Faculty of Engineering and Technology (Co-Ed)

Department of Computer Science and Engineering

B. Tech 3rd year (V and VI Semester)

Scheme of Teaching and Examination

Outcome Based Education (OBE) & Choice Based Credit System (CBCS)

(Effective from the academic year 2023-2024)

VISION OF FACULTY OF ENGINEERING AND TECHNOLOGY(Co-Ed)

To be a premier technological institution that contribute for sustainable development of our nation & the world at large through achieving excellence in technical education and research which facilitating transformation of students into socially responsible citizens and competent professionals of the highest quality.

MISSION OF FACULTY OF ENGINEERING AND TECHNOLOGY(Co-Ed)

- Provide the affordable and quality education and achieve excellence in teaching learning by designing industry need based curriculum.
- Create good research environment that produces innovations and nurture research scholars.
- Collaborate with industries and other institutions of excellence in order to exchange of expertise.
- To inculcate the significance of human values based on the concept of Dasoha Philosophy of Lord Sharnbasveshwara i.e , "service to Humanity in Service to God" and professional ethics to serve the society.

VISION OF DEPARTMENT

To be recognized globally as a department of computer science and engineering focusing on social issues, embracing new technologies, providing highly talented technocrats and entrepreneurs with sound knowledge in ethics occupying top positions and are adaptable and sustainable in ever changing technological realm. To build a strong research and teaching environment par with the latest needs.

MISSION OF DEPARTMENT

M1: To impart quality technical education by designing curriculum in collaboration with industry requirements
M2: To transform young talents into highly competent individuals who work well in a team or as a single.
M3: To train the computer science Engineering graduates to cater to the needs of society and solve real-world problems by providing strong foundation.
M4: To develop a strong, inter and multi-research culture in the department by collaborating with other department of the university.

PROGRAM EDUCATIONAL OBJECTIVES (PEO'S)

PEO 1	Apply basic knowledge, principles and skills in the field of Computer Science to meet the job specification. (Knowledge / Practical Skills)
PEO 2	Implement the responsibility for solving problems analytically, critically, effective, innovative and market- oriented. (Critical Thinking and Problem Solving / Life-long Learning and Information Management / Entrepreneurship Skills/Researcher)
PEO 3	Acts effectively as an individual or in a group to convey information within the organization and community. (Team Working Skills / Communication Skills)
PEO 4	Practicing good values and ethics in a professional manner in the community and able to act as a leader. (Professional, Social, Ethics, and Humanity / Leadership Skills)

PROGRAM OUTCOMES (PO'S)

PO1: Engineering Knowledge: Apply the knowledge of mathematics, science, engineering fundamentals and computing to solve Computer Science and Engineering related problems.

PO2: Problem Analysis : Identify, formulate , Research literature and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences and engineering sciences.

PO3: Design / Development of Solutions: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for public health and safety, and cultural ,societal

PO4: Conduct Investigations of Complex Problems: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

PO5: Modern tool usage :Create, select and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities related to Computer Science and Engineering with an understanding of the limitations.

PO6: The engineer and society: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

PO7: Environment and sustainability: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of and need for sustainable development.

PO8: Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

PO9: Individual and Team Work: Function effectively as an individual and as a member or leader to diverse teams, and in multidisciplinary settings.

PO10: Communication: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective report and design documentation, make effective presentations, and give and receive clear instructions.

PO11: Project Management and Finance: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

PO12: Life-Long Learning: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

PROGRAM SPECIFIC OUTCOME(PSO'S)

Graduates of the Computer Science and Engineering program will be able to

PSO 1: Understand and recognize the fundamental concepts in basic science, humanities and programming languages like C/C++/java etc. to solve engineering problems.

PSO 2: Design, develop, apply concepts from diverse fields , analyse various computer science engineering design and management principles, mathematical foundations, sustainability and emerging challenges in the computation domain for effective computational solutions for real-life and research problems.

PSO 3: Apply modern programming languages, frameworks, and software tools in engineering and emerging trends principles to develop viable solutions for Information Technology Enabled Services and diverse fields.

Sharnbasva University, Kalaburagi
Scheme of Teaching and Examination 2021-22
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
(Effective from the academic year 2021-22)

Programme: B. Tech: Computer Science and Engineering

V SEMESTER

Sl.. No.	Course Code		Course Title	Teaching Department	Teaching Hours/week			Examination				Credits
					Theory Lecture	Tutorial	Practical/ Drawing	Duration in Hours	CIE Marks	SEE Marks	Total Marks	
					L	T	P					
1	HSS	21CS51	Software Engineering and Project Management	CSE	3			3	50	50	100	03
2	PCC	21CS52	System Software and Compiler Design	CSE	2	1		3	50	50	100	03
3	PCC	21CS53	Computer Networks	CSE	3			3	50	50	100	03
4	PEC	21CS54X	Professional Elective Course-I	CSE	3			3	50	50	100	03
5	OEC	21XX55X	Open Elective Course-I	CSE	4			4	50	50	100	04
6	PCC	21CSL56	System Software and Compiler Design Lab	CSE			2	3	50	50	100	01
7	PCC	21CSL57	Computer Networks Lab	CSE			2	3	50	50	100	01
8	PEC	21CSL58X	Professional Elective Course-1 Lab	CSE			2	3	50	50	100	01
9	PW	21PRJ59	Project-V	CSE			2	3	50	50	100	01
10	AEC	21AEC510X	Ability Enhancement Course-V	CSE			2	3	50	50	100	01
Total					**15**	**1**	**10**	**31**	**500**	**500**	**1000**	**21**

Note: PCC- Programme Core Course, PEC- Professional Elective Course, PW-Project Work, HSS-Humanity and Social Science, OEC- Open Elective Course, AEC-Ability Enhancement Course.
Project (PRJ): Based on the ability /abilities of the student/s and recommendations of the mentor, a single discipline or multidisciplinary mini project can be assigned to an individual student or to a group having not more than 4 students

Professional Elective Course-I			
Course code under 21CS54X	**Course Title**	**Course code under 21CSL58X**	**Course Title**
21CS541	Unix System Programming	**21CSL581**	Unix System Programming Lab
21CS542	Cloud Computing	**21CSL582**	Cloud Computing Lab
21CS543	Mobile Application Development	**21CSL583**	Mobile Application Development Lab

Open Elective Course-I (offered by the Department to other Department students)	
Course code under 21CS55X	Course Title
21CS551	Introduction to Data Structures
21CS552	Introduction to Database Management System
21CS553	Automata Theory and Computability

Ability Enhancement Course-V	
Course code under 21AEC510X	**Course Title**
21AEC5101	Angular JS and Node JS
21AEC5102	Django

AICTE Activity Points: In case students fail to earn the prescribed activity points, eighth semester Grade Card shall be issued only after earning the required activity points. Student shall be admitted for the award of the degree only after the release of the Eighth semester Grade Card.

Sharnbasva University, Kalaburagi Scheme of Teaching and Examination 2021-22
Outcome Based Education (OBE) and Choice Based Credit System (CBCS)
(Effective from the academic year 2021-22)

Programme: B. Tech: Computer Science and Engineering

VI SEMESTER

Sl. No.	Course Code		Course Title	Teaching Department	Teaching Hours/week			Examination				Credits
					Theory Lecture	Tutorial	Practical/ Drawing	Duration in Hours	CIE Marks	SEE Marks	Total Marks	
					L	T	P					
1	PCC	21CS61	Full Stack Web Development	CSE	3			3	50	50	100	03
2	PCC	21CS62	Machine Learning	CSE	2	1		3	50	50	100	03
3	PEC	21CS63X	Professional Elective Course-II	CSE	3			3	50	50	100	03
4	PEC	21CS64X	Professional Elective Course-III	CSE	3			3	50	50	100	03
5	OEC	21XX65X	Open Elective Course-II	CSE	4			3	50	50	100	04
6	PCC	21CSL66	Full Stack Web Development Lab	CSE			2	3	50	50	100	01
7	PCC	21CSL67	Artificial Intelligence and Machine Learning Lab	CSE			2	3	50	50	100	01
8	PW	21PRJ68	Project-VI	CSE			2	3	50	50	100	01
9	HSC	21AU69	Research Methodology and Intellectual Property Rights	Humanities	1			3	50	50	100	01
10	AEC	21ACS610	AWS Framework	CSE			2	3	50	50	100	01
	Internship	--	Internship	To be carried out during vacation *								

Total	**16**	**1**	**8**	**30**	**500**	**500**	**1000**	**21**

Note: PCC-Professional Core Course, PEC-Professional Elective Course, OEC-Open Elective Course, PW-Project Work, HSS-Humanity and Social Science, AEC- Ability Enhancement Course. Internship-To be carried out during the vacation/s of VI and VII semesters or VII and VIII semesters

Project (PRJ): Based on the ability /abilities of the student/s and recommendations of the mentor, a single discipline or multidisciplinary mini project can be assigned to an individual student or to a group having not more than 4 students.

Professional Elective Course-II	
Course code under 21CS63X	**Course Title**
21CS631	Wireless Sensor Networks
21CS632	Computer Graphics and Fundamentals of Image Processing
21CS633	Dot Net Framework for Application Development

Professional Elective Course-III	
Course code under 21CS64X	**Course Title**
21CS641	Cryptography and Network Security
21CS642	Business Intelligence
21CS643	Natural Language Processing

Open Elective Course-II (offered by the Department to other Department students)	
Course code under 21CS65X	**Course Title**
21CS651	Introduction to Big Data
21CS652	Introduction to Cloud Computing
21CS653	System Software and Compiler Design

AICTE Activity Points: In case students fail to earn the prescribed activity points, eighth semester Grade Card shall be issued only after earning the required activity points. Student shall be admitted for the award of the degree only after the release of the Eighth semester Grade Card.

SOFTWARE ENGINEERING AND PROJECT MANAGEMENT
[As per Choice Based Credit System (CBCS)scheme] (Effective from the academic year 2023-2024)
SEMESTER V

Course Code	**21CS51**	**CIE Marks**	**50**
Number of Lecture Hours/Week	**03**	**SEE Marks**	**50**
Total Number of Lecture Hours	**40**	**Exam Hours**	**03**

CREDITS – 03

Course Objectives:

1. Outline software engineering principles and activities involved in building large software programs
2. Describe the process of requirement gathering, classification, specification and validation.
3. Explain the role of DevOps in Agile Implementation.
4. Recognize the importance Project Management with its methods and methodologies.
5. Identify software quality parameters and quantify software using measurements and metrics.

Modules	**Hours**
Module I	
Introduction: Need for Software Engineering, Professional Software Development, Software Engineering Ethics. Case Studies. **Software Evolution:** The evolving role of software, Software, The changing nature of software, Software engineering, A Process Framework, Process Patterns, Process Assessment, Personal and Team Process Models, Process Technology, Product and Process.**Software Processes**: Models: Waterfall Model, Incremental Model and Spiral Model, Process activities.	**08**
Module II	
Requirements Engineering: Requirements Engineering Processes**,** Functional and non-functional requirements**,** The software Requirements Document, Requirements Specification, Requirements validation, Requirements Management **System Models**: Context models, Interaction models, Structural models, Behavioural models, Model-driven engineering, Cost estimation models. **Architectural Design**: Architectural design decisions, Architectural patterns	**08**
Module III	

Software Testing: A Strategic Approach to Software Testing, Strategic Issues, Test Strategies for Conventional Software, Test Strategies for Object -Oriented Software, Validation Testing, System Testing, The Art of Debugging. **Software Evolution**: Evolution processes, Program evolution dynamics, Software maintenance, Legacy system management. **Agile Methodology & DevOps:** Before Agile – Waterfall, Agile Development	**08**
Module IV	
Introduction to Project Management: Introduction, Project and Importance of Project Management, Contract Management, Activities Covered by Software Project Management, Plans, Methods and Methodologies, some ways of categorizing Software Projects, Stakeholders, Setting Objectives, Business Case, Project Success and Failure, Management and Management Control, Project Management lifecycle, Traditional versus Modern Project Management Practices.	**08**
Module V	
Activity Planning: Objectives of Activity Planning, when to Plan, Project Schedules, Sequencing and Scheduling Activities, Network Planning Models, Forward Pass– Backward Pass, identifying critical path, Activity Float, Shortening Project Duration, Activity on Arrow Networks. **Software Quality:**Introduction, The Place of Software Quality in Project Planning, Importance of Software Quality, Software Quality Models, ISO 9126, Quality Management Systems, Process Capability Models, Techniques to Enhance Software Quality, Quality Plans.	**08**

Question paper pattern:

- The question paper will have ten questions.
- There will be 2 questions from each module.
- Each question will have questions covering all the topics under a module
- The students will have to answer 5 full questions, selecting one full question from each module.

Text Books:

1. Roger S. Pressman: Software Engineering-A Practitioners approach, 7th Edition, Tata McGraw Hill.
2. Michael Blaha, James Rumbaugh: Object Oriented Modelling and Design with UML, 2nd Edition, Pearson Education, 2005.
3. Bob Hughes, Mike Cotterell, Rajib Mall: Software Project Management, 6th Edition, McGraw Hill Education, 2018.
4. Deepak Gaikwad, Viral Thakkar, DevOps Tools from Practitioner's Viewpoint, Wiley.
5. Ian Sommerville: Software Engineering, 9th Edition, Pearson Education, 2012.

Reference Books: 1. Pankaj Jalote: An Integrated Approach to Software Engineering, Wiley India. **E-books and Online course materials** 1. https://onlinecourses.nptel.ac.in/noc20_cs68/preview 2. https://www.youtube.com/watch?v=WxkP5KR_Emk&list=PLrjkTql3jnm9b5nr-ggx7Pt1G4UAHeFlJ 3. http://elearning.vtu.ac.in/econtent/CSE.php
Online Courses and Video Lectures 1.http://elearning.vtu.ac.in/econtent/courses/video/CSE/15CS42.html 2. https://nptel.ac.in/courses/128/106/128106012/ (DevOps)

CO#	COURSE OUTCOMES
CO1	Understand the activities involved in software engineering and analyze the role of various process models.
CO2	To build a suitable system model using modelling techniques & architectural designs.
CO3	Describe various software testing methods and to understand the importance of agile methodology and DevOps.
CO4	Illustrate the role of project planning and quality management in software development.
CO5	Understand the importance of activity planning and different planning models.

CO-PO-PSO mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	-	-	-	-	3	-	-	-	-	2	-	3	-
CO2	3	2	3	-	-	-	3	-	-	-	-	2	-	3	-
CO3	3	2	3	-	-	-	3	-	-	-	-	2	-	3	-
CO4	3	-	3	-	-	-	3	-	2	-	3	2	-	3	-
CO5	3	2	-	-	-	-	3	-	-	-	-	2	-	3	-

SYSTEM SOFTWARE AND COMPILER DESIGN
[As per Choice Based Credit System (CBCS)scheme]
(Effective from the academic year 2023-2024)
SEMESTER V

Course Code	**21CS52**	**CIE Marks**	**50**
Number of Lecture Hours/Week	**03**	**SEE Marks**	**50**
Total Number of Lecture Hours	**40**	**Exam Hours**	**03**

CREDITS – 03

Course Objectives:

1. Define System Software.
2. Familiarize with source file, object file and executable file structures and libraries
3. Describe the front-end and back-end phases of compiler and their importance to students

Modules	**Hou**
Module I	
Introduction to System Software, Machine Architecture of SIC and SIC/XE. **Assemblers:** Basic assembler functions, machine-dependent assembler features, machine-independent assembler features, and assembler design options.	**08**
Module II	
Introduction: Language Processors, The structure of a compiler, The evaluation of programming languages, The science of building compilers, Applications of compiler technology. **Lexical Analysis:** The role of lexical analyzer, Input buffering, Specifications of token, recognition of tokens.	**08**
Module III	
Syntax Analysis: Introduction, Role of Parser, Context Free Grammar, Top Down Parsers: Recursive Decent Parsing, LL(1) Grammar, Bottom-Up Parsers: Handle Pruning, Shift Reduce Parsing, LR(0), SLR-Parsing.	**08**
Module IV	
Lex and Yacc –The Simplest Lex Program, Grammars, Parser-Lexer Communication, A YACC Parser, The Rules Section, Running LEX and YACC, LEX and Hand-Written Lexers, Using LEX - Regular Expression, Examples of Regular Expressions, A Word Counting Program, Using YACC – Grammars, Recursive Rules, Shift/Reduce Parsing, What YACC Cannot Parse, A YACC Parser - The Definition Section, The Rules Section, The LEXER, Compiling and Running a Simple Parser, Arithmetic Expressions and Ambiguity.	**08**
Module V	
Syntax Directed Translation, Intermediate code generation, Code generation	**08**

Question paper pattern:

- The question paper will have ten questions.
- There will be 2 questions from each module.
- Each question will have questions covering all the topics under a module.

The students will have to answer 5 full questions, selecting one full question from each module.

Text Books:

1. System Software by Leland. L. Beck, D Manjula, 3rd edition, 2012
2. Alfred V Aho, Monica S. Lam, Ravi Sethi, Jeffrey D. Ullman , Compilers-Princ ip Techniques and Tools, Pearson, 2nd edition, 2007
3. Doug Brown, John Levine, Tony Mason, lex &yacc, O'Reilly Media, October 2012.

Reference Books:

1. Systems programming – Srimanta Pal , Oxford university press, 2016
2. System programming and Compiler Design, K C Louden, Cengage Learning
3. System software and operating system by D. M. Dhamdhere TMG
4. Compiler Design, K Muneeswaran, Oxford University Press 2013.

E-books and Online course materials

1.http://sit.ac.in/html/component/csedept/csecoursematerial/SSCDNotes.pdf

Online Courses and Video Lectures

1.https://onlinecourses.nptel.ac.in/noc21_cs07/preview
2.https://www.youtube.com/playlist?list=PL1A5A6AE8AFC187B7

CO#	COURSE OUTCOMES
CO1	Understand and apply the fundamental concepts of system software, and the core functions of assemblers and loaders.
CO2	Analyze, design, and implement lexical analyzers, evaluate programming languages, and understand the structure and functioning of compilers.
CO3	Develop Top Down and Bottom-Up Parser
CO4	Design and implement lexical analyzers and parsers using Lex and YACC
CO5	Apply syntax-directed translation methods, generate intermediate code, and implement code generation techniques

CO-PO-PSO mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	3	3	-	-	2	3	-	-	-	-	2	2	3	-
CO2	3	3	3	-	-	2	-	-	-	-	-	2	2	3	-
CO3	3	3	3	-	-	2	-	-	-	-	-	2	2	3	-
CO4	3	2	2	-	-	2	-	-	-	-	-	-	2	3	-
CO5	3	3	3	-	-	2	-	-	-	-	-	2	2	3	-

COMPUTER NETWORKS
[As per Choice Based Credit System (CBCS)scheme]
(Effective from the academic year 2023-2024)
SEMESTER V

Course Code	**21CS53**	**CIE Marks**	**50**
Number of Lecture Hours/Week	**03**	**SEE Marks**	**50**
Total Number of Lecture Hours	**40**	**Exam Hours**	**03**

CREDITS – 03

Course Objectives:

1. Explain the protocol stacks (OSI and TCP/IP) for data communication.
2. Discuss the MAC protocols, error detection & correction strategies for data transmission over the networking devices.
3. Describe the standards for data communication with routing protocols.
4. Illustrate the client server communication using TCP or UDP protocols and other application-level protocols.

Modules	**Hours**
Module I	
Introduction to Data Communication and Networking: Internet history and Internet today, Data Communications, Networks, Network Topologies, Classification of Networks, Protocols & Standards. **Layered Architectures**: Tasks, The OSI model, Layers in OSI model, TCP/IP Protocol suite, Addressing. **Introduction to switching**: Circuit Switched Networks, Datagram Networks, Virtual Circuit Networks	**08**
Module II	
Physical Layer: Introduction to Transmission Media, Periodic Analog signals, Digital signals, shanon capacity, performance, Parallel transmission, serial transmission, AM, FM, PM. **Coding:** Line Coding. Introduction to Multiplexing: FDM, WDM, TDM, FHSS, DSSS. Error Detection and Correction: Introduction, cyclic Codes: CRC, Internet checksum. Framing	**08**
Module III	
Data Link Protocols: Point-to-Point Protocol. MAC Protocols: classification of MAC protocols, Random access (ALOHA, CSMA/CD, CSMA/CA), Controlled Access (Reservation, Polling, Token passing), Channelization Protocols (FDMA, TDMA, CDMA) **Introduction to Networking Devices**: Repeaters, Hubs, Bridges, Routers, and High layered switches, Gateways, Virtual LAN. **Standards**: IEEE Standards, Standard Ethernet, Gigabit Ethernet. IEEE 802.11: Architecture	**08**

Module IV	
Network Layer: IPv4 addresses, IP Datagram format, ICMP Messages, Introduction to Mobile IP for mobility management, IPv6 addresses, IPv6 Packet Format, Transition from IPv4 to IPv6 **Routing algorithms** :Distance Vector, Link State and Path vector, Unicast Routing protocols(RIP, OSPF), multicast protocols such DVMRP, PIM.	**08**
Module V	
Transport Layer: Introduction to Stop and Wait, GoBack-N, Selective repeat N, Piggybacking. Services and port numbers, User Datagram Protocol (UDP): UDP Segment, Transmission Control. Protocol (TCP): TCP Segment, TCP Connection Set up, Application of TCP and UDP. TCP flow control, TCP error control, TCP Congestion Control and options. **Application Layer:** Client server programming using UDP and TCP, DNS, SMTP. Introduction to Remote Login Protocols: TELNET Protocol and SSH Protocol.	**08**

Question paper pattern:

- The question paper will have ten questions.
- There will be 2 questions from each module.
- Each question will have questions covering all the topics under a module.

The students will have to answer 5 full questions, selecting one full question from each module.

Text Books:

1. Behrouz A Forouzan,"Data Communications and Networking", 5th Edition, McGraw – Hill, 2016.
2. Nader F. Mir,"Computer and Communication Networks", Pearson Education, 2009

Reference Books:

1. Alberto Leon-Garcia and Indra idjaja,"Communication Networks – Fundamental Concepts and Key Architectures", 2nd Edition Tata McGraw – Hill, 2004.
2. Andrew S. Tanenbaum,"Computer Networks", 4th Edition, Pearson Education, 2005.
3. Larry L. Peterson and Bruce S. Davie, "Computer Networks- A system Approach", 5th Edition, Elsevier, 2012.
4. William Stallings, "Data and Computer Communications", 10th Edition, Pearson Education, 2008.
5. Douglas E. Comer, "Internetworking with TCP/IP Vol.1" , 6th Edition, Pearson, 1995.

E-books and Online course materials

1. IEEE Transactions on Networking.
2. Elsevier Journal of Computer Networks
3. Springer Journal of communications and Information networks

Online Courses and Video Lectures

1. https://www.udemy.com/topic/computer-network/
2. https://www.coursera.org/courses?query=computer%20network
3. https://nptel.ac.in/courses/106/105/106105183/
4. https://www.edx.org/learn/computer-networking

CO#	COURSE OUTCOMES
CO1	Analyze the basic principles of Computer Networks and enumerate the functions of OSI and TCP/IP architectures.
CO2	Analyze the transmission medias, multiplexing methods and apply the properties for error detection and correction.
CO3	Analyze the protocols of data link layer, concepts of networking devices and standards.
CO4	Recognize the need for network layer and evaluate the performance of network and analyze routing algorithms.
CO5	Analyze transport layer services, protocols and principles of application layers to achieve the technological challenge.

CO-PO-PSO mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	3	-	-	-	2	-	-	-	-	-	2	2	3	-
CO2	3	3	3	-	-	2	-	-	-	-	-	2	2	3	-
CO3	3	3	3	-	-	2	-	-	-	-	-	2	2	3	-
CO4	3	3	3	-	-	2	-	-	-	-	-	3	2	3	-
CO5	3	3	-	-	-	2	-	-	-	-	-	3	2	3	-

UNIX SYSTEM PROGRAMMING
[As per Choice Based Credit System (CBCS)scheme] (Effective from the academic year 2023-2024)
SEMESTER V

Course Code	**21CS541**	**CIE Marks**	**50**
Number of Lecture Hours/Week	**03**	**SEE Marks**	**50**
Total Number of Lecture Hours	**40**	**Exam Hours**	**03**

CREDITS – 03

Course Objectives:

1. Understand the basic concept of UNIX architecture and basic UNIX
2. Able to analyze the different types of files and commands used in UNIX.
3.. Students will be use UNIX commands in solving problems.

Modules	**Hours**
Module I	
Introduction: UNIX and ANSI Standards: The ANSI C Standard, The ANSI/ISO C++ Standards, Difference between ANSI C and C++, The POSIX Standards, The POSIX.1 FIPS Standard, The X/Open Standards. UNIX and POSIX APIs: The POSIX APIs, The UNIX and POSIX Development Environment, API Common Characteristics. Unix Basics :UNIX Architecture, Files and Directories, File Types, The UNIX File Attributes, Inodes in UNIX, Application Program Interface to Files, Directory Files, Hard and Symbolic Links	**08**
Module II	
UNIX File APIs: General File APIs, File and Record Locking, Directory File APIs, Device File APIs, FIFO File APIs, Symbolic Link File APIs, General File Class, regfile Class for Regular Files, dirfile Class for Directory Files, FIFO File Class, Device File Class, Symbolic Link File Class, File Listing Program. UNIX Processes: The Environment of a UNIX Process: Introduction, main function, Process Termination, Command-Line Arguments, Environment List, Memory Layout of a C Program, Shared Libraries, Memory Allocation, 57 Environment Variables, setjmp and longjmp Functions, getrlimit, setrlimit Functions, UNIX Kernel Support for Processes.	**08**
Module III	

Process Control : Introduction, Process Identifiers, fork, vfork, exit, wait, waitpid, wait3, wait4 Functions, Race Conditions, exec Functions, Changing User IDs and Group IDs, Interpreter Files, system Function, Process Accounting, User Identification, Process Times, I/O Redirection. Process Relationships: Introduction, Terminal Logins, Network Logins, Process Groups, Sessions, Controlling Terminal, tcgetpgrp and tcsetpgrp Functions, Job Control, Shell Execution of Programs, Orphaned Process Groups	**08**
Module IV	
Signals and Daemon Processes: Signals: The UNIX Kernel Support for Signals, signal, Signal Mask, sigaction, The SIGCHLD Signal and the waitpid Function, The sigsetjmp and siglongjmp Functions, Kill, Alarm, Interval Timers, POSIX.lb Timers. Daemon Processes: Introduction, Daemon Characteristics, Coding Rules, Error Logging, Client-Server Model.	**08**
Module V	
Interprocess Communication – 1: Overview of IPC Methods, Pipes, popen, pclose Functions, Coprocesses, FIFOs, System V IPC, Message Queues, Semaphores. Interprocess Communication – 2: Shared Memory, Client-Server Properties, Stream Pipes, Passing File Descriptors, An Open Server- Version 1, Client-Server Connection Functions.	**08**

Question paper pattern:

- The question paper will have ten questions.
- Therewillbe2questionsfrom each module.
- Each question will have questions covering all the topics under a module.

The students will have to answer 5 full questions, selecting one full question from each module.

Text Books:

1. Terrence Chan: UNIX System Programming Using C++, Prentice Hall India, 1999. (Chapters 1, 5, 6, 7, 8, 9, 10)
2. W. Richard Stevens: Advanced Programming in the UNIX Environment, 2nd Edition, Pearson Education, 2005. (Chapters 7, 8, 9, 13, 14, 15)

Reference Books:

1. Marc J. Rochkind: Advanced UNIX Programming, 2nd Edition, Pearson Education, 2005.
2. Maurice J Bach: The Design of the UNIX Operating System, Pearson Education, 1987. 58
3. Uresh Vahalia: UNIX Internals: The New Frontiers, Pearson Education, 2001.

E-books and Online course materials:
http://www.free-ebooks.net/

Online Courses and Video Lectures
https://nptel.ac.in/courses

CO#	COURSE OUTCOMES
CO1	Demonstrate understanding and usage of UNIX commands, file systems, and basic file management to provide the environment for code execution.
CO2	Implement selected file permissions, shell scripting, and regular expressions for task automation and applying modern engineering techniques to deploy IT tools.
CO3	Apply UNIX file APIs and process control methods in system-level programming.
CO4	Develop client-server applications using user management and IPC methods.
CO5	Utilize signal handling and daemon processes to create reliable client-server programs.

CO-PO-PSO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	2	-	3	-	-	-	3	-	-	-	-	2	2	3	-
CO2	2	-	3	-	3	-	2	-	-	-	-	2	2	3	-
CO3	3	-	3	-	-	-	2	-	-	-	-	2	2	3	-
CO4	2	-	3	-	-	-	2	-	-	-	-	2	2	3	-
CO5	2	-	3	-	-	-	2	-	-	-	-	2	2	3	-

CLOUD COMPUTING
[As per Choice Based Credit System (CBCS)scheme] (Effective from the academic year 2023-2024)
SEMESTER– V

Course Code	**21CS542**	**CIE Marks**	**50**
Number of Lecture Hours/Week	**03**	**SEE Marks**	**50**
Total Number of Lecture Hours	**40**	**Exam Hours**	**03**

CREDITS – 03

Course Objectives:

1. Understand cloud computing and virtualization.
2. Understand types of clouds and cloud application platform.
3. To learn thread programming and task programming concepts.
4. To impart the concepts of Map-Reduce programming.
5. Understand cloud platforms in industry, applications.

Module-I	**Hours**
Introduction to Cloud Computing: Historical Developments, Building Cloud Computing Environments, Application Development, Infrastructure and System Development, Computing Platforms and Technologies. **Virtualization:** Introduction, Characteristics of Virtualized, Environments Taxonomy of Virtualization Techniques, Virtualization and Cloud Computing, Pros and Cons of Virtualization, Technology Examples.	**08**
Module-II	
Cloud Computing Architecture: Introduction, Cloud Reference Model, Types of Clouds , Economics of the Cloud, Open Challenges **Aneka :cloud Application Platform: Framework** overview, Anatomy of the Aneka container, Building Aneka clouds, Cloud Programming and Management	**08**
Module-III	
Concurrent Computing: Thread Programming, Introducing Parallelism for Single Machine Computation, Programming Applications with Threads, **High-Throughput Computing: Task Programming** Task Computing, Task-based Application Models **Data Intensive Computing: Map-Reduce Programming** What is Data- Intensive Computing, Characterizing Data-Intensive Computations, Technologies for Data-Intensive computing.	**08**
Module-IV	
Cloud Infrastructure: Service -and Compliance -level agreements, Responsibility Sharing Between User and Cloud Service Provider, User Experience, Software Licensing **Cloud Security:** Cloud Security Risks, Security: The Top Concern for Cloud Users, Privacy and Privacy Impact Assessment, Trust, Operating	**08**

System Security, virtual Machine Security, Security of Virtualization, Security Risks Posed by Shared Images, Security Risks Posed by Management OS.Xoar:Breaking the Monolithic Design of the TCB	
Module-V	
Cloud Platforms in Industry: Amazon Web Services, Google AppEngine, Microsoft Azure. **Cloud Applications**: Scientific Applications, Business and Consumer Applications.	**08**

Question paper pattern:

- The question paper will have ten questions.
- There will be two questions from each module.
- Each question will have questions covering all the topics under a module.
- The students will have to answer 5 full questions, selecting one full question from each module.

Text Books:

1. Rajkumar Buyya, Christian Vecchiola, and Thamarai Selvi Mastering Cloud. Computing McGraw Hill Education

2. Dan C. Marinescu, Cloud Computing Theory and Practice, Morgan Kaufmann, Elsevier 2013.

Reference Books:

1. **Dr.Kumar Saurabh,cloud computing**
2. **David S.Linthicum, cloud computing and SOA Convergence in your enterprise**

E-books and Online course materials:

- https://www.javatpoint.com/cloud-computing-tutorial
- https://www.tutorialspoint.com/cloud_computing/index.html

Online Courses and Video Lectures

https://www.digimat.in/nptel/courses/video/106105167/L01.html

CO#	COURSE OUTCOMES
CO1	Apply the concept of cloud computing to analyze virtualization concept.
CO2	Identify the types of clouds and deploy the cloud application platform to provide architect solutions.
CO3	Analyze and develop high performance applications using concurrent and parallel computing.
CO4	Identify the security risks related to virtualization in cloud computing.
CO5	Visualization of cloud applications in different industries and fields.

CO-PO-PSO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	3	2	-	-	3	3	-	-	-	-	2	2	3	-
CO2	3	3	2	-	-	3	3	-	-	-	-	2	2	3	-
CO3	3	3	2	-	-	3	3	-	-	-	-	2	2	3	-
CO4	3	3	2	-	-	3	3	-	-	-	-	2	2	3	-
CO5	3	3	2	-	-	3	3	-	-	-	-	2	2	3	-

MOBILE APPLICATION DEVELOPMENT
[As per Choice Based Credit System (CBCS)scheme] (Effective from the academic year 2023-2024)
SEMESTER V

Course Code	**21CS543**	**CIE Marks**	**50**
Number of Lecture Hours/Week	**03**	**SEE Marks**	**50**
Total Number of Lecture Hours	**40**	**Exam Hours**	**03**

CREDITS – 03

Course Objectives:

1.Learn to setup Android application development environment 2.Illustrate user interfaces for interacting with apps and triggering actions 3.Interpret tasks used in handling multiple activities
4. Identify options to save persistent application data
5. Appraise the role of security and performance in Android applications

Module I	**Hours**
Get started, Build your first app, Activities, Testing, debugging and using support libraries	**08**
Module II	
User Interaction, Delightful user experience, Testing your UI	**08**
Module III	
Background Tasks, Triggering, scheduling and optimizing background tasks	**08**
Module IV	
All about data, Preferences and Settings, Storing data using SQLite, Sharing data with content providers, Loading data using Loaders	**08**
Module V	
Permissions, Performance and Security, Firebase and AdMob, Publish	**08**

Question paper pattern:
The question paper will have ten questions. There will be 2 questions from each module.
Each question will have questions covering all the topics under a module.
The students will have to answer 5 full questions, selecting one full question from each module.

Reference Books:
Erik Hellman, "Android Programming – Pushing the Limits", 1st Edition, Wiley India Pvt Ltd, 2014.

Dawn Griffiths and David Griffiths, "Head First Android Development", 1st Edition, O'Reilly SPD Publishers, 2015. J F DiMarzio, "Beginning Android Programming with Android Studio", 4th Edition, Wiley India Pvt Ltd, 2016. ISBN-13: 978-8126565580 Anubhav Pradhan, Anil V Deshpande, " Composing Mobile Apps" using Android, Wiley 2014, ISBN: 978-81-265-4660-2
E-books and Online course materials https://www.gitbook.com/book/google-developer-training/android- developerfundamentals-course-concepts/details (Download pdf file from the above link)
Online Courses and Video Lectures https://www.youtube.com/watch?v=aS_9RbCyHg

CO#	COURSE OUTCOMES
CO1	Create, test and debug Android application by setting up Android development environment.
CO2	Implement adaptive, responsive user interfaces that work across a wide range of devices.
CO3	Understand long running tasks and background work in Android applications
CO4	Demonstrate methods in storing, sharing and retrieving and Analyze data in Android applications.
CO5	Describe the steps involved in publishing Android application to share with the world

CO-PO-PSO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	3	2	-	-	3	3	-	-	-	-	-	2	3	-
CO2	3	3	2	-	-	3	3	-	-	-	-	-	2	3	-
CO3	3	3	2	-	-	3	3	-	-	-	-	-	2	3	-
CO4	3	3	2	-	-	3	3	-	-	-	-	-	2	3	-
CO5	3	3	2	-	-	3	3	-	-	-	-	-	2	3	-

INTRODUCTION TO DATA STRUCTURES
[As per Choice Based Credit System (CBCS)scheme]
(Effective from the academic year 2023-2024)
SEMESTER V

Course Code	**21CS551**	**CIE Marks**	**50**
Number Lecture Hour/Week	**04**	**SEE Marks**	**50**
Number of Lecture Hours	**50**	**Exam Hours**	**03**

CREDITS-04

Course Objectives:

1. To provide the knowledge of basic data structures and their implementations.
2. To understand the importance of data structures in the context of writing efficient programs.
3. To Illustrate Linear representation of data structures like stack, Queues and Linked lists.
4. To Illustrate Non-Linear representation of data structures like Trees and Graphs.
5. To develop skills to apply appropriate data structures in problem-solving.

Modules	**Hours**
Module -I	
Introduction to Data Structures: Types of Data Structures, Operations on Data Structures. Structures, Unions, Pointers, Dynamic Memory Allocation.	**10**
Module -II	
Arrays: Introduction to Arrays, Operations, Applications. Stacks: Introduction to Stacks, Operations, Applications.	**10**
Module -III	
Queues: Introduction to Queues, Operations, Applications. Types of Queues: Circular Queues, Double-Ended Queues, Priority Queues.	**10**
Module -IV	
Linked Lists: Introduction to List, Operations. Types of Link List: Circular Link List, Double Ended Link List.	**10**
Module-V	
Trees: Introduction to Trees, Terminologies, Operations. Graphs: Introduction to Graphs, BFS, DFS, Applications.	**10**

Question paper pattern:

- The question paper will have ten questions.
- There will be 2 questions from each module.
- Each question will have questions covering all the topics under a module

The students will have to answer 5 full questions, selecting one full question from each module.

Text Book:

1. A. M. Padma Reddy, "Systematic Approach To Data Structures Using C", Publishers, 2010.

Reference Books:

1. Data structures using C, E Balagurusamy, McGraw Hill Education (India) Pvt. Ltd, 2013.
2. Ellis Horowitz and Sartaj Sahni, Fundamentals of Data Structures in C, 2nd Ed, Universities Press, 2014.

E-books and Online course materials

1. https://caucse.club/wp-content/uploads/2022/05/Fundamentals-of-Data-Structures-in-C-

Ellis-Horowitz-Sartaj-Sahni-etc.-.pdf 2. **https://pdfcoffee.com/data-structures-with-c-by-schaum-lipschutz-pdf-free.html.**
Online Courses and Video Lectures 1. **https://nptel.ac.in/courses/106102064.**

CO#	COURSE OUTCOMES
CO1	Identify data structures and their operations, and apply structures, unions, pointers, and dynamic memory allocation in C for efficient memory management.
CO2	Implement arrays and stacks, perform basic operations on them, and apply these structures to solve computational problems efficiently.
CO3	Construct and manipulate different types of queues including circular, double-ended, and priority queues, and demonstrate their applications in real-world scenarios.
CO4	Implement and perform insertion, deletion, and traversal operations on singly, circular, and doubly linked lists through programming."
CO5	Illustrate tree and graph structures and apply BFS and DFS traversal algorithms to solve problems in hierarchical and network-based models.

CO-PO-PSO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	3	2	-	-	-	-	-	-	-	-	2	2	3	-
CO2	3	3	2	-	-	-	-	-	-	-	-	2	2	3	-
CO3	3	3	2	-	-	-	-	-	-	-	-	2	2	3	-
CO4	3	3	2	-	-	-	-	-	-	-	-	2	2	3	-
CO5	3	3	2	-	-	-	-	-	-	-	-	2	2	3	-

INTRODUCTION TO DATABASE MANAGEMENT SYSTEM
[As per Choice Based Credit System (CBCS)scheme]
(Effective from the academic year 2023-2024)
SEMESTER V

Course Code	**21CS552**	**CIE Marks**	**50**
Number of Lecture Hours/Week	**04**	**SEE Marks**	**50**
Total Number of Lecture Hours	**50**	**Exam Hours**	**03**

CREDITS – 04

Course Objectives:

1. Describe the features of database management systems.
2. Differentiate between database systems and file systems.
3. Use conceptual modelling tools like ER diagrams and design database schemas based on the conceptual model.
4. Write queries in relational algebra / SQL.
5. Normalize a given database schema.

Modules	**Hours**
Module I	
Introduction: Introduction to database, relational data model, DBMS architecture,data independence and data abstraction, DBA, database users, end users, front end tools	10
Module II	
Data Modelling: Entity types, entity set, attribute and key, relationships, relation types, ER diagrams, database design using ER diagram	**10**
Module III	
Relational Data Model: Relational model concepts, relational constraints, primary and foreign key, candidate key, alternate, composite, super-key. **Data redundancy, Normalization:** 1NF, 2NF, 3NF.	**10**
Module IV	
Structured Query Language: Introduction to SQL, concepts of Data Definition Language (DDL) and Data Manipulation Language (DML), DDL queries like create a data base, drop a database, create table, drop table, alter table, DML queries like inserting data in a table, update in a table, delete data from a table, filter data.	**10**
Module V	
Structured Query Language (continued..): Create relationships between database tables, auto increment, check, Null values, aggregate functions - min, max, count, average, sum, nested sub- queries, group by, having, exists, case, order by. Join operations - inner, left join, right join, natural join and Cartesian product. Overview of forms and reports.**Introduction to Transaction Processing,** Transaction and System concepts, Desirable properties of Transactions, Characterizing schedules based on	**10**

recoverability, Characterizing schedules based on Serializability, Transaction support in SQL.	

Question paper pattern:

- The question paper will have ten questions.
- There will be 2 questions from each module.
- Each question will have questions covering all the topics under a module.

The students will have to answer 5 full questions, selecting one full question from each module.

Text Books:

1. Bayross, I. (2010) SQL, Pl/SQL the Programming Language of Oracle. 4th edition. BPB Publications.
2. Elmsasri, R., & Navathe, S.(2017). Fundamentals of Database Systems. 7th edition. Pearson Education.
3. Silberschatz, A., Korth, H. F., & Sudarshan, S. (2011), Database System Concepts. 6th edition. Tata McGraw-Hill Education.

Reference Books:

1. Date, C.J., Kanman, A. & Swamynathan, S. (2006). An Introduction to Database Systems. 8th edition. Pearson Education.
2. Ramakrishnan, R. Gehrke, J. (2014), Database Management Systems. 3rd edition. Tata McGraw Hill Education.
3. MWidenius, M., Axmark, D., Cole, J., Lentz, A., & Dubois, P. (2002). MySQL Reference Manual.O'Reilly Community Press.

E-books and Online course materials

SQL and Relational Theory(How to Write Accurate SQL code), C.J. Date, O'REILLY Publication
SQL A Beginner's Guide, Andy Oppel, Robert Sheldon, McGraw Hill Publication

Online Courses and Video Lectures

1. https://nptel.ac.in/courses/106/105/106105175/
2. https://onlinecourses.nptel.ac.in/noc21_cs04/
3. https://nptel.ac.in/courses/106/106/106106093/
4. https://www.tutorialspoint.com/dbms/index.htm

CO#	COURSE OUTCOMES
CO1	Illustrate the fundamental concepts of databases, DBMS architecture and the roles of various types of database users.
CO2	Design and interpret Entity-Relationship (ER) models to represent real-world data scenarios and apply them for effective database design.
CO3	Analyze relational data models and apply normalization techniques (1NF, 2NF, 3NF) to minimize data redundancy and improve data integrity.
CO4	Write and execute basic SQL queries
CO5	Perform advanced SQL operations including joins, subqueries, grouping, and transactions

CO-PO-PSO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	3	2	-	-	-	2	-	-	-	-	2	2	3	-
CO2	3	3	2	-	-	-	2	-	-	-	-	2	2	3	-
CO3	3	3	2	-	-	-	2	-	-	-	-	2	2	3	-
CO4	3	3	2	-	-	-	2	-	-	-	-	2	2	3	-
CO5	3	3	2	-	-	-	2	-	-	-	-	2	2	3	-

AUTOMATA THEORY AND COMPUTABILITY
As per Choice Based Credit System (CBCS) scheme
Effective from the academic year 2023-2024)
SEMESTER –V

Course Code	**21CS553**	**CIE Marks**	**50**
Number of Lecture Hours/Week	**04**	**SEE Marks**	**50**
Total Number of Lecture Hours	**50**	**Exam Hours**	**03**

CREDITS – 04

Course Objectives:

1. Introduce the fundamental concepts of Automata Theory, Formal Languages and compiler design
2. Principles Demonstrate Application of Automata Theory and Formal Languages in the field of compiler design
3. Develop understanding of computation through Push Down Automata and Turing Machines
4. Introduce activities carried out in different phases of Phases compiler
5. Identify the undecidability problems.

Modules	**Hours**
Module I	
Why study the Theory of Computation, Languages and Strings: Strings, Languages. A Language Hierarchy, Computation, **Finite State Machines (FSM)**: Deterministic FSM, Regular languages, Designing FSM, Nondeterministic FSMs, From FSMs to Operational Systems, Simulators for FSMs, Minimizing FSMs.	**10**
Module II	
Regular Expressions (RE): what is a RE?, Kleene's theorem, Applications of REs, Manipulating and Simplifying REs. Regular Grammars: Definition, Regular Grammars and Regular languages. Regular Languages (RL) and Non-regular Languages: How many RLs, To show that a language is regular, Closure properties of RLs, to show some languages are not RLs.	**10**
Module III	
Context-Free Grammars (CFG): Introduction to Rewrite Systems and Grammars, CFGs and languages, designing CFGs, simplifying CFGs, proving that a Grammar is correct, Derivation and Parse trees, Ambiguity, Normal Forms. Pushdown Automata (PDA): Definition of non-deterministic PDA, Deterministic and Non-deterministic PDAs, No determinism and Halting, alternative equivalent definitions of a PDA, alternatives that are not equivalent to PDA.	**10**

CO#	COURSE OUTCOMES

Module IV	
Algorithms and Decision Procedures for CFLs: Simplification of CFG, Elimination of ε- production and Unit Symbol, CFLs are closed under Union, Concatenation and Star- closure. CFLs are not closed under Intersection and complementation. **Turing Machine**: Turing machine model, Representation, Language acceptability by TM, design of TM, Techniques for TM construction. Extension to the basic Turing Machine	**10**
Module V	
Program techniques for Turing machine, The model of Linear Bounded automata, Multi- stack Machines, TM with semi-infinite tape. **Decidability:** Definition of an algorithm, decidability, decidable languages, Undecidable languages, halting problem of TM, Post correspondence problem. Complexity: Growth rate of functions, the classes of P and NP, Quantum Computation: quantum computers, Church- Turing thesis. **Applications:** G.1 Defining syntax of programming language, Appendix J: Security	10

Question paper pattern:

- The question paper will have ten questions.
- There will be 2 questions from each module.
- Each question will have questions covering all the topics under a module.
- The students will have to answer 5 full questions, selecting one full question from each module.

Textbooks:

1. John E. Hopcroft, Rajeev Motwani & Jeffrey D Ullman "Introduction to Automata Theory, Languages and Computation" Second Edition.
2. Peter Linz "An Introduction to Formal Languages and Automata" Fifth Edition.
3. A. M. Padma Reddy "Finite Automata and Formal Languages" A Simple Approach.

Reference Books:

1. S. P. Eugene Xavier "Theory of automata, formal languages and computation".
2. Basavaraj S. Anami & Karibasappa K. G "Formal Languages and Automata Theory"

CO1	Design a computational model Finite state machine with conversion between different types of FA and minimize the given FA for any regular language
CO2	Develop regular expressions, languages and apply it for designing compilers.
CO3	Develop context free grammar, push down automata for the given language and conversion between PDA & CFG.
CO4	Simplify CFG & apply the concept of Turing machine for a given Language.
CO5	Analyze and understand decidability and undesirability of various problems with their complexity analysis.

CO-PO-PSO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	3	2	-	-	-	-	-	-	-	-	-	2	3	-
CO2	3	3	2	-	-	-	-	-	-	-	-	-	2	3	-
CO3	3	3	2	-	-	-	-	-	-	-	-	-	2	3	-
CO4	3	3	2	-	-	-	-	-	-	-	-	-	2	3	-
CO5	3	3	2	-	-	-	-	-	-	-	-	-	2	3	-

SYSTEM SOFTWARE AND COMPILER DESIGN LAB
[As per Choice Based Credit System (CBCS)scheme] (Effective from the academic year 2023-2024)
SEMESTER – V

Course Code	**21CSL56**	**CIE Marks**	**50**
Number of Lecture Hours/Week	**02**	**SEE Marks**	**50**
Total Number of Lecture Hours	**30**	**Exam Hours**	**03**

CREDITS-01

Course Objectives:

1. To make students familiar with Lexical Analysis and Syntax Analysis phases of Compiler Design and implement programs on these phases using LEX & YACC tools and/or C/C++/Java
2. To understand the various phases in the design of a compiler.
3. To understand the design of top-down and bottom-up parsers.
4. To understand syntax directed translation schemes.

PART-A

Execute the following programs using LEX:

1. a.Program to count the number of characters, words, spaces and lines in a given input file.
b.Program to count the numbers of comment lines in a given C program. Also eliminate them and copy the resulting program into separate file.

2.a.Program to recognize a valid arithmetic expression and to recognize the identifiers and operators present. Print them separately.
b.Write a LEX program to scan reserved words and identifiers of C language

Execute the following programs using YACC:

3. Program to evaluate an arithmetic expression involving operators +, -, * and /.
4. Program to recognize a valid variable, which starts with a letter, followed by any number of letters or digits.
5. a.Program to recognize strings 'aaab', 'abbb', 'ab' and 'a' using the grammar (anbn, n>= 0).
b.Program to recognize the grammar (anb, n>= 10).

PART -B

6. Design, develop and implement program to construct Predictive / LL(1)Parsing Table for the grammar rules: $A \rightarrow aBa, B \rightarrow bB|\varepsilon$. Use this table to parsethe sentence: abba$

7. Design, develop and implement program to demonstrate Shift Reduce Parsing technique for the grammar rules: $E \rightarrow E + T|T T \rightarrow T * F|F$, $F \rightarrow (E)|id$ and parse the sentence: id + id * id.

8. Design, develop and implement syntax-directed definition of "if E then S1" and "if E then S1 else S2"

9. Write a yacc program that accepts a regular expression as input and produce its parse tree as output.

10. Design, develop and implement a program to generate the machine code usingTriples for the statement A = -B * (C +D) whose intermediate code in three-addressform:

T1 = -B T2 = C +
D
T3 = T1 + T2 A = T3

Conduct of Practical Examination:

• Experiment distribution

a) For laboratories having only one part: Students are allowed to pick one experiment from the lot with equal opportunity.

b) For laboratories having PART A and PART B: Students are allowed to pick one experiment from PART A and one experiment from PART B, with equal opportunity.

• Change of experiment is allowed only once and marks allotted for procedure to be made zero of the changed part only.

• Marks Distribution

SEE are mentioned here, writeup-15%, Conduction procedure and result in -70%, Viva-voce 15% of maximum marks. SEE for practical shall be evaluated for 5 0 marks

CO#	COURSE OUTCOMES
CO1	Demonstrate theoretical concept of System Software and Compiler Design through series of experiment
CO2	Develop a program for LEX and YaCC using the programming language.
CO3	Debug and troubleshoot issues effectively.
CO4	Analyze the data and interpret the results.
CO5	Prepare a well-organized laboratory report.

CO-PO-PSO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	2	3	3	-	-	-	-	-	-	-	-	-	2	-	3
CO2	2	3	3	-	-	-	-	-	-	-	-	-	2	-	3
CO3	2	3	3	-	-	-	-	-	-	-	-	-	2	-	3
CO4	2	3	3	-	-	-	-	-	-	-	-	-	2	-	3
CO5	2	3	3	-	-	-	-	-	-	-	-	-	2	-	3

COMPUTER NETWORKS LAB
[As per Choice Based Credit System (CBCS)scheme] (Effective from the academic year 2023-2024)
SEMESTER V

Course Code	**21CSL57**	**CIE Marks**	**50**
Number of Lecture Hours/Week	**02**	**SEE Marks**	**50**
Total Number of Lecture Hours	**30**	**Exam Hours**	**03**

CREDITS-01

Course Objectives:

1. Explain the ns3 simulator, installation and its application.
2. Illustrate the creation of point to point link, TCP, UDP protocols its connection.
3. Demonstrate the connection establishment of network computing devices.
4. Discuss tracking, testing, analyzing the network.

Simulation programs using NS3 simulator

1. Introduction to: (a) discrete event simulation, (b) ns3, (c) ns3 Installation, (d) NetAnim.
2. Write a NS3 program to connect two nodes with a point to point link, which have unique interface. Analyze the network performance using UDP client server
3. Write a NS3 program to demonstrate bus topology. Analyze the performance using UDP based applications
4. Write a NS3 program to demonstrate star topology. Analyze the performance using UDP based applications.
5. Write a NS3 program to implement FTP using TCP bulk transfer, Analyze the performance
6. Write a NS3 program to connect two nodes with a point to pointlink, which have unique interface. Analyse the traffic control using TCP by changing suitable parameters.
7. Write NS 3 Program to configure two nodes on an 802.11b physical layer, with802.11b NICs in adhoc mode, and by default, sends one packet of 1000 (application) bytes to the other node. The physical layer is configured to receive at a fixed RSS (regardless of the distance and transmit power); therefore, changing position of the nodes has no effect. Analyze the performance
8. Install packet tracer, and consider a topology and configure VLAN.
9. Install NMAP, and execute atleast 10 commands to demonstrate the scanning of networks hosts and ports.

Conduct of Practical Examination:

• Experiment distribution
   a) For laboratories having only one part: Students are allowed to pick one experiment from the lot with equal opportunity.
   b) For laboratories having PART A and PART B: Students are allowed to pick one experiment from PART A and one experiment from PART B, with equal opportunity.

• Change of experiment is allowed only once and marks allotted for procedure to be made zero of the changed part only.

• Marks Distribution

SEE are mentioned here, writeup-15%, Conduction procedure and result in -70%, Viva-voce 15% of maximum marks. SEE for practical shall be evaluated for 50 marks

CO1	Demonstrate various networking concepts through a series of experiments for communication.
CO2	Develop Computer network programs using various software tools like NS3, Packet tracer, NetAnim etc
CO3	Debug and troubleshoot software issues effectively.
CO4	Analyze the data and interpret the results.
CO5	Prepare a well-organized laboratory report.

CO-PO-PSO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	2	3	3	-	-	-	-	-	-	-	-	2	2	-	3
CO2	2	3	3	-	-	-	-	-	-	-	-	2	2	-	3
CO3	2	3	3	-	-	-	-	-	-	-	-	2	2	-	3
CO4	2	3	3	-	-	-	-	-	-	-	-	2	2	-	3
CO5	2	3	3	-	-	-	-	-	-	-	-	2	2	-	3

UNIX SYSTEM PROGRAMMING LAB
[As per Choice Based Credit System (CBCS)scheme] (Effective from the academic year 2023-2024)
SEMESTER– V

Course Code	**21CSL581**	**CIE Marks**	**50**
Number of Lecture Hours/Week	**02**	**SEE Marks**	**50**
Total Number of Lecture Hours	**30**	**Exam Hours**	**03**

CREDITS-01

Course Objectives:

1. To be able to introduce Unix System Programming basics and program design with functions.
2. To understand a range of Object-Oriented Programming, as well as in-depth data, file and communication processing techniques.
3. To understand the high-performance programs designed to strengthen the practical expertise.

1. Write a C/C++ POSIX compliant program to check the following limits:
(i) No. of clock ticks
(ii) Max. no. of child processes
(iii) Max. path length
(iv) Max. no. of characters in a file name
(v) Max. no. of open files/ process

2. Write a C/C++ POSIX compliant program that prints the POSIX defined configuration options supported on any given system using feature test macros.

3. Consider the last 100 bytes as a region. Write a C/C++ program to check whether the region is locked or not. If the region is locked, print pid of the process which has locked. If the region is not locked, lock the region with an exclusive lock, read the last 50 bytes and unlock the region.

4. Write a C/C++ program which demonstrates interprocess communication between a reader process and a writer process. Use mkfifo, open, read, write and close APIs in your program.

5. a) Write a C/C++ program that outputs the contents of its Environment list.
b) Write a C / C++ program to emulate the UNIX ln command.

6. Write a C/C++ program to illustrate the race condition.

7. Write a C/C++ program that creates a zombie and then calls system to execute the ps command to verify that the process is zombie.

8. Write a C/C++ program to avoid zombie process by forking twice.

9. Write a C/C++ program to implement the system function.

10. Write a C/C++ program to set up a real-time clock interval timer using the alarm

API.
Conduct of Practical Examination: • Experiment distribution a) For laboratories having only one part: Students are allowed to pick one experiment from the lot with equal opportunity. b) For laboratories having PART A and PART B: Students are allowed to pick one experiment from PART A and one experiment from PART B, with equal opportunity. • Change of experiment is allowed only once and marks allotted for procedure to be made zero of the changed part only. • Marks Distribution SEE are mentioned here, writeup-15%, Conduction procedure and result in -70%, Viva-voce 15% of maximum marks. SEE for practical shall be evaluated for 50 marks

CO#	COURSE OUTCOMES
CO1	Understand fundamental POSIX system concepts such as process control, file handling, inter-process communication, and system limits in a UNIX environment.
CO2	Design and develop POSIX-compliant system programs
CO3	Debug and troubleshoot issues effectively.
CO4	Analyze the data and interpret the results.
CO5	Prepare a well-organized laboratory report.

CO-PO-PSO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	2	3	3	-	-	-	-	-	-	-	-	2	2	-	3
CO2	2	3	3	-	-	-	-	-	-	-	-	2	2	-	3
CO3	2	3	3	-	-	-	-	-	-	-	-	2	2	-	3
CO4	2	3	3	-	-	-	-	-	-	-	-	2	2	-	3
CO5	2	3	3	-	-	-	-	-	-	-	-	2	2	-	3

CLOUD COMPUTING LAB
[As per Choice Based Credit System (CBCS)scheme] (Effective from the academic year 2023-2024)
SEMESTER– V

Course Code	**21CSL582**	**CIE Marks**	**50**
Number of Lecture Hours/Week	**02**	**SEE Marks**	**50**
Total Number of Lecture Hours	**30**	**Exam Hours**	**03**

CREDITS-01

Course Objectives:

1. To learn the design and development process involved in creating a cloud based applications.
2. To learn to implement and use parallel programming using Hadoop.
3. Learn the concepts of multithread and Map Reduce Programming.

1. Install Virtual box/VMware workstation with different flavours of linux or windows OS on top of windows7 or 8.
2. Install a C compiler in the virtual machine created using virtual box and execute simple programs.
3. Install Google App Engine. Create hello world app and other simple web applications using java/python.
4. Use GAE launcher to launch the web applications.
5. Simulate a cloud scenario using cloudSim and run a scheduling algorithm that is not present in CloudSim.
6. Find a procedure to transfer the files from one virtual machine to another virtual machine.
7. Find a procedure to launch virtual machine using trystack (online OpenStack Demo Version)
8. Install Hadoop single node cluster and run simple applications like word count.
9. Develop a java program to demonstrate simple thread programming.
10. Develop a java program to demonstrate Multithread.
11. Develop a java program to demonstrate simple MapReduce programming

Conduct of Practical Examination:

• Experiment distribution
   a) For laboratories having only one part: Students are allowed to pick one experiment from the lot with equal opportunity.
   b) For laboratories having PART A and PART B: Students are allowed to pick one experiment from PART A and one experiment from PART B, with equal opportunity.

• Change of experiment is allowed only once and marks allotted for procedure to be made zero of the changed part only.

• Marks Distribution

SEE are mentioned here, writeup-15%, Conduction procedure and result in -70%, Viva-voce 15% of maximum marks. SEE for practical shall be evaluated for 5 0 marks

CO#	COURSE OUTCOMES
CO1	Demonstrate the concepts of virtualization, threading concepts in cloud.
CO2	Design and develop solutions to given problems related with cloud computing
CO3	Debug syntactical errors, and troubleshoot programming issues effectively.
CO4	Analyze the programs and interpret the results
CO5	Prepare a well-organized Cloud Computing laboratory report

CO-PO-PSO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	2	3	3	-	-	-	-	-	-	-	-	-	2	-	3
CO2	2	3	3	-	-	-	-	-	-	-	-	-	2	-	3
CO3	2	3	3	-	-	-	-	-	-	-	-	-	2	-	3
CO4	2	3	3	-	-	-	-	-	-	-	-	-	2	-	3
CO5	2	3	3	-	-	-	-	-	-	-	-	-	2	-	3

MOBILE APPLICATION DEVELOPMENT LAB
[As per Choice Based Credit System (CBCS)scheme] (Effective from the academic year 2023-2024)
SEMESTER V

Course Code	**21CSL583**	**CIE Marks**	**50**
Number of Lecture Hours/Week	**02**	**SEE Marks**	**50**
Total Number of LectureHours	**30**	**Exam Hours**	**03**
Credits-01			

Course Objectives:

1. Learn and acquire the art of android programming
2. Configure Android studio to run the applications.
3. Understand and implement android's user interface functions.
4. Create, modify and query on SQlite database
5. Inspect different methods of sharing data using services.

1. Create an Android application to design a visiting card. The Visiting card should have a company logo at the top right corner. The company name should be displayed in Capital letters, aligned to the center. Information like the name of the employee, job title, phone number, address, email, fax and the website address is to be displayed. Insert a horizontal line between the job title and the phone number
2. . Develop an android application using controls like Button, TextView, EditText for designing a calculator having basic functionality like addition, subtraction, Multiplication and division.
3. Create a SIGN up activity with Username and password. Validation of password should happen based on the following rules:
   - Password should contain uppercase and lowercase letters.
   - Password should contain letters and numbers.
   - Password should contain special characters.
   - Minimum length of the password (the default value is 8)

   On successful SIGN UP proceed to the next login activity. Here the user should SIGN IN using the Username and Password created during signup activity. If the Username and password are matched then navigate to the next activity which displays a message saying "Successful Login" or else display a toast message saying "Login Failed". The user is given only two attempts and after that display a toast message saying "Failed Login Attempts" and disable the SIGN IN button. Use bundle to transfer information from one activity to another.
4. Develop an application to set an image as wallpaper. On click of a button, the wallpaper image should start to change randomly every 30 seconds.
5. Write a program to create an activity with two buttons START and STOP. On pressing of the START button, the activity must start the counter by displaying the numbers from one and the counter must keep on counting until the STOP button is pressed. Display the counter value in a Text View control.
6. Create two files of XML and JSON type with values for City Name, Latitude,

Longitude, Temperature and Humidity. Develop an application to create an activity with two buttons to parse the XML and JSON files which when clicked should display the data in their respective layouts side by side.

7. Develop a simple application with one Edit Text so that the user can write some text in it. Create a button called "Convert Text to speech" that converts the user input text into voice.
8. Create an activity like a phone dealer with CALL and SAVE buttons. On pressing the CALL button, it must call the phone number and on pressing the SAVE button it must save the number to the phone contacts.

PART- B

1. Write a program to enter Medicine Name, Date and Time of the Day as input from the user and store it in the SQLite database. Input for Time of the Day should be either Morning or Afternoon or Evening or Night. Trigger an alarm based on the date and time of the day and display the Medicine Name.
2. Develop a content provider application with an activity called "Meeting Schedule" which takes date, Time and Meeting Agenda as input from the user and store this information into the SQLite database. Create another application with an activity called "Meeting Info" having Date Picker control, which on the selection of a date should display the Meeting agenda information for that particular date, else it should display a toast message saying "No Meeting on this Date".
3. Create an application to receive an incoming SMS which is notified to the user. On clicking this SMS notification, the message content and the number should be displayed on the screen. Use appropriate emulator control to send the SMS message to your application.
4. Write a program to create an activity having a text box, and also save, open and create buttons. The user has to write some text in the text box. On pressing the create button the text should be saved as a text file in Mustard. On subsequent changes to the text, the save button should be pressed to store the latest content to the same file. On pressing the open button, it should display the contents from the previously stored files in the text box. If the user tries to save the contents in the Textbox to a file without creating it, then a toast message has to be displayed saying "First create a File".
5. Create an application to demonstrate a basic media player that allows the user to forward, backward, play and pause an audio, also, make use of the indicator in the seek bar to move the audio forward or backward as required.
6. Develop an application to demonstrate the use of asynchronous tasks in android. The asynchronous task should implement the functionality of a simple moving banner. On pressing the Start task button, the banner message should stop. Let the banner message be "Demonstration of Asynchronous Task".
7. Develop an application that makes use of the clipboard framework for copying and pasting of the text. The activity consists of two Edit Text controls and two buttons to trigger the copy and paste functionality.
8. Create an AIDL service that calculates car Loan EMI. The formula to calculate EMI is

$$E= P * (r(1+r)^n)/((1+r)^n-1)$$

Were
E= The EMI payable on the car loan amount

P=The car loan principal amount
r = the interest rate value computed on a monthly basis
n = the loan tenure in the form of months

The down payment amount has to be deducted from the principal amount paid towards buying the car. Develop an application that makes use of this AIDL service to calculate the EMI. This application should have four Edit Text to read the Principal amount, Down payment, Interest rate, loan term (in months) and a button named as " Calculate Monthly EMI". On click of this button, the result should be shown in a Text View. Also, calculate the EMI by varying the Loan term and interest rate values.

• Experiment distribution
  a) For laboratories having only one part: Students are allowed to pick one experiment from the lot with equal opportunity.
  b) For laboratories having PART A and PART B: Students are allowed to pick one experiment from PART A and one experiment from PART B, with equal opportunity.

• Change of experiment is allowed only once and marks allotted for procedure to be made zero of the changed part only.

• Marks Distribution

SEE are mentioned here, writeup-15%, Conduction procedure and result in -70%, Viva-voce 15% of maximum marks. SEE for practical shall be evaluated for 50 marks

CO#	COURSE OUTCOMES
CO1	Demonstrate Create, test and debug android application by setting up android development environment
CO2	Implement adaptive, responsive user interfaces that work across a wide range of devices.
CO3	Debug and troubleshoot issues effectively.
CO4	Analyze the data and interpret the results.
CO5	Prepare a well-organized laboratory report.

CO-PO-PSO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	2	3	3	-	-	-	-	-	-	-	-	-	2	-	3
CO2	2	3	3	-	-	-	-	-	-	-	-	-	2	-	3
CO3	2	3	3	-	-	-	-	-	-	-	-	-	2	-	3
CO4	2	3	3	-	-	-	-	-	-	-	-	-	2	-	3
CO5	2	3	3	-	-	-	-	-	-	-	-	-	2	-	3

PROJECT-V [As per Choice Based Credit System (CBCS) scheme] (Effective from the academic year 2023-2024) SEMESTER – V			
Course Code	**21PRJ59**	**CIE Marks**	**50**
Number of Lecture Hours/Week	**02**	**SEE Marks**	**50**
Total Number of Lecture Hours	**30**	**Exam Hours**	**03**
CREDITS – 01			

Course Objectives:

1. Identify real-world problems across programming, databases, and networking domains and understand their business and technical implications.
2. Apply systematic methodologies to design, implement, and optimize solutions.
3. Resolve technical challenges through debugging, research, and collaboration.
4. Take responsibility for specific roles in a team and collaborate effectively to achieve project goals.
5. Present project progress and findings clearly and confidently to both technical and non- technical audiences.
6. Document the entire project in a structured, professional laboratory report.

Project Guidelines:

- Project work shall preferably be batch wise.
- Evaluation is based on concept clarity, system design, implementation, testing, presentation, and documentation quality, with a focus on proper coding standards, teamwork, and effective communication.
- Viva-voce examination in project work shall be conducted batch-wise.
- Minimum requirement of CIE marks for Project work shall be 50% of the maximum marks.
- Students failing to secure a minimum of 50% of the CIE marks in Project work shall not be eligible for the SEE Project examination.
- For a pass in a Project/Viva-voce examination, a student shall secure a minimum of 40% of the maximum marks prescribed.

CO#	COURSE OUTCOMES
CO1	Identify the topic from various domains (example programming databases, networking) to real world problems.
CO2	Develop methodology for the problem.
CO3	Resolve issues that arise during the project.
CO4	Learn to assign and accept roles and responsibilities within a team and write a good technical report.
CO5	Exhibit skills in presenting their project findings & progress orally

CO-PO-PSO mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	3	3	-	2	1	-	-	2	2	2	-	-	-	3
CO2	3	3	3	-	2	1	-	-	2	2	2	-	-	-	3
CO3	3	3	3	-	2	1	-	-	2	2	2	-	-	-	3
CO4	3	3	3	-	2	1	-	-	2	2	2	-	-	-	3
CO5	3	3	3	-	2	1	-	-	2	2	2	-	-	-	3

Angular JS and Node JS
[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2023-2024)
SEMESTER – V

Course Code	21AEC5101	CIE Marks	50
Number of Lecture Hours/Week	02	SEE Marks	50
Total Number of LectureHours	30	Exam Hours	03

CREDITS – 01

Course Objectives: This course will enable students

- To learn the basics of Angular JS framework.
- To understand the Angular JS Modules, Forms, inputs, expression, data bindings and Filters
- To gain experience of modern tool usage (VS Code, Atom or any other] in developing Web applications

List of Programs

1. Develop Angular JS program that allows user to input their first name and last name and display their full name. **Note**: The default values for first name and last name may be included in the program.
2. Develop an Angular JS application that displays a list of shopping items. Allow users to add and remove items from the list using directives and controllers.**Note**: The default values of items may be included in the program.
3. Develop a simple Angular JS calculator application that can perform basic mathematical operations (addition, subtraction, multiplication, division) based on user input.
4. Write an Angular JS application that can calculate factorial and compute square based on given user input.
5. Develop AngularJS application that displays a details of students and their CGPA. Allow users to read the number of students and display the count. **Note**: Student details may be included in the program.
6. Develop an AngularJS program to create a simple to-do list application. Allow users to add, edit, and delete tasks.**Note**: The default values for tasks may be included in the program.
7. Write an AngularJS program to create a simple CRUD application (Create, Read, Update, and Delete) for managing users.
8. DevelopAngularJS program to create a login form, with validation for the username and password fields.
9. Create an AngularJS application that displays a list of employees and their salaries. Allow users to search for employees by name and salary. **Note**: Employee details may be included in the program.
10. Create AngularJS application that allows users to maintain a collection of items. The application should display the current total number of items, and this count should automatically update as items are added or removed. Users should be able to add items to the collection and remove them as needed.

Conduct of Practical Examination:

• Experiment distribution

a) For laboratories having only one part: Students are allowed to pick one experiment from the lot with equal opportunity.
b) For laboratories having PART A and PART B: Students are allowed to pick one experiment from PART A and one experiment from PART B, with equal opportunity.

• Change of experiment is allowed only once and marks allotted for procedure to be made zero of the changed part only.

• Marks Distribution

SEE are mentioned here, writeup-15%, Conduction procedure and result in -70%, Viva-voce 15% of maximum marks. SEE for practical shall be evaluated for 5 0 marks

CO #	Course Outcome
CO1	Understand the core concepts of AngularJS including data binding, directives, controllers, expressions, and form validation for building dynamic web applications.
CO2	Design and develop interactive and responsive web applications using AngularJS features
CO3	Debug and troubleshoot issues effectively.
CO4	Analyze the data and interpret the results.
CO5	Prepare a well-organized laboratory report.

CO-PO-PSO mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	2	2	3	-	2	2	2	-	-	2	-	3	-	-	3
CO2	2	2	3	-	2	2	2	-	-	2	-	3	-	-	3
CO3	2	2	3	-	2	2	2	-	-	2	-	3	-	-	3
CO4	2	2	3	-	-	2	2	-	-	2	-	3	-	-	3
CO5	2	2	3	-	-	2	2	-	-	2	-	3	-	-	3

DJANGO
[As per Choice Based Credit System (CBCS)scheme] (Effective from the academic year 2023-2024)
SEMESTER V

Course Code	**21AEC5102**	**CIE Marks**	**50**
No of Teaching Hours/Week	**02**	**SEE Marks**	**50**
Total No. of Hours	**30**	**Exam Hours**	**03**
Credits:01			

Course Objectives:

1. To learn the basics of Django.
2. To understand the Django models.
3. To Understand Django Admin interfaces.
4. To Understand Advanced Django Concepts.
5. To understand basics of MVT.

1. **Setting Up Your Environment**: Install Python and pip, Install Django using pip, Create a virtual environment
2. **Django Basics:** Project vs. App in Django, creating a new Django project, Creating a Django App
3. **Understanding project structure**: Running the development server
4. **Models and Database**: Creating models, Migrating databases, Django Admin Interface
5. **Views and Templates**: Views, Creating views, URL patterns, Class-based views vs. Function-based views. Templates: Template language basics, Template inheritance, Template tags and filters. Static files in Django
6. **Forms and User Authentication**: Forms, Creating HTML forms, Django form handling Form validation.
7. **User Authentication**: Creating user authentication views, User registration and login, Using Djangos built-in authentication system.
8. **Advanced Concepts**: Django ORM, Querying the database using Django ORM, Aggregations and annotations, Django REST Framework: Introduction to building APIs, setting up Django REST Framework, Creating API views and serializers
9. **Project and Deployment**: Building a Simple Project, apply knowledge to build a small project (e.g., a blog or a to-do app), Implementing advanced features like user comments or likes.
10. **Version Control (Git)**: Basics of version control, initializing a Git repository, Committing changes.
11. **Deployment**: Deploying a Django project to a hosting service (e.g., Heroku), Configuring databases and environment variables
12. **Optimization and Caching**: Improving performance with caching, Database query Optimization

Conduct of Practical Examination:

• Experiment distribution
   a) For laboratories having only one part: Students are allowed to pick one experiment from the lot with equal opportunity.
   b) For laboratories having PART A and PART B: Students are allowed to pick one experiment from PART A and one experiment from PART B, with equal opportunity.

• Change of experiment is allowed only once and marks allotted for procedure to be made zero of the changed part only.

• Marks Distribution

SEE are mentioned here, writeup-15%, Conduction procedure and result in -70%, Viva-voce 15% of maximum marks. SEE for practical shall be evaluated for 50 marks

CO#	COURSE OUTCOMES
CO1	Build and deploy scalable Django applications.
CO2	Utilize Django's built-in tools and third-party libraries effectively.
CO3	Implement REST APIs for web and mobile applications.
CO4	Analyze the data and interpret the results.
CO5	Prepare a well organized Django laboratory report.

CO-PO-PSO mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	2	2	3	-	2	2	2	-	-	2	-	3	-	-	3
CO2	2	2	3	-	2	2	2	-	-	2	-	3	-	-	3
CO3	2	2	3	-	2	2	2	-	-	2	-	3	-	-	3
CO4	2	2	3	-	2	2	2	-	-	2	-	3	-	-	3
CO5	2	2	3	-	2	2	2	-	-	2	-	3	-	-	3

FULL STACK WEB DEVELOPMENT [As per Choice Based Credit System (CBCS)scheme] (Effective from the academic year 2023-2024) SEMESTER – VI			
Course Code	**21CS61**	**CIE Marks**	**50**
Number of Lecture Hours/Week	**03**	**SEE Marks**	**50**
Total Number of Lecture Hours	**40**	**Exam Hours**	**03**
CREDITS – 03			
Course Objectives:			
1. Explain the use of learning full stack web development. 2. Make use of rapid application development in the design of responsive web pages. 3.Illustrate Models, Views and Templates with their connectivity in Django for full stack web development. 4. Demonstrate the use of state management and admin interfaces automation in Django. CLO 5. Design and implement Django apps containing dynamic pages with SQL databases.			
Modules			**Hours**
Module-I			
MVC based Web Designing: Web framework, MVC Design Pattern, Django Evolution, Views, Mapping URL to Views, Working of Django URL Confs and Loose Coupling, Errors in Django, Wild Card patterns in URLS.			**08**
Module II			
Django Templates and Models: Template System Basics, Using Django Template System, Basic Template Tags and Filters, MVT Development Pattern, Template Loading, Template Inheritance, MVT Development Pattern. Configuring Databases, Defining and Implementing Models, Basic Data Access, Adding Model String Representations, Inserting/Updating data, Selecting and deleting objects, Schema Evolution			**08**
Module III			
Django Admin Interfaces and Model Forms: Activating Admin Interfaces, Using Admin Interfaces, Customizing Admin Interfaces, Reasons to use Admin Interfaces. Form Processing, Creating Feedback forms, Form submissions, custom validation, creating Model Forms, URLConf Ticks, Including Other URLConfs.			**08**
Module IV			
Generic Views and Django State Persistence: Using Generic Views, Generic Views of Objects, Extending Generic Views of objects, Extending Generic Views. MIME Types, Generating Non-HTML contents like CSV and PDF, Syndication Feed Framework, Sitemap framework, Cookies, Sessions, Users and Authentication.			**08**
Module V			
jQuery and AJAX Integration in Django: Ajax Solution, Java Script, XHTML Http Request and Response, HTML, CSS, JSON, I Frames,			**08**

Settings of Java Script in Django, jQuery and Basic AJAX, jQuery AJAX Facilities, Using jQuery UI Autocomplete in Django	

Question paper pattern:

The question paper will have ten questions each question is set for 20 marks.
There will be 2 full questions from each module each of the questions under a module (with a maximum of 3 sub-questions), **should have a mix of topics under that module**. The students have to answer 5 full questions, selecting one full question from each module.
Marks scored by the student shall be proportionally scaled down to 50 Marks.

Text Books:

1. Adrian Holovaty, Jacob Kaplan Moss, The Definitive Guide to Django: Web Development Done Right, Second Edition, Springer-Verlag Berlin and Heidelberg GmbH & Co. KG Publishers, 2009
2. Jonathan Hayward, Django Java Script Integration: AJAX and jQuery, First Edition, Pack Publishing, 2011

Reference Books:

1. Aidas Berborites, Jake Kronika, Django 3 Web Development Cookbook, Fourth Edition, Packet Publishing, 2020
2. William Vincent, Django for Beginners: Build websites with Python and Django, First Edition, Amazon Digital Services, 2018
3. Antonio Mele, Django3 by Example, 3rd Edition, Pack Publishers, 2020
4. Arun Ravindran, Django Design Patterns and Best Practices, 2nd Edition, Pack Publishers, 2020.
5. Julia Elman, Mark Lavin, Light weight Django, David A. Bell, 1st Edition, Oreily Publications, 2014

Books and Online course materials:

1. MVT architecture with Django: https://freevideolectures.com/course/3700/django-tutorials
2. Using Python in Django: https://www.youtube.com/watch?v=2BqoLiMT3Ao
3. Model Forms with Django: https://www.youtube.com/watch?v=gMM1rtTwKxE
4. Real time Interactions in Django: https://www.youtube.com/watch?v=3gHmfoeZ45k
5. AJAX with Django for beginners: https://www.youtube.com/watch?v=3VaKNyjlxAU

CO#	Course Outcomes
CO1	Understand the working of MVT based full stack web development with Django.
CO2	Designing of Models and Forms for rapid development of webpages.
CO3	Analyze the role of Template Inheritance and Generic views for developing full stack web applications.
CO4	Apply the Django frame work libraries to render non-HTML contents like CSV and PDF.
CO5	Perform jQuery-based AJAX integration to Django Apps to build responsive full stack web applications.

CO-PO-PSO mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	3	-	-	2	2	-	-	-	-	2	2	3	-
CO2	3	2	3	-	-	2	2	-	-	-	-	2	2	3	-
CO3	3	2	3	-	-	2	2	-	-	-	-	2	2	3	-
CO4	3	2	3	-	-	2	2	-	-	-	-	3	2	3	-
CO5	3	2	3	-	-	2	2	-	-	-	-	3	2	3	-

MACHINE LEARNING **[As per Choice Based Credit System (CBCS)scheme] (Effective from the academic year 2023-2024)** **SEMESTER – VI**			
Course Code	**21CS62**	**CIE Marks**	**50**
Number of Lecture Hours/Week	**03**	**SEE Marks**	**50**
Total Number of Lecture Hours	**40**	**Exam Hours**	**03**
CREDITS – 03			
Course Objectives:			
1. Become familiar with AI toward problem solving, inference, perception, knowledge representation, and learning. 2. To interpret the different supervised classification methods and tree-based models 3. To understand concept learning, ANN, Bayes classifier, k nearest neighbor.			
Modules			**Hours**
Module I			
Introduction: Well posed learning problems, Designing a Learning system, Perspective and Issues in Machine Learning. **Concept Learning:** Concept learning task, Concept learning as search, Find-S algorithm, Version space, Candidate Elimination algorithm, Inductive Bias.			**08**
Module II			
Decision Tree Learning: Decision tree representation, Appropriate problems for decision tree learning, Basic decision tree learning algorithm, hypothesis space search in decision tree learning, Inductive bias in decision tree learning, Issues in decision tree learning.			**08**
Module III			
Artificial Neural Networks: Introduction, Neural Network representation, Appropriate problems, Perceptrons, Back propagation algorithm.			**08**
Module IV			
Bayesian Learning: Introduction, Bayes theorem, Bayes theorem and concept learning, ML and LS error hypothesis, ML for predicting probabilities, MDL principle, Naive Bayes classifier, Bayesian belief networks, EM algorithm			**08**
Module V			

Evaluating Hypothesis: Motivation, Estimating hypothesis accuracy, Basics of sampling theorem, General approach for deriving confidence intervals, Difference in error of two hypothesis, Comparing learning algorithms .**Instance Based Learning:** Introduction, k-nearest neighbor learning, locally weighted regression, radial basis function, cased-based reasoning, **Reinforcement Learning:** Introduction, Learning Task, Q Learning	**08**

Question paper pattern:

The question paper will have ten questions each question is set for 20 marks.
There will be 2 full questions from each module each of the questions under a module (with a maximum of 3 sub-questions), **should have a mix of topics under that module**. The students have to answer 5 full questions, selecting one full question from each module.
Marks scored by the student shall be proportionally scaled down to 50 Marks.

Text Books:

Tom M. Mitchell, Machine Learning, India Edition 2013, McGraw Hill Education.

Reference Books:

Trevor Hastie, Robert Tibshirani, Jerome Friedman, h The Elements of Statistical Learning, 2nd edition, springer series in statistics.
Ethem Alpaydın, Introduction to machine learning, second edition, MIT press.

E-books and Online course materials:

https://www.drssridhar.com/?page_id=1053
https://www.universitiespress.com/resources?id=9789393330697
https://onlinecourses.nptel.ac.in/noc23_cs18/preview
https://www.geeksforgeeks.org/machine-learning/
https://www.w3schools.com/python/python_ml_getting_started.asp
https://www.tutorialspoint.com/machin

CO#	Course Outcomes
CO1	Apply ML algorithms, address challenges, and use concept learning techniques to solve real-world problems.
CO2	Design solutions for classification problems using decision trees
CO3	Applying the knowledge of perceptrons and backpropagation algorithm to solve the real world problem .
CO4	Apply Bayesian learning using bayes theorem, naive bayes classifier and EM Algorithm.
CO5	Evaluate the hypothesis and explore the fundamentals of instance based and reinforcement learning

CO-PO-PSO mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	3	-	-	-	2	-	-	-	-	2	2	3	-
CO2	3	2	3	-	-	-	2	-	-	-	-	2	2	3	-
CO3	3	2	3	-	-	-	2	-	-	-	-	2	2	3	-
CO4	3	2	3	-	-	-	2	-	-	-	-	2	2	3	-
CO5	3	2	3	-	-	-	2	-	-	-	-	2	2	3	-

WIRELESS SENSOR NETWORKS
[As per Choice Based Credit System (CBCS)scheme]
(Effective from the academic year 2023-2024)
SEMESTER – VI

Course Code	**21CS631**	**CIE Marks**	**50**
Number of Lecture Hours/Week	**03**	**SEE Marks**	**50**
Total Number of LectureHours	**40**	**Exam Hours**	**03**

CREDITS – 03

Course Objectives:

1. To provide an overview about sensor networks and emerging technologies.
2. To study about the node and Network Architecture of sensor nodes and its executionenvironment.

Modules	**Hours**
Module I	
INTRODUCTION: Unique Constraints and challenges, Advantages of Sensor Networks, Sensor Networks Applications, Collaborative Processing, Key Definitions of Sensor Networks.	**08**
Module II	
CANONICAL PROBLEM: LOCALIZATION AND TRACKING: A Tracking Scenario, Problem Formulation, Distributed Representation and Inference of States, Distributed Representation and Inference of States, Tracking Multiple Objects, Sensor Models, Performance Comparison and Metrics.	**08**
Module III	
NETWORKING SENSORS: MAC Protocols for Wireless Sensor Networks, Low Duty Cycle Protocols And Wakeup Concepts – SMAC, -B-MAC Protocol, IEEE 802.15.4 standard and ZigBee, the Mediation Device Protocol, Wakeup Radio Concepts, Address and Name Management, Assignment of MAC Addresses, Routing Protocols Energy-Efficient Routing, Geographic Routing.	**08**
Module IV	

INFRASTRUCTURE ESTABLISHMENT: Topology Control, Clustering, Time Synchronization, Localization and Positioning, SensorTasking and Control	**08**
Module V	
SENSOR NETWORK PLATFORMS AND TOOLS: Sensor Node Hardware – Berkeley Motes, Programming Challenges, Node-level software platforms, Node level Simulators, State-centric programming.	**08**

Question paper pattern:

- The question paper will have ten questions.
- There will be 2 questions from each module.
- Each question will have questions covering all the topics under a module.
- The students will have to answer 5 full questions, selecting one full question from eachmodule.

Text Books:

1. Feng Zhao & Leonidas J.Guibas, "Wireless Sensor Networks-An Information ProcessingApproach", Elsevier, 2007
2. Holger Karl & Andreas Willig, "Protocols and Architectures for Wireless SensorNetworks",John Wiley, 2005.
3. Waltenegus Dargie , Christian Poellabauer, "Fundamentals Of Wireless Sensor Networks –Theory And Practice", John Wiley & Sons Publications, 2011

Reference Books:

1. KazemSohraby, Daniel Minoli, & TaiebZnati, "Wireless Sensor Networks-Technology,Protocols, and Applications", John Wiley, 2007.
2. Anna Hac, "Wireless Sensor Network Designs", John Wiley, 2003

e- Resources & other digital material

1. http://pages.di.unipi.it/bonuccelli/sensori.pdf

CO#	COURSE OUTCOMES
CO1	Describe the overview of wireless sensor networks and enabling technologies for wireless sensor networks.
CO2	Apply the design principles of WSN architectures and operating systems for simulating environment situations.
CO3	Apply various concepts for assignment of MAC address.
CO4	Select the appropriate infrastructure, topology, joint routing and information aggregation for wireless sensor networks
CO5	Analyze the sensor networks platform and tools state-centric programming.

CO-PO-PSO mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	3	2	-	-	2	2	-	-	-	-	2	2	3	-
CO2	3	3	2	-	-	2	2	-	-	-	-	2	2	3	-
CO3	3	3	2	-	-	2	2	-	-	-	-	2	2	3	-
CO4	3	3	2	-	-	2	2	-	-	-	-	2	2	3	-
CO5	3	3	2	-	-	2	2	-	-	-	-	2	2	3	-

COMPUTER GRAPHICS AND FUNDAMENTALS OF IMAGE PROCESSING **[As per Choice Based Credit System (CBCS)scheme]** **(Effective from the academic year 2023-2024)** **SEMESTER – VI**			
Course Code	**21CS632**	**CIE Marks**	**50**
Number of Lecture Hours/Week	**03**	**SEE Marks**	**50**
Total Number of Lecture Hours	**40**	**Exam Hours**	**03**
CREDITS – 03			
Course Objectives:			
1.Overview of Computer Graphics along with its applications. 2.Exploring 2D and 3D graphics mathematics along with OpenGL API's. 3.Use of Computer graphics principles for animation and design of GUI's. 4.Introduction to Image processing and Open CV. 5.Image segmentation using Open CV.			
Modules			**Hours**
Module I			
Overview: Computer Graphics hardware and software and OpenGL: Computer Graphics: Video Display Devices, Raster-Scan Systems Basics of computer graphics, Application of Computer Graphics. OpenGL: Introduction to OpenGL, coordinate reference frames, specifying two-dimensional world coordinate reference frames in OpenGL, OpenGL point functions, OpenGL line functions, point attributes, line attributes, curve attributes, OpenGL point attribute functions, OpenGL line attribute functions, Line drawing algorithms (DDA, Bradenham's).			**08**
Module II			
2D and 3D graphics with OpenGL: 2D Geometric Transformations: Basic 2D Geometric Transformations, matrix representations and homogeneous coordinates, 2D Composite transformations, other 2D transformations, raster methods for geometric transformations, OpenGL raster transformations, OpenGL geometric transformation's function, 3D Geometric Transformations: Translation, rotation, scaling, composite 3D transformations, other 3D transformations, OpenGL geometric transformations functions			**08**
Module III			
Interactive Input Methods and Graphical User Interfaces: Graphical Input Data, Logical Classification of Input Devices, Input Functions for Graphical Data, Interactive Picture-Construction Techniques, Virtual- Reality Environments, OpenGL Interactive Input-Device Functions, OpenGL Menu Functions, Designing a Graphical User Interface. Computer Animation: Design of Animation Sequences, Traditional Animation Techniques, General Computer-Animation Functions, Computer-Animation Languages, Character Animation, Periodic Motions, OpenGL Animation Procedures.			**08**
Module IV			
Introduction to Image processing: overview, Nature of IP, IP and its related fields, Digital Image representation, types of images. Digital Image			

Processing Operations: Basic relationships and distance metrics, Classification of Image processing Operations.	**08**
Module V	
Image Segmentation: Introduction, classification, detection of discontinuities, Edge detection (up to canny edge detection(included)).	**08**
Question paper pattern: The question paper will have ten questions each question is set for 20 marks. There will be 2 full questions from each module each of the questions under a module (with a maximum of 3 sub-questions), **should have a mix of topics under that module**. The students have to answer 5 full questions, selecting one full question from each module. Marks scored by the student shall be proportionally scaled down to 50 Marks.	
Text Books: 1. Donald D Hearn, M Pauline Baker and WarrenCarithers: Computer Graphics with OpenGL 4th Edition, Pearson, 2014 2. S. Sridhar, Digital Image Processing, second edition, Oxford University press 2016.	
Reference Books: 1. Edward Angel: Interactive Computer Graphics- A Top-Down approach with OpenGL, 5th edition. Pearson Education, 2008 2. James D Foley, Andries Van Dam, Steven K Feiner, John F Huges Computer graphics with OpenGL: Pearson education	
books and Online course materials: https://nptel.ac.in/courses/106/106/106106090/ https://nptel.ac.in/courses/106/102/106102063/ https://nptel.ac.in/courses/106/103/106103224/ https://nptel.ac.in/courses/106/102/106102065/ https://www.tutorialspoint.com/opencv/ (Tutorial, Types of Images, Drawing Functions)	

CO#	COURSE OUTCOMES
CO1	Construct geometric objects using Computer Graphics principles and OpenGL APIs.
CO2	Analyze OpenGL APIs and related mathematics for 2D and 3D geometric Operations on the objects.
CO3	Design GUI with necessary techniques required to animate the created objects
CO4	Implementing the basics of Image processing applications.
CO5	Apply Image segmentation techniques for developing simple applications.

CO-PO-PSO mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	3	2	-	-	2	2	-	-	-	-	2	2	3	-
CO2	3	3	2	-	-	2	2	-	-	-	-	2	2	3	-
CO3	3	3	2	-	-	2	2	-	-	-	-	2	2	3	-
CO4	3	3	2	-	-	2	2	-	-	-	-	2	2	3	-
CO5	3	3	2	-	-	2	2	-	-	-	-	2	2	3	-

DOT NET FRAMEWORK FOR APPLICATION DEVELOPMENT
[As per Choice Based Credit System (CBCS)scheme]
(Effective from the academic year 2023-2024)
SEMESTER – VI

Course Code	**21CS633**	**CIE Marks**	**50**
Number of Lecture Hours/Week	**03**	**SEE Marks**	**50**
Total Number of Lecture Hours	**40**	**Exam Hours**	**03**

CREDITS – 03

Course Objectives:

1. Build applications on Visual Studio .NET platform by understanding the syntax and semantics of C#
2. Demonstrate Object Oriented Programming concepts in C# programming language
3. Design custom interfaces for applications and leverage the available built-in interfaces in building complex applications.
4. Illustrate the use of generics and collections in C#
5. Compose queries to query in-memory data and define own operator behavior

Modules	**Hours**
Module I	
Introducing Microsoft Visual C# and Microsoft Visual Studio 2015: Welcome to C#, Working with variables, operators and expressions, Writing methods and applying scope, Using decision statements, Using compound assignment and iteration statements, Managing errors and exceptions	**08**
Module II	
Understanding the C# object model: Creating and Managing classes and objects, Understanding values and references, Creating value types with enumerations and structures, Using arrays	**08**
Module III	
Understanding parameter arrays, Working with inheritance, Creating interfaces and defining abstract classes, Using garbage collection and resource management	**08**
Module IV	
Defining Extensible Types with C#: Implementing properties to access fields, Using indexers, Introducing generics, Using collections	**08**
Module V	
Defining Extensible Types with C#: Implementing properties to access fields, Using indexers, Introducing generics, Using collections	**08**

Question paper pattern:
The question paper will have ten questions each question is set for 20 marks.
There will be 2 full questions from each module each of the questions under a module

(with a maximum of 3 sub-questions), **should have a mix of topics under that module**. The students have to answer 5 full questions, selecting one full question from each module. Marks scored by the student shall be proportionally scaled down to 50 Marks.
Text Books: John Sharp, Microsoft Visual C# Step by Step, 8th Edition, PHI Learning Pvt. Ltd. 2016
Reference Books: Christian Nagel, "C# 6 and .NET Core 1.0", 1st Edition, Wiley India Pvt Ltd, 2016. Andrew Stellman and Jennifer Greene, "Head First C#", 3rd Edition, O"Reilly Publications, 2013.

CO#	COURSE OUTCOMES
CO1	Build applications on Visual Studio .NET platform by understanding the syntax and semantics of C#
CO2	Demonstrate Object Oriented Programming concepts in C# programming language
CO3	Design custom interfaces for applications and leverage the available built-in interfaces in building complex applications.
CO4	Illustrate the use of generics and collections in C#
CO5	Compose queries to query in-memory data and define own operator behavior

CO-PO-PSO mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	3	2	-	-	2	-	-	-	-	-	2	2	3	-
CO2	3	3	2	-	-	2	-	-	-	-	-	2	2	3	-
CO3	3	3	2	-	-	2	-	-	-	-	-	2	2	3	-
CO4	3	3	2	-	-	2	-	-	-	-	-	2	2	3	-
CO5	3	3	2	-	-	2	-	-	-	-	-	2	2	3	-

CRYPTOGRAPHY & NETWORK SECURITY [As per Choice Based Credit System (CBCS)scheme] (Effective from the academic year 2023-2024) SEMESTER– VI			
Course Code	**21CS641**	**CIE Marks**	**50**
Number of Lecture Hours/Week	**03**	**SEE Marks**	**50**
Total Number of Lecture Hours	**40**	**Exam Hours**	**03**
Credits–03			

Course Objectives:

1. To understand Cryptography, Network Security and its principles
2. To Analyze different Cryptography algorithms
3. To Illustrate Public and Private key cryptography
4. To Explain Key management, distribution and certification
5. To understand necessary Approaches and Techniques to build protection mechanisms in order tosecurecomputer networks.

Modules	Hours
Module I	
Classical Encryption Techniques: Symmetric Cipher Model, Cryptography, Cryptanalysis and Brute- Force Attack, Substitution Techniques, Caesar Cipher, Monoalphabetic Cipher, Playfair Cipher, Hill Cipher, Polyalphabetic Cipher, One Time Pad. **Block Ciphers and the Data Encryption Standard:** Traditional block Cipher structure, Stream Ciphers and Block Ciphers, Motivation for the Feistel Cipher structure, the Feistel Cipher, The data encryption standard, DES encryption, DES decryption, A DES example, results, the avalanche effect, the strength of DES, the use of 56-Bit Keys, the nature of the DES algorithm, timing attacks, Block cipher design principles, number of rounds, design of function F, key schedule algorithm	**08**
Module II	
Public-Key Cryptography and RSA: Principles of public-key cryptosystems. Public-key cryptosystems. Applications for public-key cryptosystems, requirements for public-key cryptosystems. Public-key cryptanalysis. The RSA algorithm, description of the algorithm, computational aspects, the security of RSA. **Other Public-Key Cryptosystems:** Diffie-Hellman key exchange, The algorithm, key exchange protocols, man in the middle attack, Elgamal Cryptographic systems.	**08**

Module III	
distribution scenario, Hierarchical key control, session key lifetime, a transparent key control scheme, decentralized key control, controlling key usage, Symmetric key distribution using asymmetric encryption, simple secret key distribution, secret key distribution with confidentiality and authentication, A hybrid scheme, distribution of public keys, public announcement of public keys, publicly available directory, public key authority, public keys certificates.	**08**
Module IV	
X-509 certificates. Certificates, X-509version 3 Public key infrastructure. **User Authentication:** Remote user Authentication principles, Mutual Authentication, one- way authentication, remote user Authentication using Symmetric encryption, Mutual Authentication, one- way Authentication, **Kerberos**, Motivation, Kerberos version 4, Kerberos version 5, Remote user Authentication using Asymmetric encryption, Mutual Authentication, one-way Authentication.	**08**
Module V	
Electronic Mail Security: Pretty good privacy, S/MIME, **IP Security:** IP Security overview, IP Security policy, Encapsulating Security payload,Combining security associations, Internet key exchange.	**08**
Question paper pattern: • The question paper will have ten questions. • Therewillbe2questionsfrom each module. • Each question will have questions covering all the topics under a module. • The students will have to answer 5full questions, selecting one full question from each module.	
Text Books: William Stallings: Cryptography and Network Security, Pearson 6th edition. **Reference Books:** 1. V. K Pachghare: Cryptography and Information Security, PHI 2nd Edition 2. BehrouzA.Foruzan, Cryptography and Network Security, Tata McGraw Hill 2007	

CO#	COURSE OUTCOMES
CO1	Implement encryption methods like Caesar Cipher and DES.
CO2	Analyze algorithms like RSA and Diffie-Hellman.
CO3	Apply key distribution techniques, manage key control, and ensure confidentiality and authentication in secure communications.
CO4	Assess and verify secure authentication methods, including different protocols.
CO5	Evaluate email security measures and secure IP communications using protocols.

CO-PO-PSO mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	3	2	-	-	2	-	-	-	-	-	2	2	3	-
CO2	3	3	2	-	-	2	-	-	-	-	-	2	2	3	-
CO3	3	3	2	-	-	2	-	-	-	-	-	2	2	3	-
CO4	3	3	2	-	-	2	-	-	-	-	-	2	2	3	-
CO5	3	3	2	-	-	2	-	-	-	-	-	2	2	3	-

BUSINESS INTELLIGENCE
[As per Choice Based Credit System (CBCS)scheme]
(Effective from the academic year 2023-2024)
SEMESTER – VI

Course Code	**21CS642**	**CIE Marks**	**50**
Number of Lecture Hours/Week	**03**	**SEE Marks**	**50**
Total Number of Lecture Hours	**40**	**Exam Hours**	**03**

Credits –03

Course Objectives:

1. Explain the Business Intelligence, Analytics and Decision Support system
2. List the technologies for Decision making, Automated decision systems
3. Explain sentiment analysis techniques
4. Illustrate Multi-criteria Decision making systems, predictive modelling techniques

Modules	**Hours**
Module I	
An Overview of Business Intelligence, Analytics, and Decision Support: Information Systems Support for Decision Making, An Early Framework for Computerized Decision Support, The Concept of Decision Support Systems, A Framework for Business Intelligence, Business Analytics Overview, Brief Introduction to Big Data Analytics	**08**
Module II	
Decision Making: Introduction and Definitions, Phases of the Decision, Making Process, The Intelligence Phase, Design Phase, Choice Phase, Implementation Phase, Decision Support Systems Capabilities, Decision Support Systems Classification, Decision Support Systems Components.	**08**
Module III	
Neural Networks and Sentiment Analysis: Basic Concepts of Neural Networks, Developing Neural Network-Based Systems, Illuminating the Black Box of ANN with Sensitivity, Support Vector Machines, A Process Based Approach to the Use of SVM, Nearest Neighbor Method for Prediction, Sentiment Analysis Overview, Sentiment Analysis Applications, Sentiment Analysis Process,, Sentiment Analysis, Speech Analytics.	**08**
Module IV	
Model-Based Decision Making: Decision Support Systems modeling, Structure of mathematical models for decision support, Certainty, Uncertainty, and Risk, Decision modeling with spreadsheets,Mathematical programming optimization, Decision Analysis with Decision Tables and Decision Trees, Multi-Criteria Decision Making With Pairwise Comparisons.	**08**
Module V	
Automated Decision Systems and Expert Systems: Automated Decision Systems, The Artificial Intelligence field, Basic concepts of Expert Systems, Applications of Expert Systems, Structure of Expert Systems, Knowledge Engineering, Development of Expert Systems.	**08**

Question paper pattern:
The question paper will have ten questions each question is set for 20 marks.

There will be 2 full questions from each module each of the questions under a module (with a maximum of 3 sub-questions), **should have a mix of topics under that module**. The students have to answer 5 full questions, selecting one full question from each module. Marks scored by the student shall be proportionally scaled down to 50 Marks.
Text Books: Ramesh Sharda, Dursun Delen, Efraim Turban, J.E.Aronson,Ting-Peng Liang, David King, "Business Intelligence and Analytics: System for Decision Support", 10th Edition, Pearson Global Edition, 2013
Reference Books: Data Analytics: The Ultimate Beginner's Guide to Data Analytics Paperback – 12 November 2017by Edward Mize.
E-books and Online course materials: https://www.youtube.com/watch?v=zbcCdoHeS4w

CO#	COURSE OUTCOMES
CO1	Understand the concepts Business Intelligence, Analytics and Decision Support
CO2	Analyze the decision making model techniques and decision support system components.
CO3	Apply the neural networks to analyze textual data understand the principles of sentimental analysis.
CO4	Identify the modeling technique for decision analysis
CO5	Understand the Basics about Artificial Intelligence and Expert Systems.

CO-PO-PSO mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	3	2	-	-	2	-	-	-	-	-	2	2	3	-
CO2	3	3	2	-	-	2	-	-	-	-	-	2	2	3	-
CO3	3	3	2	-	-	2	-	-	-	-	-	2	2	3	-
CO4	3	3	2	-	-	2	-	-	-	-	-	2	2	3	-
CO5	3	3	2	-	-	2	-	-	-	-	-	2	2	3	-

Natural Language Processing
[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2023-2024)
SEMESTER– VI

Course Code	**21CS643**	**CIE Marks**	**50**
Number of Lecture Hours/Week	**03**	**SEE Marks**	**50**
Total Number of Lecture Hours	**40**	**Exam Hours**	**03**

Credits–03

Course Objectives:

- Will be able to understand the wide spectrum of problem statements, tasks, and solutionapproaches within NLP
- Will be able to implement and evaluate different NLP applications
- Evaluate various algorithms and approaches for the given task, dataset, and stage of the NLPproduct.
- Understand best practices, opportunities, and the roadmap for NLP from a business and productleader's perspective.

Modules	**Hours**
Module I	
Overview and language modeling: Overview: Origins and challenges of NLP Language and Grammar-Processing Indian Languages- NLP Applications Information Retrieval Language Modeling: Various Grammar- based Language Models-Statistical Language Model.	**08**
Module II	
Word level and syntactic analysis: Word Level Analysis: Regular Expressions Finite-State Automata-Morphological Parsing- Spelling Error Detection and correction-Words and Word classes-Part-of Speech Tagging. Syntactic Analysis: Context-free Grammar-Constituency- Parsing-Probabilistic Parsing.	**08**
Module III	
Extracting Relations from Text: From Word Sequences to Dependency Paths: Introduction, Subsequence Kernels for Relation Extraction, A Dependency-Path Kernel forRelation Extraction and Experimental Evaluation. **Mining Diagnostic Text Reports by Learning to Annotate Knowledge Roles:** Introduction, Domain Knowledge and Knowledge Roles, Frame Semantics and SemanticRole Labeling, Learning to Annotate Cases with Knowledge Roles and Evaluations.	**08**
Module IV	

Evaluating Self-Explanations in I start: Word Matching, Latent Semantic Analysis,and Topic Models,Introduction, I start: Feedback Systems, I start: Evaluation of Feedback Systems,**Textual Signatures: Identifying Text-Types Using Latent Semantic Analysis to Measure the Cohesion of Text Structures:**Introduction, Cohesion, CohMetrix, Approaches to Analyzing Texts, Latent SemanticAnalysis, Predictions, Results of Experiments. **Automatic Document Separation: A Combination of Probabilistic Classification and Finite-State Sequence Modeling:**Introduction, Related Work, Data Preparation, Document Separation as a Sequence Mapping Problem, Results. **Evolving Explanatory Novel Patterns for Semantically-Based Text Mining:** Related Work, A Semantically Guided Model for Effective Text Mining.	**08**
Module V	
INFORMATION RETRIEVAL AND LEXICAL RESOURCES Information Retrieval: Design features of Information Retrieval Systems-Classical, Nonclassical, Alternative Models of Information Retrieval – valuation Lexical Resources: World Net-Frame 0Net- Stemmers-POS Tagger- Research Corpora.	**08**

Question paper pattern:

- The question paper will have ten questions.
- There will be 2 questions from each module.
- Each question will have questions covering all the topics under a module.
- The students will have to answer 5 full questions, selecting one full question from each module.

Text Books:

1. Tanveer Siddiqui, U.S. Tiwary, "Natural Language Processing and Information Retrieval", OxfordUniversity Press, 2008.
2. Anne Kao and Stephen R. Poteet (Eds), "Natural Language Processing and Text Mining", Springer-Verlag London Limited 2007.

Reference Books:

1. Daniel Jurafsky and James H Martin, "Speech and Language Processing: An introduction to NaturalLanguage Processing, Computational Linguistics and Speech Recognition", 2nd Edition, Prentice Hall,2008.
2. James Allen, "Natural Language Understanding", 2nd edition, Benjamin/Cummings publishingcompany, 1995..
3. James Allen," Natural Language Understanding", Benjamin/Cummingsp ublishing company. 2nd edition, 1995
4. Gerald J. Kowalski and Mark.T. Maybury," Information Storage and Retrieval system", Information Storage and Retrieval system 2000.
5. Steven Bird, Ewan Klein, Edward Loper," Natural Language Processing with Python", 2009
6. Christopher D.Manning and Hinrich Schutze," Foundations of Statistical Natural Language Processing, MIT Press" 1999

CO#	COURSE OUTCOMES
CO1	Understand the fundamental concepts and techniques in NLP and evaluate their unique challenges in language modeling
CO2	Evaluate part-of-speech tagging and explore syntactic analysis through context-free grammar and probabilistic parsing.
CO3	Investigate the extraction of relations from text by analyzing word sequences and dependency paths.
CO4	Develop evolving explanatory novel patterns for semantically-based text mining through a guided model approach.
CO5	Design and evaluate features of information retrieval systems by comparing classical, non-classical, and alternative models of information retrieval.

CO-PO-PSO mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	3	2	-	-	2	-	-	-	-	-	2	2	3	-
CO2	3	3	2	-	-	2	-	-	-	-	-	2	2	3	-
CO3	3	3	2	-	-	2	-	-	-	-	-	2	2	3	-
CO4	3	3	2	-	-	2	-	-	-	-	-	2	2	3	-
CO5	3	3	2	-	-	2	-	-	-	-	-	2	2	3	-

INTRODUCTION TO BIG DATA [As per Choice Based Credit System (CBCS)scheme] (Effective from the academic year 2023-2024) SEMESTER VI			
Course Code	**21CS651**	**CIE Marks**	**50**
Number of Lecture Hours/Week	**04**	**SEE Marks**	**50**
Total Number of Lecture Hours	**50**	**Exam Hours**	**03**
CREDITS – 04			
Course Objectives:			
1. Understand Hadoop Distributed File system and examine MapReduce Programming 2. Explore Hadoop tools and manage Hadoop with Ambari 3. Use NoSQL for Big data database management and basic analysis 4. Use of machine learning algorithms for bigdata analytics			
Modules			**Hours**
Module I			
Introduction to Big Data: Big Data, Scalability and Parallel Processing, Designing Data Architecture, Data Sources, Quality, Pre-Processing and Storing, Data Storage and Analysis, Big Data Analytics Applications and Case Studies.			**10**
Module II			
Introduction to Hadoop (T1): Introduction, Hadoop and its Ecosystem, Hadoop Distributed File System, MapReduce Framework and Programming Model, HadoopYarn, Hadoop Ecosystem Tools. **Hadoop Distributed File System Basics (T2):** HDFS Design Features,Components, HDFS User Commands. **Essential Hadoop Tools (T2):** Using Apache Pig, Hive, Sqoop, Flume, Oozie,Hbase.			**10**
Module III			
NoSQL Big Data Management, MongoDB and Cassandra: Introduction, NoSQL Data Store, NoSQL Data Architecture Patterns, NoSQL to Manage BigData, Shared-Nothing Architecture for Big Data Tasks, MongoDB, Databases, Cassandra Databases.			**10**
Module IV			

MapReduce, Hive and Pig: Introduction, MapReduce Map Tasks, Reduce Tasks and MapReduce Execution, Composing MapReduce for Calculations and Algorithms, Hive, HiveQL, Pig.	**10**
Module V	
Machine Learning Algorithms for Big Data Analytics: Introduction, Estimating the relationships, Outliers, Variances, Probability Distributions, and Correlations Regression analysis, Finding Similar Items, Similarity of Sets and Collaborative Filtering, Frequent Item sets and Association Rule Mining. Text, Web Content, Link, and Social Network Analytics: Introduction, Text mining, Web Mining, Web Content and Web Usage Analytics, Page Rank, Structure of Web and analyzing a Web Graph, Social Network as Graphs and Social Network Analytics	**10**

Question paper pattern:

- The question paper will have ten questions.
- There will be two questions from each module.
- Each question will have questions covering all the topics under a module.
- The students will have to answer 5 full questions, selecting one full question from each module.

Text Books:

1. Raj Kamal and Preeti Saxena, "Big Data Analytics Introduction to Hadoop, Spark, and Machine-Learning", McGraw Hill Education, 2018 ISBN: 9789353164966, 9353164966
2. Douglas Eadline, "Hadoop 2 Quick-Start Guide: Learn the Essentials of Big Data Computing in the Apache Hadoop 2 Ecosystem", 1st Edition, Pearson Education, 2016. ISBN13: 978-9332570351

Reference Books:

1. Tom White, "Hadoop: The Definitive Guide", 4thEdition, O'Reilly Media, 2015.ISBN-13: 978-9352130672
2. Boris Lublinsky, Kevin T Smith, Alexey Yakubovich, "Professional Hadoop Solutions", 1st Edition,Wrox Press, 2014ISBN-13: 978-8126551071
3. Eric Sammer, "Hadoop Operations: A Guide for Developers and Administrators",1stEdition,O'Reilly Media, 2012.ISBN-13: 978-9350239261
4. Arshdeep Bahga, Vijay Madisetti, "Big Data Analytics: A Hands-On Approach", 1st Edition, VPT Publications, 2018. ISBN-13: 978-0996025577

E-books and Online course materials:

- https://www.youtube.com/watch?v=fUPV776pY2M
- https://www.youtube.com/watch?v=aReuLtY0YMI

https://www.youtube.com/watch?v=LOuAOZWJ9RA&list=PLxCzCOWd7aiHUUi6ZlansKbDw_cXut0El (Video Lectures)

CO#	COURSE OUTCOMES
CO1	Understand fundamentals of Big Data analytics.
CO2	Investigate Hadoop framework and Hadoop Distributed File system.
CO3	Illustrate the concepts of NoSQL using MongoDB and Cassandra for Big Data.
CO4	Demonstrate the MapReduce programming model to process the big data along with Hadoop tools.
CO5	Use Machine Learning algorithms for real world big data.

CO-PO-PSO mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	3	2	-	-	2	-	-	-	-	-	2	2	3	-
CO2	3	3	2	-	-	2	-	-	-	-	-	2	2	3	-
CO3	3	3	2	-	-	2	-	-	-	-	-	2	2	3	-
CO4	3	3	2	-	-	2	-	-	-	-	-	2	2	3	-
CO5	3	3	2	-	-	2	-	-	-	-	-	2	2	3	-

INTRODUCTION TO CLOUD COMPUTING [As per Choice Based Credit System (CBCS)scheme] (Effective from the academic year 2023-2024) SEMESTER VI			
Course Code	**21CS652**	**CIE Marks**	**50**
Number of Lecture Hours/Week	**04**	**SEE Marks**	**50**
Total Number of Lecture Hours	**50**	**Exam Hours**	**03**
Credits – 04			

Course Objectives:

1. Understand cloud computing and virtualization.
2. Understand types of clouds.
3. To learn cloud security concepts and Application development.
4. To impart the concepts of Data intensive computing, and Map Reduce.
5. Understand cloud platforms in industry, applications.

Modules	Hours
Module I	
Introduction to Cloud Computing: Historical Developments, Building Cloud Computing Environments, Application Development, Infrastructure and System Development, Computing Platforms and Technologies. **Virtualization:** Introduction, Characteristics of Virtualized, Environments Taxonomy of Virtualization Techniques, Virtualization and Cloud Computing, Pros and Cons of Virtualization, Technology Examples.	**10**
Module II	
Cloud Computing Architecture: Introduction, Cloud Reference Model, Types ofClouds , Economics of the Cloud, Open Challenges **Aneka-Cloud Application Platform**: Framework Overview, Anatomy of the AnekaContainer, Building Aneka Clouds, Cloud Programming and Management.	**10**
Module III	
Cloud security, Cloud Application Development: Cloud security risks, Security: The top concern for cloud users, Privacy and privacy impact assessment, Trust, Operating system security, Virtual machine Security,Security of virtualization, Security risks posed by shared images, Security risks	**10**

posed by a management OS, A trusted virtual machine monitor. Amazon web services: EC2 instances, Connecting clients to cloud instances through firewalls, Security rules for application and transport layer protocols in EC2, How to launch an EC2 Linux instance and connect to it, How to use S3 in java, Cloud-based simulation of a distributed trust algorithm, A trust management service, A cloud service for adaptive data streaming, Cloud based optimal FPGA synthesis .	
Module IV	
Data Intensive Computing, Map Reduce: What is Data-Intensive Computing?, Characterizing Data-Intensive Computations, Challenges Ahead, Historical Perspective, Technologies for Data-Intensive Computing, Storage Systems, Programming Platforms, Introducing the MapReduce Programming Model	**10**
Module V	
Cloud Platforms in Industry: Amazon Web Services, Google AppEngine,Microsoft Azure. **Cloud Applications**: Scientific Applications, Business and Consumer Applications.	**10**
Question paper pattern: • The question paper will have ten questions. • There will be two questions from each module. • Each question will have questions covering all the topics under a module. • The students will have to answer 5 full questions, selecting one full question from each module.	
Text Books: **1. Rajkumar Buyya, Christian Vecchiola, and Thamarai Selvi Mastering Cloud. Computing McGrawHill Education** **2. Dan C. Marinescu, Cloud Computing Theory and Practice, Morgan Kaufmann, Elsevier 2013.**	
E-books and Online course materials: • **https://www.javatpoint.com/cloud-computing-tutorial** • **https://www.tutorialspoint.com/cloud_computing/index.html** • **https://www.digimat.in/nptel/courses/video/106105167/L01.html (Video Lectures)**	

CO#	COURSE OUTCOMES
CO1	Students will be able to explain the fundamental concepts of cloud computing and analyze the virtualization concept.
CO2	Students will be able to identify and differentiate various types of cloud deployment models and cloud application platforms.
CO3	Students will be able to understand cloud security concepts, cloud Application Development procedure.
CO4	Students will be able to explain the concepts of Data intensive computing and Map Reduce programming
CO5	Students will be able to Identify industry cloud platforms and their applications.

CO-PO-PSO mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	3	2	-	-	2	-	-	-	-	-	2	2	3	-
CO2	3	3	2	-	-	2	-	-	-	-	-	2	2	3	-
CO3	3	3	2	-	-	2	-	-	-	-	-	2	2	3	-
CO4	3	3	2	-	-	2	-	-	-	-	-	2	2	3	-
CO5	3	3	2	-	-	2	-	-	-	-	-	2	2	3	-

SYSTEM SOFTWARE AND COMPILER DESIGN [As per Choice Based Credit System (CBCS)scheme] (Effective from the academic year 2023-2024) SEMESTER V			
Course Code	**21CS653**	**CIE Marks**	**50**
Number of Lecture Hours/Week	**04**	**SEE Marks**	**50**
Total Number of Lecture Hours	**50**	**Exam Hours**	**03**
Credits – 04			
Course Objectives:			
1. Define System Software. 2. Familiarize with source file, object file and executable file structures and libraries 3. Describe the front-end and back-end phases of compiler and their importance to students			
Modules			**Hours**
Module I			
Introduction to System Software, Machine Architecture of SIC and SIC/XE. **Assemblers:** Basic assembler functions, machine-dependent assembler features, machine-independent assembler features, and assembler design options.			**10**
Module II			
Introduction: Language Processors, The structure of a compiler, The evaluation of programming languages, The science of building compilers, Applications of compiler technology. **Lexical Analysis:** The role of lexical analyzer, Input buffering, Specifications of token, recognition of tokens.			**10**
Module III			
Syntax Analysis: Introduction, Role of Parser, Context Free Grammar, Top Down Parsers: Recursive Decent Parsing, LL(1) Grammar, Bottom-Up Parsers: Handle Pruning, Shift Reduce Parsing, LR(0), SLR-Parsing.			**10**
Module IV			
Lex and Yacc –The Simplest Lex Program, Grammars, Parser-Lexer Communication, A YACC Parser, The Rules Section, Running LEX and YACC, LEX and Hand-Written Lexers, Using LEX - Regular Expression, Examples of Regular Expressions, A Word Counting Program, Using YACC – Grammars, Recursive Rules, Shift/Reduce Parsing, What YACC Cannot Parse, A YACC Parser - The Definition Section, The Rules Section, The LEXER, Compiling and Running a Simple Parser, Arithmetic Expressions and Ambiguity.			**10**

Module V	
Syntax Directed Translation, Intermediate code generation, Code generation	**10**
Question paper pattern: The question paper will have ten questions. There will be 2 questions from each module. Each question will have questions covering all the topics under a module. The students will have to answer 5 full questions, selecting one full question from each module.	
Text Books: 1. System Software by Leland. L. Beck, D Manjula, 3rd edition, 2012 2. Alfred V Aho, Monica S. Lam, Ravi Sethi, Jeffrey D. Ullman , Compilers-Principles, Techniques and 3. Tools, Pearson, 2nd edition, 2007 4. Doug Brown, John Levine, Tony Mason, lex &yacc, O'Reilly Media, October 2012.	
Reference Books: 1. Systems programming – Srimanta Pal , Oxford university press, 2016 2. System programming and Compiler Design, K C Louden, Cengage Learning 3. System software and operating system by D. M. Dhamdhere TMG 4. Compiler Design, K Muneeswaran, Oxford University Press 2013.	
E-books and Online course materials **1.http://sit.ac.in/html/component/csedept/csecoursematerial/SSCDNotes.pdf**	
Online Courses and Video Lectures **1.https://onlinecourses.nptel.ac.in/noc21_cs07/preview** **2.https://www.youtube.com/playlist?list=PL1A5A6AE8AFC187B7**	

CO#	COURSE OUTCOMES
CO1	Understand and apply the fundamental concepts of system software, and the core functions of assemblers and loaders.
CO2	Analyze, design, and implement lexical analyzers, evaluate programming languages, and understand the structure and functioning of compilers.
CO3	Develop Top Down and Bottom-Up Parser
CO4	Design and implement lexical analyzers and parsers using Lex and YACC
CO5	Apply syntax-directed translation methods, generate intermediate code, and implement code generation techniques

CO-PO-PSO mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	3	2	-	-	-	-	-	-	-	-	-	2	3	-

CO2	3	3	2	-	-	-	-	-	-	-	-	-	2	3	-
CO3	3	3	2	-	-	-	-	-	-	-	-	-	2	3	-
CO4	3	3	2	-	-	-	-	-	-	-	-	-	2	3	-
CO5	3	3	2	-	-	-	-	-	-	-	-	-	2	3	-

FULL STACK WEB DEVELOPMENT LAB
[As per Choice Based Credit System (CBCS)scheme]
(Effective from the academic year 2023-2024)
SEMESTER – VI

Course Code	**21CSL66**	**CIE Marks**	**50**
Number of Lecture Hours/Week	**02**	**SEE Marks**	**50**
Total Number of Lecture Hours	**30**	**Exam Hours**	**03**

Credits– 01

Course Objectives:

1. Explain the use of learning full stack web development.
2. Make use of rapid application development in the design of responsive web pages.
3. Illustrate Models, Views and Templates with their connectivity in Django for full stack web development.
4. Demonstrate the use of state management and admin interfaces automation in Django.
5. Design and implement Django apps containing dynamic pages with SQL databases.

List of Experiments:

1. Installation of Python, Django and Visual Studio code editors can be demonstrated.
2. Creation of virtual environment, Django project and App should be demonstrated
3. Develop a Django app that displays current date and time in server
4. Develop a Django app that displays date and time four hours ahead and four hours before as an offset of current date and time in server
5. Develop a simple Django app that displays an unordered list of fruits and ordered list of selected students for an event
6. Develop a layout.html with a suitable header (containing navigation menu) and footer with copyright and developer information. Inherit this layout.html and create 3 additional pages: contact us, About Us and Home page of any website.
7. Develop a Django app that performs student registration to a course. It should also display list of students registered for any selected course. Create students and course as models with enrolment as *ManyToMany* field.
8. For student and course models created in Lab experiment for Module2, register admin interfaces, perform migrations and illustrate data entry through admin forms.
9. Develop a Model form for student that contains his topic chosen for project, languages used and duration with a model called project
10. For students enrolment developed in Module 2, create a generic class view which displays list of students and detail view that displays student details for any selected student in the list. **.**
11. Develop example Django app that performs CSV and PDF generation for any models created in previous laboratory component.
12. Develop a registration page for student enrolment as done in Module 2 but without page refresh using AJAX.
13. Write a program to create a voting application using React JS.
14. Create a web application to manage the TO-DO list of users, where users can login and manage their to-do items.

Conduct of Practical Examination:

• Experiment distribution

a) For laboratories having only one part: Students are allowed to pick one experiment from the lot with equal opportunity.

b) For laboratories having PART A and PART B: Students are allowed to pick one experiment from PART A and one experiment from PART B, with equal opportunity.

• Change of experiment is allowed only once and marks allotted for procedure to be made zero of the changed part only.

• Marks Distribution

SEE are mentioned here, writeup-15%, Conduction procedure and result in -70%, Viva-voce 15% of maximum marks. SEE for practical shall be evaluated for 50 marks

CO#	COURSE OUTCOMES
CO1	Demonstrate theoretical knowledge of Full Stack Web Development by conducting a series of hands-on experiments.
CO2	Develop a Program using Python/Django.
CO3	Debug and troubleshoot issues effectively.
CO4	Analyze the data and interpret the results.
CO5	Prepare a well-organized laboratory report.

CO-PO-PSO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	2	3	3	-	-	2	2	-	-	-	-	-	2	-	3
CO2	2	3	3	-	-	2	2	-	-	-	-	-	2	-	3
CO3	2	3	3	-	-	2	2	-	-	-	-	-	2	-	3
CO4	2	3	3	-	-	2	2	-	-	-	-	-	2	-	3
CO5	2	3	3	-	-	2	2	-	-	-	-	-	2	-	3

ARTIFICIAL INTELLIGENCE & MACHINE LEARNING LAB
[As per Choice Based Credit System (CBCS)scheme]
(Effective from the academic year 2023-2024)
SEMESTER – VI

Course Code	**21CSL67**	**CIE Marks**	**50**
Number of Lecture Hours/Week	**02**	**SEE Marks**	**50**
Total Number of Lecture Hours	**30**	**Exam Hours**	**03**

Credits – 01

Course Objectives:

1. Implement and evaluate AI and ML algorithms in Python programming language
2. Able to understand the ANN and back propagation algorithms by experimenting on datasets
3. Able to understand the implementation of plotting graphs by importing certain packages

List of Experiments

1.Implement and Demonstrate Depth First Search Algorithm for Water Jug Problem
2.Implement and Demonstrate the Travelling Salesman Problem
3. Write a Program to Implement Alpha-Beta Pruning using Python
4. Write a program to implement 8 puzzle problem
5. Implement and Demonstrate FIND-S Algorithm for finding the most Specific Hypothesis based on a given set of training samples. Read the training data from a .CSV file.
6. For a given set of training data examples stored in a .CSV file, implement and demonstrate the Candidate-Elimination algorithm to output a description of the set of all hypotheses consistent with the training examples.
7. Build an Artificial Neural Network by implementing the Backpropagation algorithm and test the same using appropriate data sets.
8. Apply EM algorithm to cluster a set of data stored in a .CSV file. Use the same data set for clustering using k-Means algorithm. Compare the results of these two algorithms and comment on the quality of clustering. You can add Java/Python ML library classes/API in the program.
9. Write a program to implement k-Nearest Neighbor algorithm to classify the iris data set. Print both correct and wrong predictions. Java/Python ML library classes can be used for this problem.
10. Implement the non-parametric Locally Weighted Regression algorithm in order to fit data points. Select appropriate data set for your experiment and draw graphs

Conduct of Practical Examination:

• Experiment distribution
   a) For laboratories having only one part: Students are allowed to pick one experiment from the lot with equal opportunity.
   b) For laboratories having PART A and PART B: Students are allowed to pick one experiment from PART A and one experiment from PART B, with equal opportunity.

• Change of experiment is allowed only once and marks allotted for procedure to be made zero of the changed part only.

• Marks Distribution

SEE are mentioned here, writeup-15%, Conduction procedure and result in -70%, Viva-voce 15% of maximum marks. SEE for practical shall be evaluated for 5 0 marks

CO#	COURSE OUTCOMES
CO1	Apply theoretical knowledge of Artificial Intelligence and Machine Learning by designing, conducting, and analyzing a series of experiments
CO2	Develop a program using python and essential machine learning libraries in spyder/jupyter
CO3	Diagnose, debug, and troubleshoot common issues in AI and ML workflows
CO4	Analyze the data and interpret the results.
CO5	Prepare a well-organized laboratory report.

CO-PO-PSO Mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	2	3	3	-	-	2	2	-	-	-	-	-	2	-	3
CO2	2	3	3	-	-	2	2	-	-	-	-	-	2	-	3
CO3	2	3	3	-	-	2	2	-	-	-	-	-	2	-	3
CO4	2	3	3	-	-	2	2	-	-	-	-	-	2	-	3
CO5	2	3	3	-	-	2	2	-	-	-	-	-	2	-	3

PROJECT-VI [As per Choice Based Credit System (CBCS) scheme] (Effective from the academic year 2023-24) SEMESTER – VI			
Course Code	**21PRJ68**	**CIE Marks**	**50**
Number of Lecture Hours/Week	**02**	**SEE Marks**	**50**
Total Number of Lecture Hours	**30**	**Exam Hours**	**03**
Credits – 01			

Course Objectives:

1. Identify real-world problems across programming, databases, and networking domains and understand their business and technical implications.
2. Apply systematic methodologies to design, implement, and optimize solutions.
3. Resolve technical challenges through debugging, research, and collaboration.
4. Take responsibility for specific roles in a team and collaborate effectively to achieve project goals.
5. Present project progress and findings clearly and confidently to both technical and non-technical audiences.
6. Document the entire project in a structured, professional laboratory report.

Project Guidelines:

- Project work shall preferably be batch wise.
- Evaluation is based on concept clarity, system design, implementation, testing, presentation, and documentation quality, with a focus on proper coding standards, teamwork, and effective communication.
- Viva-voce examination in project work shall be conducted batch-wise.
- Minimum requirement of CIE marks for Project work shall be 50% of the maximum marks.
- Students failing to secure a minimum of 50% of the CIE marks in Project work shall not be eligible for the SEE Project examination.
- For a pass in a Project/Viva-voce examination, a student shall secure a minimum of 40% of the maximum marks prescribed.

CO#	COURSE OUTCOMES
CO1	Identify the topic from various domains (example programming databases, networking) to real world problems.
CO2	Develop methodology for the problem.
CO3	Resolve issues that arise during the project .
CO4	Learn to assign and accept roles and responsibilities within a team and write a good technical reports.
CO5	Exhibit skills in presenting their project findings & progress orally

CO-PO-PSO mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	3	3	-	2	1	-	-	2	2	2	-	-	-	3
CO2	3	3	3	-	2	1	-	-	2	2	2	-	-	-	3
CO3	3	3	3	-	2	1	-	-	2	2	2	-	-	-	3
CO4	3	3	3	-	2	1	-	-	2	2	2	-	-	-	3
CO5	3	3	3	-	2	1	-	-	2	2	2	-	-	-	3

RESEARCH METHODOLOGY& INTELLECTUAL PROPERTY RIGHTS [As per Choice Based Credit System (CBCS)scheme] (Effective from the academic year 2023-2024) SEMESTER – VI			
Course Code	**21AU69**	**CIE Marks**	**50**
Number of Lecture Hours/Week	**01**	**SEE Marks**	**50**
Total Number of Lecture Hours	**20**	**Exam Hours**	**03**
Credits –01			
Course Objectives:			
1. To Understand the knowledge on basics of research and its types. 2. To Learn the concept of Literature Review, Technical Reading, Attributions and Citations. **3.** To learn Ethics in Engineering Research. 4. To Discuss the concepts of Intellectual Property Rights in engineering			
Modules			**Hours**
Module I			
Introduction: Meaning of Research, Objectives of Engineering Research, and Motivation in Engineering Research, Types of Engineering Research, Finding and Solving a Worthwhile Problem. Ethics in Engineering Research, Ethics in Engineering Research Practice, Types of Research Misconduct, Ethical Issues Related to Authorship.			**04**
Module II			
Literature Review and Technical Reading, New and Existing Knowledge, Analysis and Synthesis of Prior Art Bibliographic Databases, Web of Science, Google and Google Scholar, Effective Search: The Way Forward Introduction to Technical Reading Conceptualizing Research, Critical and Creative Reading, Taking Notes While Reading, Reading Mathematics and Algorithms, Reading a Datasheet. Attributions and Citations: Giving Credit Wherever Due, Citations: Functions and Attributes, Impact of Title and Keywords on Citations, Knowledge Flow through Citation, Citing Datasets, Styles for Citations, Acknowledgments and Attributions, What Should Be Acknowledged, Acknowledgments in, Books Dissertations, Dedication or Acknowledgments.			**04**
Module III			
Introduction To Intellectual Property: Role of IP in the Economic and Cultural Development of the Society, IP Governance, IP as a Global Indicator of Innovation, Origin of IP History of IP in India. Major Amendments in IP Laws and Acts in India. Patents: Conditions for Obtaining a Patent Protection, To Patent or Not to Patent an Invention. Rights Associated with Patents. Enforcement of Patent Rights. Inventions Eligible for Patenting. Non-Patentable Matters. Patent Infringements. Avoid Public Disclosure of an Invention before Patenting. Process of Patenting. Prior Art Search. Choice of Application to be Filed. Patent Application Forms. Jurisdiction of Filing Patent Application. Publication. Pre-grant Opposition. Examination. Grant of a Patent. Validity of Patent Protection. Post-grant Opposition. Commercialization of a Patent. Need for a Patent Attorney/Agent. Can a Worldwide Patent be Obtained. Do I Need First to			**04**

File a Patent in India. Patent Related Forms. Fee Structure. Types of Patent Applications. Commonly Used Terms in Patenting. National Bodies Dealing with Patent Affairs. Utility Models. Process of Patenting. Prior Art Search. Choice of Application to be Filed. Patent Application Forms. Jurisdiction of Filing Patent Application. Publication. Pre-grant Opposition. Examination. Grant of a Patent. Validity of Patent Protection. Post-grant Opposition. Commercialization of a Patent. Need for a Patent Attorney/Agent. Can a Worldwide Patent be Obtained. Do I Need First to File a Patent in India. Patent Related Forms. Fee Structure. Types of Patent Applications. Commonly Used Terms in Patenting. National Bodies Dealing with Patent Affairs. Utility Models.	
Module IV	
Copyrights and Related Rights: Classes of Copyrights. Criteria for Copyright. Ownership of Copyright. Copyrights of the Author. Copyright Infringements. Copyright Infringement is a Criminal Offence. Copyright Infringement is a Cognizable Offence. Fair Use Doctrine. Copyrights and Internet. Non-Copyright Work. Copyright Registration. Judicial Powers of the Registrar of Copyrights. Fee Structure. Copyright Symbol. Validity of Copyright. Copyright Profile of India. Copyright and the word 'Publish'. Transfer of Copyrights to a Publisher. Copyrights and the Word 'Adaptation'. Copyrights and the Word 'Indian Work'. Joint Authorship. Copyright Society. Copyright Board. Copyright Enforcement Advisory Council (CEAC). International Copyright Agreements, Conventions and Treaties. Interesting Copyrights Cases. Trademarks: Eligibility Criteria. Who Can Apply for a Trademark. Acts and Laws. Designation of Trademark Symbols. Classification of Trademarks. Registration of a Trademark is Not Compulsory. Validity of Trademark. Types of Trademark Registered in India. Trademark Registry. Process for Trademarks Registration. Prior Art Search. Famous Case Law: Coca-Cola Company vs. Bisleri International Pvt. Ltd.	**04**
Module V	
Industrial Designs: Eligibility Criteria. Acts and Laws to Govern Industrial Designs. Design Rights. Enforcement of Design Rights. Non-Protectable Industrial Designs India. Protection Term. Procedure for Registration of Industrial Designs. Prior Art Search. Application for Registration. Duration of the Registration of a Design. Importance of Design Registration. Cancellation of the Registered Design. Application Forms. Classification of Industrial Designs. Designs Registration Trend in India. International Treaties. Famous Case Law: Apple Inc. vs. Samsung Electronics Co. Geographical Indications: Acts, Laws and Rules Pertaining to GI. Ownership of GI. Rights Granted to the Holders. Registered GI in India. Identification of Registered GI. Classes of GI. Non-Registerable GI. Protection of GI. Collective or Certification Marks. Enforcement of GI Rights. Procedure for GI Registration Documents Required for GI Registration. GI Ecosystem in India. Case Studies on Patents. Case study of Curcuma (Turmeric) Patent, Case study of Neem Patent, Case study of Basmati patent. IP Organizations In India. Schemes and Programmes	**04**

Question paper pattern:

The question paper will have ten questions each question is set for 20 marks.
There will be 2 full questions from each module each of the questions under a module (with a maximum of 3 sub-questions), **should have a mix of topics under that module**.
The students have to answer 5 full questions, selecting one full question from each module.
Marks scored by the student shall be proportionally scaled down to 50 Marks.

Text Books:

Doing Data Science, Cathy O'Neil and Rachel Schutt, O'Reilly Media, Inc O'Reilly Media, Inc, 2013
Data Visualization workshop, Tim Grobmann and Mario Dobler, Packt Publishing, ISBN 9781800568112

Reference Books:

Mining of Massive Datasets, Anand Rajaraman and Jeffrey D. Ullman, Cambridge University Press, 2010
Data Science from Scratch, Joel Grus, Shroff Publisher /O'Reilly Publisher Media
A handbook for data driven design by Andy krik

E-books and Online course materials:

https://nptel.ac.in/courses/106/105/106105077/
ttps://www.oreilly.com/library/view/doing-data-science/9781449363871/toc01.html
http://book.visualisingdata.com/
https://matplotlib.org/
https://docs.python.org/3/tutorial/
https://www.tableau.com/

CO#	COURSE OUTCOMES
CO1	To introduce engineering research, develop problem-solving skills, and emphasize research ethics.
CO2	To explore literature review, technical reading, and proper citation practices in research.
CO3	To understand IP's role in development and the patenting process, including rights, protection, and commercialization.
CO4	To examine copyrights and trademarks, including ownership, registration, infringement, and related laws.
CO5	To study industrial designs and geographical indications, including registration, rights, protection, and related laws.

CO-PO-PSO mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	3	2	-	-	-	-	-	-	-	-	-	2	3	-
CO2	3	3	2	-	-	-	-	-	-	-	-	-	2	3	-
CO3	3	3	2	-	-	-	-	-	-	-	-	-	2	3	-
CO4	3	3	2	-	-	-	-	-	-	-	-	-	2	3	-

CO5	3	3	2	-	-	-	-	-	-	-	-	-	2	3	-

AWS Framework
[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2023-2024)
SEMESTER – VI

Course Code	**21ACS610**	**CIE Marks**	50
Number of Lecture Hours/Week	**02**	**SEE Marks**	50
Total Number of LectureHours	**30**	**Exam Hours**	03

CREDITS – 01

Course Objectives: This course will enable students
Designed to give you a comprehensive understanding of the foundational services offered by AWS including compute, storage, networking, database, and identity & access management (IAM).

PART A

Illustration of the following services.

- Application Auto Scaling
- Amazon Aurora
- AWS Cloud9
- Amazon CloudFormation
- Amazon CloudFront
- AWS Cloud Shell
- AWS CloudTrail
- Amazon CloudWatch
- AWS Code Commit
- Amazon Cognito
- Amazon Comprehend
- AWS Deep Racer
- Amazon DynamoDB
- Amazon EC2 Auto Scaling
- AWS Elastic Beanstalk
- Amazon Elastic Block Store (EBS)
- Amazon Elastic Compute Cloud (EC2)
- Amazon Elastic Container Registry (ECR)
- Amazon Elastic File System (EFS)
- Amazon Elastic Inference
- Elastic Load Balancing
- Amazon Event Bridge
- Amazon Forecast
- AWS Glue
- AWS Glue Data Brew
- AWS Identity and Access Management (IAM)
- AWS Key Management Service (KMS)
- AWS Lambda
- Amazon Lex
- Amazon LightSail
- Amazon Marketplace Subscriptions (Amazon ML)
- Amazon Polly
- Amazon Recognition
- Amazon Relational Database Service (RDS)
- AWS Resource Groups & Tag Editor
- AWS Robemakers
- Amazon Sage Maker
- AWS Secrets Manager
- AWS Security Token Service (STS)
- AWS Service Catalog
- Amazon Simple Notification Service (SNS)
- Amazon Simple Queue Service (SQS)
- Amazon Simple Storage Service (S3)
- Amazon Simple Storage Service Glacier (S3 Glacier)
- AWS Step Functions
- AWS Systems Manager (SSM)
- Amazon Extract
- Amazon Translate
- AWS Trusted Advisor
- Amazon Virtual Private Cloud (Amazon VPC)
- AWS Well-Architected Tool

PART B

1. Introduction to AWS IAM
2. Build Your VPC and Launch a Web Server
3. Introduction to Amazon EC2
4. Working with Amazon EBS
5. Build Your DB Server and Interact with Your DB Using an App.

Conduct of Practical Examination:

• Experiment distribution

a) For laboratories having only one part: Students are allowed to pick one experiment from the lot with equal opportunity.

b) For laboratories having PART A and PART B: Students are allowed to pick one experiment from PART A and one experiment from PART B, with equal opportunity.

• Change of experiment is allowed only once and marks allotted for procedure to be made zero of the changed part only.

• Marks Distribution

SEE are mentioned here, writeup-15%, Conduction procedure and result in -70%, Viva-voce 15% of maximum marks. SEE for practical shall be evaluated for 50 marks

CO#	COURSE OUTCOMES
CO1	Demonstrate various AWS services
CO2	Implement different scenario of real-world problem using AWS service.
CO3	Debug and troubleshoot issues effectively.
CO4	Analyze the data and interpret the results.
CO5	Prepare a well-organized laboratory report.

CO-PO-PSO mapping:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	3	2	-	-	-	-	-	-	-	-	-	2	-	3
CO2	3	3	2	-	-	-	-	-	-	-	-	-	2	-	3
CO3	3	3	2	-	-	-	-	-	-	-	-	-	2	-	3
CO4	3	3	2	-	-	-	-	-	-	-	-	-	2	-	3
CO5	3	3	2	-	-	-	-	-	-	-	-	-	2	-	3