6 marks] (Ceiling 36 Marks)

11. Write a Java program to demonstrate the use of a scanner class to read user input for two numbers and then display their sum and difference.
12. Explain the concept of multidimensional arrays in Java with an example of declaring and initializing a 2D array.
13. Describe the concept of method overriding in Java and provide an example of a base class and a derived class with overridden methods.
14. Write a Java program to demonstrate the concept of interface implementation where an interface defines abstract methods and a class implements them.
15. Briefly explain the working principle of try-with-resources statement for handling resources in Java.
16. Write a Java program to read data from a text file line by line and display the contents on the console.
17. Explain the concept of database connectivity using JDBC in Java and the steps involved in establishing a connection to a database.

18. Write a Java program using JDBC to create a table in a database and insert a new record into the table.

Section C

[Answer any one. Each question carries 10 marks] (1 x 1 = 10 Marks)

19. a) Design a Java class named Student with attributes like name, roll number, and marks. Include methods to get and set these attributes.

    b) Write a separate Java class named StudentDemo with a main method to create an object of the Student class, set its attributes, and then display the student information using getter methods.

20. a) Develop a simple Java GUI application using Swing that displays a text field and a button. When the button is clicked, the text entered in the text field should be displayed in a label.

    b) Briefly explain the concept of layout managers in Swing and describe two common layout managers used for arranging GUI components.

***

FIFTH SEMESTER (CUFYUGP) DEGREE EXAMINATIONS, OCTOBER 2024

Computer Science

CSC5CJ303- Full Stack Web Development

(2024 Admissions)

Time: Two Hours **Maximum: 70 Marks**

Section A

[Answer All. Each question carries 3 marks] (Ceiling 24 Marks)

1. Define the term "full stack web development" and explain the different technologies involved in this field.
2. Differentiate between semantic tags and non-semantic tags in HTML. Provide examples of each.
3. Briefly explain the concept of CSS selectors used to style HTML elements.
4. What is the purpose of the CSS Box Model?
5. Explain the difference between var, let, and const keywords used for declaring variables in JavaScript.
6. Briefly describe the concept of data types in JavaScript and provide examples of primitive data types.
7. What are the advantages of using functions in JavaScript code?
8. Define the term "Node.js" and explain its role in server-side development.
9. Briefly describe the concept of event handling in Node.js.
10. Explain the difference between SQL and NoSQL databases.

Section B

[Answer All. Each question carries 6 marks] (Ceiling 36 Marks)

11. Write HTML code to create a responsive web page with a header, navigation bar, main content section, and a footer. Use appropriate semantic tags and CSS styling.
12. Explain the concept of DOM (Document Object Model) and how JavaScript interacts with the DOM to manipulate web page elements.
13. Write a JavaScript function to check if a given number is even or odd.
14. Briefly describe the concept of asynchronous programming in Node.js and its benefits.
15. Explain the concept of Express.js framework and its role in building web applications using Node.js.
16. Briefly describe the concept of components and their lifecycle methods in React.js.
17. Explain the difference between state and props in React components.
18. Briefly describe the functionalities offered by MongoDB as a NoSQL database.

Section C

[Answer any one. Each question carries 10 marks] (1 x 1 = 10 Marks)

19. a) Develop a simple single-page application (SPA) using React.js that displays a list of products with their names and prices. Allow users to add items to a shopping cart and display the total cart value.

    b) Briefly explain the concept of routing in React applications and its importance.

20. a) Design and develop a web application using Node.js and Express.js that allows users to create and manage a list of tasks. Users should be able to add new tasks, mark tasks as complete, and view a list of pending and completed tasks. Explain the functionalities used in your code. b) Briefly describe the concept of RESTful APIs and their advantages in web development.

***

FIFTH SEMESTER (CUFYUGP) DEGREE EXAMINATIONS, OCTOBER 2024

Computer Science

CSC6CJ304- Digital Electronics and Computer Architecture

(2024 Admissions)

Time: Two Hours **Maximum: 70 Marks**

Section A

[Answer All. Each question carries 3 marks] (Ceiling 24 Marks)

1. Convert the decimal number 23 to binary and explain the steps involved.
2. Briefly describe the concept of 2's complement representation in binary numbers.
3. Explain the truth table and logic symbol of a NAND gate.
4. Derive the simplified Boolean expression for a XOR gate using basic gates.
5. Briefly explain the concept of K-maps and their use in simplifying logic expressions.
6. What is a combinational logic circuit? Provide an example.
7. Briefly describe the operation of a D flip-flop with a truth table.
8. Differentiate between synchronous and asynchronous counters.
9. Define the term "computer architecture" and its basic components.
10. Explain the difference between RISC and CISC processor architectures.

Section B

[Answer All. Each question carries 6 marks] (Ceiling 36 Marks)

11. Perform the following binary arithmetic operations: a) Add 1101 (binary) and 1010 (binary) using the 2's complement method. b) Subtract 1011 (binary) from 1100 (binary) using the 2's complement method.
12. Design a logic circuit using basic gates (AND, OR, NOT) to implement the following Boolean expression: Y = A'B + AB'C
13. Simplify the following Boolean expression using a K-map: F(A, B, C, D) = Σm(0, 1, 2, 4, 5, 7, 8, 10, 12, 13, 14, 15)
14. Explain the working principle of a full adder circuit with a truth table.
15. Briefly describe the operation of a 4x1 multiplexer with a diagram.
16. Explain the concept of a synchronous counter and design a mod-4 counter using D flip-flops.
17. Briefly describe the role of the control unit in a computer and its functionalities.
18. Explain the concept of pipelining in a processor and its benefits.

Section C

[Answer any one. Each question carries 10 marks] (1 x 1 = 10 Marks)

19. a) Design a combinational logic circuit using basic gates to convert a 4-bit binary number to its corresponding gray code equivalent. Explain the logic behind your design.

    b) Briefly describe the concept of memory hierarchy in a computer system and its importance.

20. a) Explain the concept of a microprogrammed control unit in a processor and its advantages over a hardwired control unit.

    b) Briefly describe different types of memory organization within the memory hierarchy.

    c) Explain the concept of Direct Memory Access (DMA) for data transfer between a processor and I/O devices.

***

SIXTH SEMESTER (CUFYUGP) DEGREE EXAMINATIONS, OCTOBER 2024

Computer Science

CSC6CJ305- Principles of Operating System

(2024 Admissions)

Time: Two Hours **Maximum: 70 Marks**

Section A

[Answer All. Each question carries 3 marks] (Ceiling 24 Marks)

1. Briefly describe the history and evolution of operating systems.
2. Explain the main objectives and functions of an operating system.
3. Define the term "process" in operating systems and explain different process states.
4. Briefly describe the concept of a Process Control Block (PCB) and its contents.
5. Differentiate between preemptive and non-preemptive scheduling algorithms in process management.
6. Briefly explain the concept of inter-process communication (IPC) mechanisms.
7. What is a critical section in process synchronization?
8. Explain the concept of semaphores used for process synchronization.
9. Define deadlock in an operating system and explain the necessary conditions for deadlock to occur.
10. Briefly describe methods for handling deadlocks (prevention, avoidance, detection & recovery).

Section B

[Answer All. Each question carries 6 marks] (Ceiling 36 Marks)

11. Compare and contrast short-term and long-term scheduling algorithms used in process management.
12. Explain the concept of the First Come First Served (FCFS) scheduling algorithm with an example.
13. Briefly describe the working principle of the Shortest Job First (SJF) scheduling algorithm. Why is SJF not always practical?
14. Explain the concept of semaphores with a code example demonstrating their use for process synchronization.
15. Consider the classical Reader-Writer problem in process synchronization. Describe a solution using semaphores.
16. Differentiate between contiguous and non-contiguous memory allocation techniques.
17. Explain the concept of paging as a memory management technique with a diagram.
18. Briefly describe the concept of virtual memory and its benefits over physical memory.

Section C

[Answer any one. Each question carries 10 marks]   (1 x 1 = 10 Marks)

19. a) Write a shell script in Linux that takes two numbers as command-line arguments and then calculates their sum, difference, product, and quotient.

    b) Briefly explain the concept of shell scripting and its advantages.

20. a) Explain the concept of segmentation as a memory management technique with a diagram. Discuss the advantages and disadvantages of segmentation compared to paging.

    b) Describe various I/O redirection operators used in Linux shell scripting (>, <, >>, <<). Provide examples of their usage.

    c) Explain the working principle of the ping command in Linux and its purpose.

***

SIXTH SEMESTER (CUFYUGP) DEGREE EXAMINATIONS, OCTOBER 2024

Computer Science

CSC6CJ306- Artificial Intelligence & Machine Learning

(2024 Admissions)

Time: Two Hours **Maximum: 70 Marks**

Section A

[Answer All. Each question carries 3 marks] (Ceiling 24 Marks)

1. Differentiate between Artificial Intelligence (AI) and Machine Learning (ML) approaches to problem solving.
2. Briefly describe various applications of Artificial Intelligence in real-world scenarios.
3. Explain the concept of uninformed search algorithms in AI problem solving. Provide an example.
4. Briefly describe the A* search algorithm and its advantages over uninformed search algorithms.
5. What is knowledge representation in AI? Explain two common knowledge representation methods.
6. Briefly differentiate between forward and backward reasoning in AI.
7. Define an artificial neural network (ANN) and its basic structure.
8. Explain the concept of supervised learning in machine learning.
9. Briefly describe the K-means clustering algorithm for unsupervised machine learning.
10. What is the importance of feature engineering in machine learning projects?

Section B

[Answer All. Each question carries 6 marks] (Ceiling 36 Marks)

11. Compare and contrast depth-first search and breadth-first search algorithms used in AI problem solving. Analyze their time and space complexities.
12. Explain the concept of heuristic functions used in informed search algorithms. Provide an example of a heuristic function for a specific problem.
13. Briefly describe the process of knowledge representation using propositional logic with an example.
14. Explain the working principle of a single-layer perceptron model in artificial neural networks.
15. Briefly describe the concept of backpropagation used for training multi-layer perceptrons.
16. Differentiate between classification and regression algorithms in supervised machine learning. Provide an example of each.
17. Explain the concept of dimensionality reduction in machine learning and its benefits. Briefly describe Principal Component Analysis (PCA).
18. Describe the steps involved in building and evaluating a classification model using machine learning.

Section C

[Answer any one. Each question carries 10 marks] (1 x 1 = 10 Marks)

19. a) Implement a depth-first search algorithm in Python to find a path from a start node to a goal node in a maze represented as a 2D array.

    b) Briefly explain the concept of search space complexity in AI problem solving.

20. a) Explain the concept of a decision tree algorithm used for classification in supervised machine learning. How does a decision tree make predictions on new data?

    b) Write a Python code snippet to perform K-means clustering on a sample dataset.

    c) Briefly describe the concept of overfitting in machine learning and methods to prevent it.

***

SIXTH SEMESTER (CUFYUGP) DEGREE EXAMINATIONS, OCTOBER 2024

Computer Science

CSC8EJ402- System Software

(2024 Admissions)

Time: Two Hours **Maximum: 70 Marks**

Section A

[Answer All. Each question carries 3 marks] (Ceiling 24 Marks)

1. Define the term "System Programming" and explain its goals.
2. Explain the difference between compilers, assemblers, linkers, and loaders.
3. Describe the stages involved in the compilation process and the purpose of each stage.
4. What are system calls? Provide examples of different types of system calls.
5. Discuss the principles of lexical analysis in the context of compiler design.
6. Explain the role of macros and macro processors in system programming.
7. Define the concepts of relocation and linking. How are they related to each other?
8. Differentiate between absolute loaders and relocating loaders.
9. What are the key data structures used in compilers? Explain their significance.
10. Describe the phases involved in a compiler with a focus on code optimization

Section B

[Answer All. Each question carries 6 marks] (Ceiling 36 Marks)

11. Discuss the design and functionality of a two-pass assembler.
12. Explain the process of debugging in system programming. Highlight the use of debugging tools and techniques.
13. Analyze the impact of optimization techniques in the compilation process. Provide examples to support your analysis.
14. Evaluate the advantages and disadvantages of various linking and loading schemes.
15. Describe the principles of compiler design and their application in writing a simple compiler.
16. How do system calls facilitate various system-level tasks? Provide examples of system calls used in process management.
17. Discuss the standard C library functions commonly used for system calls. Provide examples to illustrate their usage.
18. Explain the concept of intermediate code generation in compiler design. How does it contribute to the overall compilation process?

Section C

[Answer any one. Each question carries 10 marks]　　　(1 x 1 = 10 Marks)

19. Design a simple macro and demonstrate its usage in a sample assembly code.
20. Develop a high-level algorithm for a basic compiler, outlining the key phases and their interactions. Provide a brief explanation of each phase

***

SEVENTH SEMESTER (CUFYUGP) DEGREE EXAMINATIONS, OCTOBER 2024

Computer Science

CSC7CJ401- Theory of Computation

(2024 Admissions)

Time: Two Hours **Maximum: 70 Marks**

Section A

[Answer All. Each question carries 3 marks] (Ceiling 24 Marks)

1. Define the term "formal language" in the context of Theory of Computation.
2. Briefly describe the Chomsky hierarchy for classifying formal languages.
3. Differentiate between Deterministic Finite Automata (DFA) and Non-deterministic Finite Automata (NFA).
4. Explain the concept of ε-moves in Non-deterministic Finite Automata.
5. Briefly describe the pumping lemma for regular languages and its application.
6. What is a Pushdown Automata (PDA)? Explain its basic components.
7. Differentiate between Deterministic Pushdown Automata (DPDA) and Non-deterministic Pushdown Automata (NPDA).
8. Briefly explain the concept of Context-Free Grammars (CFGs) used to define context-free languages.
9. What is the Church-Turing Thesis?
10. Define the terms "decidability" and "undecidability" in the context of computational problems.

Section B

[Answer All. Each question carries 6 marks] (Ceiling 36 Marks)

11. Construct a Deterministic Finite Automata (DFA) that recognizes binary strings ending with a 10.
12. Briefly explain the steps involved in converting a Regular Expression (RE) to a Non-deterministic Finite Automata (NFA).
13. Prove the closure of regular languages under union using a formal proof with set theory notation.
14. Explain the concept of an instantaneous description of a Pushdown Automata (PDA) during configuration transition.
15. Briefly describe the pumping lemma for context-free languages and its application.
16. Convert the following Context-Free Grammar (CFG) to Chomsky Normal Form (CNF):
    S -> AB | a A -> bS | b B -> ε
17. Briefly explain the CYK algorithm used to check membership of a string in a context-free language.

18. Explain the concept of the Halting Problem and its significance in understanding the limitations of computation.

Section C

[Answer any one. Each question carries 10 marks] (1 x 1 = 10 Marks)

19. a) Design a Turing Machine that recognizes strings over the alphabet {0, 1} containing an even number of 0s.

    b) Briefly explain the concept of the Universal Turing Machine and its significance in the Theory of Computation.

20. a) Explain Rice's Theorem and its implications for the undecidability of properties of programs.

    b) Briefly describe the classes P and NP in computational complexity theory. Explain the P vs. NP problem.

***

SEVENTH SEMESTER (CUFYUGP) DEGREE EXAMINATIONS, OCTOBER 2024

Computer Science

CSC7CJ402- System Security

(2024 Admissions)

Time: Two Hours **Maximum: 70 Marks**

Section A

[Answer All. Each question carries 3 marks] (Ceiling 24 Marks)

1. Define the CIA triad (Confidentiality, Integrity, Availability) in system security. Briefly explain the importance of each element.
2. Differentiate between threats and vulnerabilities in system security.
3. Briefly describe common types of system attacks and their impact.
4. Explain the role of attackers in system security and different types of attackers based on their motivations.
5. What is the importance of operating system security?
6. Briefly describe file protection mechanisms used in operating systems.
7. Define the term "database security" and its key objectives.
8. Explain the concept of risk analysis in system security planning.
9. Briefly describe different types of security policies that can be implemented in an organization.
10. What is the purpose of security controls in information security?

Section B

[Answer All. Each question carries 6 marks] (Ceiling 36 Marks)

11. Explain how buffer overflow vulnerabilities can be exploited by attackers to gain unauthorized access to a system.
12. Briefly describe different types of malicious code (viruses, worms, etc.) and how they spread within a system.
13. Explain control mechanisms used to prevent or mitigate program-based threats like buffer overflows.
14. Describe different memory protection techniques used in operating systems to safeguard data integrity.
15. Briefly explain the concept of access control lists (ACLs) used for file system security.
16. Explain the concept of authentication and different authentication mechanisms used in computer systems.
17. Briefly describe the concept of multilevel security in database management systems.
18. Explain the importance of data integrity and reliability in database security.

Section C

[Answer any one. Each question carries 10 marks]   (1 x 1 = 10 Marks)

19. a) Discuss the challenges faced in implementing strong password policies in an organization.

    b) Briefly describe different biometric authentication methods used for user identification.

20. a) Explain the concept of a Trusted Operating System (TOS) and its role in enhancing system security.

    b) Describe different types of security controls (preventive, detective, corrective) and their functionalities.

***

SEVENTH SEMESTER (CUFYUGP) DEGREE EXAMINATIONS, OCTOBER 2024

Computer Science

CSC7CJ403- Advanced Data Structures and algorithms

(2024 Admissions)

Time: Two Hours **Maximum: 70 Marks**

Section A

[Answer All. Each question carries 3 marks] (Ceiling 24 Marks)

1. Define the term "data structure" and explain its characteristics.
2. Briefly describe the concept of Abstract Data Types (ADTs) in computer science.
3. Explain the terms "time complexity" and "space complexity" used for analyzing algorithms.
4. Differentiate between Big O notation and Big Theta notation used for asymptotic analysis of algorithms.
5. Briefly describe the Brute-Force search algorithm and provide an example of its application.
6. Explain the concept of the Divide-and-Conquer strategy used for algorithm design.
7. What is a Greedy Algorithm? Briefly describe a real-world application of a greedy algorithm.
8. Define a Binary Search Tree (BST) and its basic operations.
9. Briefly explain the concept of a graph data structure and its basic operations.
10. What is a Heap data structure? Differentiate between Min-Heaps and Max-Heaps.

Section B

[Answer All. Each question carries 6 marks] (Ceiling 36 Marks)

11. Briefly describe the steps involved in the Merge Sort algorithm and analyze its time and space complexity.
12. Explain the concept of the Knapsack problem and how it can be solved using a Branch-and-Bound technique.
13. Describe Kruskal's algorithm for finding the minimum spanning tree of a graph.
14. Explain the concept of Dynamic Programming and its application to solve the Longest Common Subsequence (LCS) problem.
15. Briefly describe the concept of backtracking algorithms and provide an example of a problem that can be solved using backtracking.
16. Perform an in-order traversal on the following Binary Search Tree and write down the elements visited: A -> B -> C -> D -> E
17. Explain the concept of Breadth-First Search (BFS) traversal used for graphs and its applications.

18. Briefly describe the concept of Heapsort, a sorting algorithm based on heap data structures. Analyze its time complexity.

Section C

[Answer any one. Each question carries 10 marks] (1 x 1 = 10 Marks)

19. a) Implement a function in Python to perform a linear search on a singly linked list to find a specific element.

    b) Briefly explain the advantages and disadvantages of using linked lists compared to arrays as data structures.

20. a) Explain the concept of AVL trees and how they maintain balance in a binary search tree.

    b) Describe different graph traversal techniques (Depth-First Search (DFS)) and their applications.

***

SEVENTH SEMESTER (CUFYUGP) DEGREE EXAMINATIONS, OCTOBER 2024

Computer Science

CSC7CJ404- Blockchain Technology

(2024 Admissions)

Time: Two Hours **Maximum: 70 Marks**

Section A

[Answer All. Each question carries 3 marks] (Ceiling 24 Marks)

1. Briefly explain the concept of cryptography and its role in blockchain technology.
2. Differentiate between symmetric and asymmetric cryptography used in blockchain. Provide examples of each.
3. What is a digital signature? Explain its importance in securing transactions on a blockchain network.
4. Define the term "blockchain" and its basic architecture.
5. Briefly describe the benefits and limitations of blockchain technology.
6. Differentiate between public, private, and consortium blockchains.
7. Explain the concept of consensus mechanisms in blockchain networks.
8. Briefly describe the Proof-of-Work (PoW) consensus algorithm used in Bitcoin.
9. What are smart contracts on a blockchain? Provide an example of a use case for smart contracts.
10. Briefly explain the concept of Decentralized Autonomous Organizations (DAOs).

Section B

[Answer All. Each question carries 6 marks] (Ceiling 36 Marks)

11. Explain the working principle of Elliptic Curve Cryptography (ECC) used in blockchain for key generation and digital signatures.
12. Describe the concept of a Merkle tree and its applications in ensuring data integrity within a blockchain.
13. Briefly explain the concept of decentralization and how blockchain technology promotes decentralized networks.
14. Differentiate between Byzantine Fault Tolerance (BFT) and Crash Fault Tolerance (CFT) consensus algorithms used in blockchain.
15. Explain the concept of Proof-of-Stake (PoS) as a consensus mechanism and its advantages over Proof-of-Work (PoW).
16. Describe the basic structure of a Bitcoin transaction and its lifecycle within the blockchain network.
17. Explain the concept of cryptocurrency mining and the role of miners in securing the blockchain network.
18. Briefly describe different types of blockchain oracles and their functionalities within smart contracts.

Section C

[Answer any one. Each question carries 10 marks] (1 x 1 = 10 Marks)

19. a) Explain the potential applications of blockchain technology in the healthcare industry.

    b) Briefly discuss the challenges associated with implementing blockchain technology in real-world applications.

20. a) Write a simple Python program to demonstrate the generation of a SHA-256 hash for a given string.

    b) Explain how blockchain technology can be used to enhance supply chain management processes.

***

SEVENTH SEMESTER (CUFYUGP) DEGREE EXAMINATIONS, OCTOBER 2024

Computer Science

CSC7CJ405- Internet of Things

(2024 Admissions)

Time: Two Hours **Maximum: 70 Marks**

Section A

[Answer All. Each question carries 3 marks] (Ceiling 24 Marks)

1. Define the Internet of Things (IoT) and explain its key components.
2. Differentiate between M2M communication and IoT communication.
3. Briefly explain the concept of IoT architecture with a basic diagram.
4. List the key design principles for connected devices in an IoT system.
5. What is the role of a gateway in an IoT system?
6. Explain the importance of data enrichment in IoT applications.
7. Briefly describe commonly used communication protocols in IoT networks.
8. Explain the difference between HTTP and HTTPS protocols in the context of IoT.
9. Give examples of popular hardware platforms used for IoT development.
10. Briefly describe the process of implementing an IoT system for real-time data collection and visualization.

Section B

[Answer All. Each question carries 6 marks] (Ceiling 36 Marks)

11. Explain the various sources of data in an IoT system.
12. Discuss the role of sensors and actuators in IoT applications.
13. Explain the concept of Media Access Control (MAC) in IoT communication.
14. Briefly describe the working principles of REST and SOAP protocols.
15. Explain the importance of IPv6 addressing in modern IoT networks.
16. Discuss the benefits and limitations of using wireless sensor networks (WSN) in IoT.
17. Briefly explain the functionalities of popular cloud platforms for IoT development.
18. Discuss the security vulnerabilities and threats associated with IoT systems.

Section C

[Answer any one. Each question carries 10 marks] (1 x 1 = 10 Marks)

19. Choose a real-world application of IoT (e.g., smart home, wearables, industrial automation) and explain the system architecture, data flow, and potential benefits of the application.
20. Describe the design process for an IoT system. Your explanation should include:
    * Identifying the problem or need the system will address.
    * Selecting the hardware and software components.
    * Designing the communication protocols and data flow.
    * Implementing and testing the system.

***

SEVENTH SEMESTER (CUFYUGP) DEGREE EXAMINATIONS, OCTOBER 2024

Computer Science

CSC8CJ406- Compiler Design

(2024 Admissions)

Time: Two Hours **Maximum: 70 Marks**

Section A

[Answer All. Each question carries 3 marks] (Ceiling 24 Marks)

1. Briefly explain the different phases involved in the compilation process.
2. Define the term "lexical analysis" and explain its role in a compiler.
3. Differentiate between tokens, patterns, and lexemes.
4. Explain the concept of left recursion in context-free grammars (CFGs) and methods for its elimination.
5. What is a parse tree? Explain its significance in compiler design.
6. Briefly describe the top-down and bottom-up parsing approaches.
7. Differentiate between LL(1) and SLR(1) parsing techniques.
8. Explain the concept of syntax-directed translation (SDT) in the context of compilers.
9. Discuss the various storage allocation strategies used in compilers.
10. Briefly describe the three-address code intermediate representation.

Section B

[Answer All. Each question carries 6 marks] (Ceiling 36 Marks)

11. Explain the concept of bootstrapping in compiler writing.
12. Describe the process of input buffering in lexical analysis.
13. Construct a Non-Deterministic Finite Automata (NFA) to recognize identifiers in a programming language. (You can mention the concept briefly without going into detailed construction.
14. Given the following grammar, derive the string "a+b" using leftmost derivation: E -> TE' | T E' -> +T | ε T -> F F -> (E) | id
15. Construct a parse tree for the expression "a * b + c" using the appropriate parsing technique.
16. Explain the steps involved in the bottom-up parsing approach with an example.
17. Discuss the advantages and limitations of S-attributed and L-attributed definitions in SDT.
18. Explain the concept of local and global optimization in code generation.

Section C

[Answer any one. Each question carries 10 marks] (1 x 1 = 10 Marks)

19. Design a lexical analyzer for a simple calculator program that recognizes tokens like numbers, operators (+, -, *, /), and parentheses. Use a regular expression or Finite Automata (FA) to define the token patterns.
20. Consider the following grammar for a simple assignment statement language:

    S -> id = E ;

    E -> T | E + T | E - T

    T -> F | T * F | T / F

    F -> (E) | num

    a) Construct an SLR(1) parsing table for the above grammar.

    b) Use the parsing table to parse the statement "a = b * c + d".

***

EIGHTH SEMESTER (CUFYUGP) DEGREE EXAMINATIONS, OCTOBER 2024

Computer Science

CSC8CJ407- Client Server Architecture

(2024 Admissions)

Time: Two Hours **Maximum: 70 Marks**

Section A

[Answer All. Each question carries 3 marks] (Ceiling 24 Marks)

1. Define the term "client-server architecture" and explain its key components.
2. Briefly explain the factors that led to the development of client-server systems.
3. Describe the concept of a "single system image" in client-server computing.
4. Differentiate between two-tier and three-tier client-server architecture.
5. What is the role of middleware in client-server systems?
6. Explain the Model-View-Controller (MVC) design pattern used in client-server applications.
7. Briefly describe the types of client/server network services.
8. List some common management services offered in client-server systems.
9. Discuss the potential network issues in client-server communication.
10. Explain how remote desktop protocols (RDP) are used for client-server support.

Section B

[Answer All. Each question carries 6 marks] (Ceiling 36 Marks)

11. Explain the advantages and disadvantages of client-server computing compared to traditional centralized systems.
12. Describe the development process for a typical client-server application.
13. Discuss security considerations in client-server systems.
14. Briefly explain the concept of Remote Procedure Call (RPC) and its benefits.
15. Describe the functionalities of Windows services in a client-server environment.
16. Explain the working principles of Dynamic Data Exchange (DDE) between client and server applications.
17. Briefly discuss Object Linking and Embedding (OLE) technology in client-server communication.
18. Explain the concept of server administration and its key responsibilities.

Section C

[Answer any one. Each question carries 10 marks] (1 x 1 = 10 Marks)

19. Choose a real-world application (e.g., email system, online banking) and explain how it utilizes a client-server architecture. Discuss the roles of the client and server components, data flow, and benefits of this architecture for the chosen application

20. A company is planning to develop a new e-commerce platform.

    a) Discuss the factors to consider when choosing between a two-tier and three-tier client-server architecture for this application.

    b) Describe the potential network management challenges associated with the chosen architecture and propose solutions to address them.

***

EIGHTH SEMESTER (CUFYUGP) DEGREE EXAMINATIONS, OCTOBER 2024

Computer Science

CSC8CJ489- Research Methodology

(2024 Admissions)

Time: Two Hours **Maximum: 70 Marks**

Section A

[Answer All. Each question carries 3 marks] (Ceiling 24 Marks)

1. Define research methodology and explain its importance in conducting research.
2. Differentiate between basic research and applied research. Provide an example of each.
3. Explain the concept of research problem and its significance in the research process.
4. Briefly describe the different research approaches used in various disciplines.
5. What are the key features of a good research design?
6. Differentiate between primary and secondary data sources used in research.
7. Explain the concept of data validation and its importance in research.
8. Describe the different types of sampling techniques used in data collection.
9. Explain the concept of reliability in research and methods to ensure it.
10. Briefly describe the different measures of central tendency used in data analysis.

Section B

[Answer All. Each question carries 6 marks] (Ceiling 36 Marks)

11. Discuss the various factors influencing the selection of a research topic.
12. Explain the steps involved in formulating a clear and concise research question.
13. Briefly describe the types of research designs suitable for quantitative and qualitative research approaches.
14. Discuss the advantages and disadvantages of using questionnaires as a data collection method.
15. Explain the concept of validity in research and different types of validity tests.
16. Briefly describe the process of data processing and analysis in research.
17. Explain how measures of dispersion help in understanding data variability.
18. Discuss the importance of research report writing and its key components.

Section C

[Answer any one. Each question carries 10 marks] (1 x 1 = 10 Marks)

19. Choose a field of study that interests you (e.g., computer science, psychology, education) and propose a research topic within that field.
    i. Define the research problem and research question.
    ii. Justify the need for the research.
    iii. Explain the research design you would adopt (e.g., survey, experiment, case study).
    iv. Describe the methods you would use for data collection and analysis.
20. Imagine you are a researcher studying the effectiveness of online learning platforms. Explain how you would interpret the results of your research and present your findings in a clear and concise research report. Include the steps involved in data interpretation, key findings, limitations of the study, and recommendations for future research.

***

EIGHTH SEMESTER (CUFYUGP) DEGREE EXAMINATIONS, OCTOBER 2024

Computer Science

CSC8CJ408- Parallel Computing

(2024 Admissions)

Time: Two Hours **Maximum: 70 Marks**

Section A

[Answer All. Each question carries 3 marks] (Ceiling 24 Marks)

1. Define parallel computing and explain its key advantages over traditional sequential computing.
2. Briefly describe the different parallel computational models (e.g., Flynn's Taxonomy).
4. Explain the concept of task decomposition and its importance in parallel algorithms.
5. What are the challenges associated with communication overhead in parallel computing?
6. Differentiate between one-to-all broadcast and all-to-one reduction communication models.
7. Explain the concept of message passing paradigm in parallel programming.
8. Briefly describe the functionality of the Message Passing Interface (MPI) standard.
9. What are the benefits of using threads in parallel programming?
10. Explain the role of synchronization primitives in thread management.

Section B

[Answer All. Each question carries 6 marks] (Ceiling 36 Marks)

11. Discuss the factors to consider when designing parallel algorithms for problem solving.
12. Explain the concept of load balancing in parallel computing and techniques to achieve it.
13. Briefly describe the scatter and gather communication operations in parallel algorithms.
14. Explain the concept of overlapping communication with computation in message-passing programming.
15. Discuss the functionalities of groups and communicators in MPI.
16. Explain the basic operations involved in thread creation and management using a POSIX API.
17. Briefly describe different synchronization constructs used in thread programming (e.g., mutex, semaphores).
18. Explain how OpenMP directives can be used to parallelize a matrix multiplication algorithm.

Section C

[Answer any one. Each question carries 10 marks] (1 x 1 = 10 Marks)

19. Choose a real-world problem that can be solved using parallel computing (e.g., image processing, weather simulation).
    - Explain how you would decompose the problem into smaller tasks suitable for parallel processing.
    - Describe the communication model you would use (message passing or shared memory).
    - Justify your choice of communication model and discuss the potential challenges associated with it.
20. Consider a parallel algorithm for sorting a large list of numbers using a divide-and-conquer approach.
    - Explain the steps involved in the algorithm for parallel execution using message passing paradigm.
    - Illustrate the algorithm with an example (e.g., sorting a list of 8 numbers using 2 processors).

***

EIGHTH SEMESTER (CUFYUGP) DEGREE EXAMINATIONS, OCTOBER 2024

Computer Science

CSC8EJ404- Advanced Distributed Computing

(2024 Admissions)

Time: Two Hours **Maximum: 70 Marks**

Section A

[Answer All. Each question carries 3 marks] (Ceiling 24 Marks)

1. Define distributed computation model and highlight its key aspects.
2. Discuss the goals of distributed systems and differentiate between synchronous and asynchronous execution.
3. Explain the relationship between distributed systems and parallel systems.
4. Describe the different types of distributed system models.
5. What are the hardware and software concepts related to distributed systems? Provide examples.
6. Briefly explain the role of middleware models in distributed systems.
7. Define communication and coordination in the context of distributed computing. Discuss the models of process communication.
8. Explain the concepts of consistency and atomicity in distributed systems.
9. Discuss the significance of scalability and cache coherence in multiprocessor systems.
10. What are the synchronization mechanisms used in distributed systems? Provide examples.

Section B

[Answer All. Each question carries 6 marks] (Ceiling 36 Marks)

11. Illustrate the design principles of distributed computing and communication.
12. Compare and contrast the models of communication networks in distributed systems.
13. Explain the concepts of shared memory and message-passing in distributed systems. How do they contribute to coordination?
14. Discuss the role of consensus in distributed systems. How is it achieved?
15. Describe the Bully algorithm and the Ring algorithm for leader election in distributed systems.
16. Explain the Chandy-Lamport snapshot algorithm for FIFO channels. How does it ensure consistent global states?
17. Discuss termination detection in distributed systems using distributed snapshots. How is it different from termination detection by weight throwing?
18. Analyze the scalability challenges in distributed systems and propose strategies to address them.

Section C

[Answer any one. Each question carries 10 marks] (1 x 1 = 10 Marks)

19. Choose a mutual exclusion algorithm (e.g., Maekawa's algorithm, Suzuki-Kasami's broadcast algorithm) and explain its working principle and advantages in distributed systems.
20. Discuss the issues associated with deadlock detection in distributed systems. How can deadlock handling strategies mitigate these issues?

***

EIGHTH SEMESTER (CUFYUGP) DEGREE EXAMINATIONS, OCTOBER 2024

Computer Science

CSC8EJ403- Social Networks Analysis

(2024 Admissions)

Time: Two Hours **Maximum: 70 Marks**

Section A

[Answer All. Each question carries 3 marks] (Ceiling 24 Marks)

1. Define the following terms in the context of social network analysis:
   a. Node
   b. Edge
   c. Degree
2. Explain the significance of social network structure in understanding social dynamics.
3. What are the challenges associated with analyzing social network streams?
4. Briefly discuss the concept of random walks on graphs and its relevance in social network analysis.
5. How do proximity measures contribute to link prediction in social networks?
6. Define community structure in the context of social networks. What are some statistical techniques used to discover communities?
7. Discuss the importance of incorporating content information in community discovery for heterogeneous networks.
8. Describe the Katz score and its application in link prediction.
9. Differentiate between directed and undirected networks. Provide examples of each.
10. How does SimRank contribute to understanding network similarity? Provide a brief explanation.

Section B

[Answer All. Each question carries 6 marks] (Ceiling 36 Marks)

11. Explain the process of evaluating datasets in social network analysis. What are some common evaluation metrics used?
12. Discuss the role of graph theoretic measures in semi-supervised learning.
13. Describe the Markov Clustering algorithm and its application in community detection.
14. How can random walk-based measures be utilized for clustering in social networks?
15. Compare and contrast feature-based link prediction with Bayesian probabilistic models.

16. Discuss the application of link prediction techniques in discovering new connections in social networks.
17. Explain the concept of social influence analysis. How can it be applied in recommendation systems?
18. Analyze the challenges associated with community discovery in dynamic networks. Provide potential solutions.

Section C

[Answer any one. Each question carries 10 marks] (1 x 1 = 10 Marks)

19. Choose a specific online social network platform (e.g., Facebook, Twitter) and describe how social network analysis techniques can be applied to understand its structure, connectivity patterns, and influence dynamics.
20. Select a real-world scenario (e.g., marketing campaign, information propagation) and propose a comprehensive approach using social network analysis techniques to optimize outcomes and enhance understanding

***

Model Question Papers/ Electives

FIFTH SEMESTER (ELECTIVE) (CUFYUGP) DEGREE EXAMINATIONS, OCTOBER 2024

Computer Science

CSC5EJ305a- Mathematical and Statistical Foundations for Data Science

(2024 Admissions)

Time: Two Hours **Maximum: 70 Marks**

Section A

[Answer All. Each question carries 3 marks] (Ceiling 24 Marks)

1. Explain the concept of vector addition with an example.
2. Define linear dependence and provide an example of linearly dependent and independent vectors.
3. What is the difference between a diagonal and an orthogonal matrix?
4. What is the significance of eigenvalues and eigenvectors in data analysis?
5. Define conditional probability and explain Bayes' rule with an example.
6. Differentiate between mean, median, and mode and mention their applications in data analysis.
7. Explain the concept of variance and how it measures the spread of data around the mean.
8. What is the Central Limit Theorem and what does it imply about sampling distributions?
9. Explain the difference between positive and negative correlation.
10. Briefly explain the concept of linear regression and its use in data analysis.

Section B

[Answer All. Each question carries 6 marks] (Ceiling 36 Marks)

11. Given vectors A = [1, 2] and B = [3, 4], calculate the dot product and explain its geometric interpretation.
12. Find the inverse of the matrix [[2, 1], [1, 3]] (if it exists) using any method.
13. A coin is tossed three times. Calculate the probability of getting at least two heads using the concept of conditional probability.
14. A dataset contains exam scores of students. Explain how standard deviation helps in analyzing the spread of scores.
15. Differentiate between one-tailed and two-tailed hypothesis tests and provide an example for each.
16. A sample of 20 students has an average height of 165 cm with a standard deviation of 5 cm. Construct a 95% confidence interval for the population mean height.
17. Explain the concept of sampling distribution and its importance in hypothesis testing.

18. You are analyzing customer purchase data for an online store. Explain how Principal Component Analysis (PCA) can help in reducing data dimensionality.

Section C

[Answer any one. Each question carries 10 marks] (1 x 1- = 10 Marks)

19. A company collects data on customer age and purchase amount. You are required to perform a linear regression analysis using a spreadsheet to understand the relationship between age and purchase amount. Describe the steps involved in the analysis and how you would interpret the results.

20. A research team is studying the effect of different fertilizers on plant growth. Explain how they can use ANOVA to compare the mean growth rates of plants under different fertilizer conditions using a spreadsheet. .

***

FIFTH SEMESTER (ELECTIVE) (CUFYUGP) DEGREE EXAMINATIONS, OCTOBER 2024

Computer Science

CSC5EJ306a- Exploratory Data Analysis

(2024 Admissions)

Time: Two Hours **Maximum: 70 Marks**

Section A

[Answer All. Each question carries 3 marks] (Ceiling 24 Marks)

1. Explain the importance of data visualization for business intelligence and decision making.
2. Differentiate between nominal and ordinal data attributes. Give an example for each.
3. Briefly describe the following types of charts and plots: a. Line Chart b. Bar Chart c. Pie Chart
4. Which visualization technique would be most suitable for representing the relationship between student age and exam scores? Justify your answer.
5. What category of visualization does a histogram belong to? Briefly explain its application.
6. State two advantages of using scatter plots for data analysis.
7. Differentiate between continuous and discrete data. Provide an example for each.
8. Briefly explain the concept of data visualization libraries in Python. List any two popular libraries.
9. Match the following visualizations with their applications: (a) Bar Chart (i) Showing parts of a whole (b) Histogram (ii) Comparing categories across different variables (c) Scatter Plot (iii) Visualizing distribution of data (d) Pie Chart (iv) Identifying relationships between two variables.
10. What is the difference between primary and secondary data?

Section B

[Answer All. Each question carries 6 marks] (Ceiling 36 Marks)

11. You are given a dataset containing sales figures for different product categories across various regions. Describe the steps involved in creating a stacked bar chart to compare sales across categories for each region.
12. Explain how you would use a box plot to visualize the distribution of customer ages in a dataset.
13. Briefly describe the functionalities of Matplotlib and Seaborn libraries in Python for data visualization.
14. Consider a dataset with information on customer location (city, state) and their purchase history. How would you create a choropleth map to visualize spending patterns across different states?
15. Using an example, explain how colour palettes can be used to enhance the effectiveness of data visualizations.
16. Imagine you have data on stock prices for different companies over time. How would you create a line chart with annotations to highlight significant events that might have affected the stock prices?
17. Briefly discuss the concept of storytelling with data visualization. Why is it important?

18. Explain how you would use a scatter plot with a trendline to analyze the relationship between advertising expenditure and product sales.

Section C

[Answer any one. Each question carries 10 marks] (1 x 1- = 10 Marks)

19. A marketing manager has collected data on customer demographics (age, gender, location) and their purchase behavior for a new product launch. Explain how data visualization techniques can be used to analyze this data and gain valuable insights to improve the marketing strategy. Discuss the specific types of visualizations you would recommend and how they would be helpful.

20. A social media platform wants to understand user engagement patterns across different age groups. They have collected data on user demographics (age, location) and their activity levels (likes, comments, shares). Explain how data visualization techniques can be used to analyze this data and gain insights to improve user engagement strategies. Discuss the specific types of visualizations you would recommend and how they would be helpful.

***

SIXTH SEMESTER (ELECTIVE) (CUFYUGP) DEGREE EXAMINATIONS, OCTOBER 2024

Computer Science

CSC6EJ311a- Introduction to Data Warehousing and Big Data

(2024 Admissions)

Time: Two Hours **Maximum: 70 Marks**

Section A

[Answer All. Each question carries 3 marks] (Ceiling 24 Marks)

1. Define data warehousing and explain its importance in business intelligence.
2. Differentiate between data normalization in databases and data warehousing.
3. Briefly describe the three-tier architecture of a data warehouse.
4. Explain the difference between data warehouses and data marts.
5. What are the three Vs of Big Data? Briefly explain each.
6. State the role of cloud computing and distributed processing in Big Data management.
7. What are the two main types of data processed in Big Data? Give an example of each.
8. Briefly explain the concept of Online Analytical Processing (OLAP).
9. Differentiate between OLAP and OLTP operations. Provide an example for each.
10. What are the benefits of using dimensional modeling in data warehousing?

Section B

[Answer All. Each question carries 6 marks] (Ceiling 36 Marks)

11. Explain the concept of star schema design in data warehousing. Briefly discuss its advantages and disadvantages.
12. Describe the concept of drill-down and roll-up operations in OLAP.
13. Briefly explain the functionalities of MapReduce jobs in Big Data processing.
14. Imagine a data warehouse stores sales data for a retail chain. Explain the ETL (Extract, Transform, Load) process involved in populating the data warehouse.
15. Describe the components of the Hadoop ecosystem, including HDFS, YARN, and MapReduce.
16. Briefly discuss the functionalities of Apache Pig and its advantages for processing big data.
17. Explain the role of ZooKeeper in the Hadoop ecosystem.
18. Differentiate between ROLAP and MOLAP implementations of OLAP models.

Section C

[Answer any one. Each question carries 10 marks] (1 x 1- = 10 Marks)

19. A telecommunications company wants to analyze customer call data to identify usage patterns and improve network performance. The data includes call duration, location, time of day, and customer demographics. Explain how data warehousing and big data processing can be used to achieve this objective. Discuss the relevant data warehouse architecture, ETL processes, and Big Data tools that can be employed for this purpose.

20. A hospital wants to analyze patient data to identify trends in disease prevalence, patient demographics, and treatment outcomes. The data includes patient records, diagnosis codes, and treatment details. Explain how a combination of data warehousing and Big Data processing can be used to achieve this objective. Discuss the relevant data warehouse architecture, ETL processes, and Big Data tools that can be employed for this purpose.

***

SIXTH SEMESTER (ELECTIVE) (CUFYUGP) DEGREE EXAMINATIONS, OCTOBER 2024

Computer Science

CSC6CJ312a - Advanced Python for Data Science

(2024 Admissions)

Time: Two Hours **Maximum: 70 Marks**

Section A

[Answer All. Each question carries 3 marks] (Ceiling 24 Marks)

1. Explain the difference between an array and a matrix in NumPy.
2. How can you perform basic arithmetic operations on NumPy arrays? Briefly explain with an example.
3. What is broadcasting in NumPy? Give an example to illustrate this concept.
4. Describe two methods for creating a Pandas Series.
5. How can you access specific data points within a Pandas DataFrame using indexing?
6. Briefly explain the concept of data cleansing in Python.
7. Differentiate between CSV and JSON data formats.
8. State two methods for importing CSV data into a Pandas DataFrame.
9. What are the functions used to calculate mean, median, and standard deviation in Python for data analysis?
10. Explain the concept of correlation in data analysis.

Section B

[Answer All. Each question carries 6 marks] (Ceiling 36 Marks)

11. Consider a NumPy array containing student exam scores. Write Python code to:
    - Sort the scores in ascending order.
    - Find the number of students who scored above the average.
    - Create a histogram to visualize the distribution of exam scores.
12. Imagine you have a Pandas DataFrame containing product information (product ID, name, price). Write Python code to:
    - Calculate the average price of each product category.
    - Add a new column to the DataFrame indicating a discount percentage for each product.
    - Use pivot table functionality to analyze sales data by product category and month.
13. Briefly explain the functionalities of the following Python libraries for data science:
    - Scikit-learn

- Seaborn
- Beautiful Soup

14. Explain the steps involved in processing and cleaning a messy CSV dataset containing missing values and inconsistencies.
15. Consider a dataset with information on customer purchases. Write Python code to calculate the correlation coefficient between purchase amount and customer age.
16. Briefly describe the concept of random tensors and their creation in TensorFlow.
17. Explain the following tensor operations in TensorFlow:
    - Size and rank of a tensor
    - Reshaping a tensor
18. Write Python code using TensorFlow to create a tensor filled with random values drawn from a normal distribution with a specific mean and standard deviation.

Section C

[Answer any one. Each question carries 10 marks] (1 x 1- = 10 Marks)

19. A social media company wants to analyze user engagement data to understand user behavior and preferences. The data includes user demographics (age, location), platform usage details (likes, comments, shares), and post information (category, topic). Explain how you would use Python libraries like Pandas, NumPy, and TensorFlow to achieve this objective. Discuss the specific functionalities you would employ for data manipulation, exploratory data analysis, and potential machine learning applications.

20. An e-commerce company wants to build a recommendation system to suggest relevant products to customers based on their purchase history. Describe how Python libraries like Pandas and TensorFlow can be used to prepare the data, build a recommendation model, and evaluate its performance. Discuss the relevant data processing steps, model selection techniques, and evaluation metrics that would be used in this scenario.

***

FIFTH SEMESTER (ELECTIVE) (CUFYUGP) DEGREE EXAMINATIONS, OCTOBER 2024

Computer Science

CSC5EJ305b- Machine Learning Algorithms

(2024 Admissions)

Time: Two Hours **Maximum: 70 Marks**

Section A

[Answer All. Each question carries 3 marks] (Ceiling 24 Marks)

1. Briefly define supervised learning and provide an example of a supervised learning task.
2. Differentiate between unsupervised learning and reinforcement learning.
3. Explain the concept of features and labels in machine learning datasets.
4. What is the role of optimization algorithms in machine learning?
5. Briefly explain the concept of a probability distribution and its importance in machine learning.
6. Describe the concept of vectors and their operations (addition, subtraction) relevant to machine learning.
7. What is the purpose of data preprocessing in machine learning projects?
8. Briefly explain the difference between feature selection and feature extraction techniques.
9. State two common metrics used to evaluate the performance of a regression model.
10. Explain the difference between precision and recall in evaluating a classification model.

Section B

[Answer All. Each question carries 6 marks] (Ceiling 36 Marks)

11. Consider a dataset containing customer data and their purchase history. Describe the steps involved in data preprocessing for building a machine learning model to predict customer churn (stay vs. leave) using logistic regression.
12. Explain the K-Nearest Neighbors (KNN) algorithm and its advantages and disadvantages for classification tasks in Python using Scikit-learn.
13. Briefly describe the decision tree learning algorithm and its suitability for handling complex decision boundaries.
14. Imagine you have a dataset with patient medical records. Explain how feature engineering techniques can be used to improve the performance of a machine learning model for disease prediction.
15. Write Python code using Scikit-learn to perform K-Means clustering on a dataset containing product features. Explain what insights you can gain from the clustering results.

16. Briefly describe the Support Vector Machine (SVM) algorithm and its application for specific classification problems with high dimensionality.
17. Explain the concept of cross-validation in evaluating machine learning models. Discuss different cross-validation techniques.
18. Consider a binary classification problem where you are predicting fraudulent transactions. Explain how you would use a ROC curve to analyze the performance of your machine learning model.

Section C

[Answer any one. Each question carries 10 marks] (1 x 1- = 10 Marks)

19. A company wants to develop a machine learning model to recommend movies to users based on their past viewing history and ratings. Discuss the various steps involved in building such a recommendation system. Explain the relevant machine learning algorithms you would consider, feature engineering techniques, and evaluation metrics for this task.

20. An online retailer wants to predict customer demand for different products throughout the year. Describe a machine learning approach to solve this problem. Discuss the data sources, model selection process, and challenges you might encounter in building a successful forecasting model.

***

FIFTH SEMESTER (ELECTIVE) (CUFYUGP) DEGREE EXAMINATIONS, OCTOBER 2024

Computer Science

CSC5EJ306b- Knowledge

Engineering

(2024 Admissions)

Time: Two Hours **Maximum: 70 Marks**

Section A

1. Define the terms "evidence," "data," and "information." Explain the relationship between them.
2. Differentiate between abductive reasoning and probabilistic reasoning.
3. Briefly describe the Subjective Bayesian View of Probability.
4. What are intelligent agents, and how does mixed-initiative reasoning function?
5. What is an ontology in the context of knowledge engineering?
6. Explain the concept of a problem-solving task ontology.
7. Briefly discuss the development tools used in a conventional design and development scenario.
8. Define the terms "concept" and "instance" in relation to ontologies.
9. What is meant by "transitivity" in an ontology?
10. Briefly describe the steps involved in the ontology development methodology.

Section B

[Answer All. Each question carries 6 marks] (Ceiling 36 Marks)

11. Explain how evidence-based reasoning can be applied in real-world scenarios with an example.
12. Discuss the challenges faced in agent design and development using learning technology.
13. Describe the process of inquiry-driven analysis and synthesis for evidence-based reasoning.
14. Illustrate the concept of inheritance in an ontology with a suitable example.
15. Explain the process of ontology matching with its significance in knowledge engineering.
16. Discuss the advantages and disadvantages of using a production system architecture for knowledge representation.
17. Explain how reduction and synthesis rules are used by an inference engine. Provide an example.
18. Discuss the challenges associated with reasoning with partially learned knowledge in knowledge-based systems.

Section C

[Answer any one. Each question carries 10 marks] (1 x 1- = 10 Marks)

19. Design an ontology for a specific domain (e.g., library management system, medical diagnosis system) clearly outlining the concepts, relationships, and attributes involved. Explain how reasoning with this ontology could be beneficial for your chosen domain.
20. Consider a real-world problem and propose a knowledge-based system solution. Explain how you would utilize knowledge engineering principles (e.g., ontologies, reasoning) to design and develop the system. Discuss the potential benefits and limitations of this approach.

***

FIFTH SEMESTER (ELECTIVE) (CUFYUGP) DEGREE EXAMINATIONS, OCTOBER 2024

Computer Science

CSC6CJ311b

Soft Computing (2024

Admissions)

Time: Two Hours **Maximum: 70 Marks**

Section A

1. Define soft computing and discuss its historical factors influencing development.
2. Differentiate between soft computing, hard computing, and hybrid computing.
3. Explain the basic tools of soft computing and their applications.
4. Define fuzzy logic and its role in soft computing.
5. Compare and contrast fuzzy sets with crisp sets.
6. Discuss the properties of fuzzy relations and crisp relations.
7. Define tolerance and equivalence relations in the context of soft computing.
8. Explain the concept of fuzzy membership functions and their significance.
9. Describe the process of fuzzification and defuzzification.
10. Analyze the importance of soft computing techniques in solving real-world problems.

Section B

[Answer All. Each question carries 6 marks] (Ceiling 36 Marks)

11. Discuss the role of fuzzy logic concepts in real-world problem-solving. Provide examples.
12. Compare and contrast Mamdani and Sugeno models of fuzzy inference systems.
13. Explain the design and implementation of fuzzy control systems.
14. Describe the concepts of fuzzy clustering and fuzzy neural networks. How are they applied in practice?
15. Discuss the applications of genetic algorithms in solving optimization problems.
16. Explain the operators in genetic algorithms, including coding, selection, crossover, and mutation.
17. Describe the constraints encountered in genetic algorithms and strategies to handle them.
18. Discuss the classification of genetic algorithms and their applications in different domains.

Section C

[Answer any one. Each question carries 10 marks] (1 x 1- = 10 Marks)

19. Choose a real-world scenario (e.g., traffic management, medical diagnosis) and propose a soft computing-based solution integrating fuzzy logic, neural networks, and genetic algorithms. Discuss the design considerations and potential benefits.
20. Evaluate and present a case study where soft computing techniques have been effectively applied. Discuss the problem statement, solution approach, and outcomes achieved.

SIXTH SEMESTER (ELECTIVE) (CUFYUGP) DEGREE EXAMINATIONS, OCTOBER 2024

Computer Science

CSC6CJ312 Deep Learning

(2024 Admissions)

Time: Two Hours **Maximum: 70 Marks**

Section A

1. Define the key differences between supervised and unsupervised learning.
2. Explain the concept of overfitting and underfitting in machine learning. How can you address these issues?
3. Briefly describe the role of hyperparameters in a deep learning model.
4. Differentiate between bias and variance in the context of machine learning models.
5. What is a perceptron, and how does it relate to artificial neural networks?
6. Explain the role of activation functions in neural networks. Provide examples of common activation functions.
7. Briefly describe the concept of gradient descent and its significance in training neural networks.
8. What is the purpose of regularization techniques in deep learning? Give two examples.
9. Define convolutional operations in the context of Convolutional Neural Networks (CNNs).
10. Briefly explain the concept of recurrent neural networks (RNNs).

Section B

[Answer All. Each question carries 6 marks] (Ceiling 36 Marks)

11. Explain the backpropagation algorithm for training a multi-layer perceptron. Demonstrate how to calculate gradients using the chain rule.
12. Discuss the advantages and disadvantages of the Sigmoid and ReLU activation functions. When might you choose one over the other?
13. Describe the architecture of a simple feedforward neural network for image classification. Explain how data would flow through the network.
14. Explain the concept of pooling layers and their benefits in CNNs. Provide examples of pooling operations.
15. Briefly describe two different regularization techniques used to improve the generalization of deep learning models. Explain how they work.
16. Compare and contrast the architectures of AlexNet and VGG16 for image recognition.

17. Explain how Long Short-Term Memory (LSTM) networks address the vanishing gradient problem in RNNs.
18. Discuss the concept of transfer learning in deep learning and its potential benefits.

Section C

[Answer any one. Each question carries 10 marks] (1 x 1- = 10 Marks)

19. Design and implement a deep learning model for a specific real-world application (e.g., handwriting recognition, sentiment analysis, stock price prediction) using a suitable framework like TensorFlow or PyTorch. Explain your choice of architecture, hyperparameters, and training strategy.
20. A dataset containing medical images needs to be analyzed for disease detection. Discuss your approach to building a deep learning solution for this task. Explain the specific type of deep learning model you would choose, the pre-processing steps involved, and the evaluation metrics you would use to assess the model's performance.

***

FIFTH SEMESTER (ELECTIVE) (CUFYUGP) DEGREE EXAMINATIONS, OCTOBER 2024

Computer Science

CSC5EJ306- Cloud Computing

(2024 Admissions)

Time: Two Hours **Maximum: 70 Marks**

Section A

1. Briefly define cloud computing and its historical context.
2. List and explain the key features of a desirable cloud computing environment.
3. What are the main advantages of adopting cloud computing solutions?
4. Differentiate between a cloud reference model and a cloud service model.
5. Explain the concept of a public cloud and a private cloud.
6. Briefly describe the characteristics of a hybrid cloud environment.
7. What is Infrastructure as a Service (IaaS), and what services does it typically offer?
8. Define Platform as a Service (PaaS) and provide examples of its applications.
9. Explain the concept of Software as a Service (SaaS) and its benefits for users.
10. Briefly discuss the open challenges associated with cloud computing adoption.

Section B

[Answer All. Each question carries 6 marks] (Ceiling 36 Marks)

11. Explain the role of virtualization in cloud computing and its core principles.
12. Describe the different levels of virtualization implementation (e.g., full, paravirtualization).
13. Compare and contrast the functionalities of a hypervisor and a container.
14. Discuss the advantages and disadvantages of using virtual machines in cloud environments.
15. Explain the concept of desktop virtualization and its potential benefits for organizations.
16. Briefly describe the concept of network virtualization and its role in cloud security.
17. Explain the core functionalities of Docker containers and their advantages over virtual machines.
18. Discuss the key components of Docker, including images and repositories.

Section C

[Answer any one. Each question carries 10 marks]   (1 x 1- = 10 Marks)

19. An organization is considering migrating its IT infrastructure to the cloud. Analyze the different cloud service models (IaaS, PaaS, SaaS) and recommend the most suitable option for the organization based on their specific needs. Justify your recommendation.
20. Discuss the benefits and limitations of adopting a hybrid cloud model for a company. Provide a real-world example of how a hybrid cloud can be implemented for a specific business case (e.g., e-commerce, healthcare). Explore the cloud platforms available from leading industry players (e.g., Amazon Web Services, Microsoft Azure, Google Cloud Platform) and discuss the factors to consider when choosing a cloud provider.

***

FIFTH SEMESTER (ELECTIVE) (CUFYUGP) DEGREE EXAMINATIONS, OCTOBER 2024

Computer Science

CSC5EJ305- Security and Privacy in Cloud

(2024 Admissions)

Time: Two Hours **Maximum: 70 Marks**

Section A

1. Define the CIA triad (Confidentiality, Integrity, and Availability) in the context of cloud security.
2. Briefly explain the concept of encryption and decryption. Provide an example of each.
3. What are the key principles of secure system design?
4. Differentiate between conventional cryptography and public key cryptography.
5. Explain the role of hash functions in cloud security.
6. Describe the concept of digital signatures and their role in cloud security.
7. Briefly discuss the concept of cloud bursting and its security implications.
8. What is geo-tagging, and how can it be used to enhance security in the cloud?
9. Explain the importance of secure cloud interfaces for maintaining cloud security.
10. Briefly describe the concept of cloud resource access control.

Section B

[Answer All. Each question carries 6 marks] (Ceiling 36 Marks)

11. Discuss the various security threats and vulnerabilities specific to cloud computing environments.
12. Explain the concept of secure isolation strategies in cloud security. Provide examples of such strategies.
13. How do data retention, detection, and archiving procedures contribute to cloud security?
14. Describe the concept of Role-Based Access Control (RBAC) and its benefits for access control in the cloud.
15. Explain the concept of multi-factor authentication and its role in enhancing cloud security.
16. Briefly describe the functionalities of Identity Providers (IdPs) and Service Consumers in access control.
17. Discuss the importance of OS hardening and minimization for cloud security.
18. Explain how intrusion detection and prevention systems (IDS/IPS) contribute to cloud security.

Section C

[Answer any one. Each question carries 10 marks] (1 x 1- = 10 Marks)

19. A company is planning to migrate its sensitive financial data to the cloud. Analyze the potential security risks associated with this move. Discuss various cloud security design patterns (e.g., secure isolation, data encryption) that can be implemented to mitigate these risks. You can use spreadsheets to visualize the data and access control mechanisms.

20. A cloud-based e-commerce platform needs to implement robust access control mechanisms for its customers and administrators. Discuss how various access control options (e.g., RBAC, multi-factor authentication) can be combined to achieve a secure and user-friendly access control system. Explore additional security measures (e.g., data encryption, intrusion detection) that can be implemented to protect user data and system integrity within the cloud infrastructure. You can use spreadsheets to illustrate the access control framework.

***

FIFTH SEMESTER (ELECTIVE) (CUFYUGP) DEGREE EXAMINATIONS, OCTOBER 2024

Computer Science

CSC5EJ306- Storage Technologies

(2024 Admissions)

Time: Two Hours **Maximum: 70 Marks**

Section A

1. Explain the evolution of storage architecture, highlighting its major milestones.
2. Define Direct-Attached Storage (DAS) and its key characteristics.
3. Differentiate between Network-Attached Storage (NAS) and Storage Area Network (SAN).
4. What are the characteristics of Cloud storage? Explain.
5. Discuss the features and benefits of RAID technology.
6. How does RAID impact disk performance? Explain with examples.
7. Define Storage Provisioning. What are its types?
8. Describe the components of an Intelligent Storage System.
9. Explain the concept of block-based storage system.
10. What are the different deployment models of Cloud services? Briefly explain each.

Section B

[Answer All. Each question carries 6 marks] (Ceiling 36 Marks)

11. Discuss the role of Virtualization in Storage Area Networks (SAN).
12. Explain the components and architecture of a Fibre Channel Storage Area Network (FC SAN).
13. Describe the backup purpose and considerations in detail.
14. Discuss the various backup methods and their suitability for different scenarios.
15. How does backup granularity affect recovery operations? Explain with examples.
16. Analyze the importance of backup and recovery mechanisms in the context of storage technologies.
17. Discuss the impact of Cloud storage architectures on modern data center infrastructure.
18. Compare and contrast the features of different RAID levels.

Section C

[Answer any one. Each question carries 10 marks] (1 x 1- = 10 Marks)

19. a) Evaluate the need for backup and recovery mechanisms in storage technologies. Provide examples to support your answer.
20. (b) Discuss the security needs and management requirements for storage technologies. How can these be effectively addressed in modern storage architectures?

***

SIXTH SEMESTER (ELECTIVE) (CUFYUGP) DEGREE EXAMINATIONS, OCTOBER 2024

Computer Science

CSC6CJ312- Virtualization

(2024 Admissions)

Time: Two Hours **Maximum: 70 Marks**

Section A

1. Define virtualization and explain its significance in modern computing.
2. Discuss the need for virtualization, highlighting its advantages and limitations.
3. What are the different types of hardware virtualization? Explain each with examples.
4. Differentiate between full virtualization, partial virtualization, and paravirtualization.
5. Describe the role of hypervisors in virtualization. How do they work?
6. List and explain the types of hypervisors commonly used in virtualization environments.
7. What are virtual machines? Explain the different types of virtual machines.
8. Discuss the business cases for server virtualization and its benefits.
9. Explain the concept of desktop virtualization. What are its types?
10. How does virtual server consolidation contribute to infrastructure optimization?

Section B

[Answer All. Each question carries 6 marks] (Ceiling 36 Marks)

11. Discuss the advantages and functions of network virtualization.
12. Explain VLAN-WAN architecture and its significance in network virtualization.
13. Analyze the risks associated with storage virtualization and methods to mitigate them.
14. Compare and contrast SAN, NAS, and RAID in the context of storage virtualization.
15. Describe the different types of server virtualization platforms available in the market.
16. Compare server virtualization and desktop virtualization, highlighting their differences and use cases.
17. Discuss the tools available for network virtualization and their functionalities.
18. How does memory virtualization contribute to resource optimization in virtualized environments?

Section C

[Answer any one. Each question carries 10 marks] (1 x 1- = 10 Marks)

19. Explain how virtualization technologies are utilized in the context of cloud services. Discuss the benefits and challenges.
20. Analyze the potential risks and vulnerabilities associated with virtualization. Propose strategies to mitigate these risks effectively.

***

Computer Science

CSC8EJ401- Microprocessor and its Applications

(2024 Admissions)

Time: Two Hours **Maximum: 70 Marks**

Section A

1. Define the purpose and characteristics of the 8085 microprocessor.
2. Discuss the architecture of the 8086 microprocessor.
3. Explain the differences between microprocessors and microcontrollers.
4. Describe the evolution of microprocessors from earlier generations to modern ones.
5. Differentiate between high-level, machine, and assembly languages.
6. Draw and label the pin diagram of the 8085 microprocessor.
7. Identify the addressing modes of the 8086 microprocessor.
8. How do data transfer and arithmetic instructions work in the 8086 instruction set?
9. Explain the significance of branch and loop instructions in 8086 programming.
10. What are assembler directives? Provide examples.

Section B

[Answer All. Each question carries 6 marks] (Ceiling 36 Marks)

11. Illustrate simple assembly language programs for the 8086 microprocessor.
12. Discuss the role of interrupts and interrupt service routines in the 8086 microprocessor.
13. Explain the functions of peripheral integrated circuits (ICs) in the context of microprocessor systems.
14. How are procedures and macros used in assembly language programming?
15. Analyze the features and characteristics of the Intel 80186 and 80286 microprocessors.
16. Compare the features of the Intel 80386 and 80486 microprocessors.
17. Discuss the advancements introduced in Pentium processors compared to their predecessors.
18. Explain the features and advantages of multi-core processors in modern computing.

Section C

[Answer any one. Each question carries 10 marks] (1 x 1- = 10 Marks)

19. Describe the features of the i series processors from Intel. How do they differ from previous generations?

20. Discuss the characteristics and applications of mobile processors. How do they address the unique requirements of mobile devices compared to traditional desktop processors?