Centenary Celebrated Sharnbasveshwar Vidya Vardhaka Sangha's

ಶರಣಬಸವ ವಿಶ್ವವಿದ್ಯಾಲಯ

Sharnbasva University

(Under the agesis of Cantenary Celebrated Sharnbasvshwar Vidya Vardhaka Sangha)
Kalaburagi - 585 103 Karnataka, India

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 2024-25)

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 2022-23
[As Per NEP, Outcome Based Education (OBE) and Choice Based Credit System (CBCS) Scheme]
(Effective from the academic year 2022-23)

Programme:B.Tech:Computer Science and Engineering

VSEMESTER

Sl. No.	CourseCode		CourseTitle	TeachingDepartment	Teaching Hours/week			Examination				Credits
					Theory Lecture	Tutorial	Practical/ Drawing	Duration in Hours	CIE Marks	SEE Marks	Total Marks	
					L	T	P					
1	HSS	22HSM51	Management and Entrepreneurship Development	CSE	3			3	50	50	100	03
2	PCC	22CS52	System Software and Compiler Design	CSE	2	1		3	50	50	100	03
3	PCC	22CS53	Computer Networks	CSE	3			3	50	50	100	03
4	PEC	22CS54X	Professional Elective Course -I	CSE	3			3	50	50	100	03
5	OEC	22XX55X	Open Elective – I	CSE	4			3	50	50	100	04
6	PCC	22CSL56	System Software and Compiler Design Lab	CSE			2	3	50	50	100	01
7	PCC	22CSL57	Computer Networks Lab	CSE			2	3	50	50	100	01
8	PEC	22CSL58X	Professional Elective Course Lab	CSE			2	3	50	50	100	01
9	PW	22PRJ59	Project-V	Respective Branch			2	3	50	50	100	01
10	AEC	22AXX510 X	Ability Enhancement Course-V	Respective Branch			2	3	50	50	100	01
Total					**15**	**1**	**10**	**30**	**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): A Batch of 4 students (Same Branch or Different Branches with a Guide, May undertake one project.

Professional Elective Course-I			
Coursecodeunder22CS54X	**Course Title**	**Course code under 22CSL58X**	**Course Title**
22CS541	Unix System Programming	**22CSL581**	Systems Programming Lab
22CS542	Computer Graphics and Fundamentals of Image Processing	**22CSL582**	Computer Graphics and Fundamental of Image Processing Lab
22CS543	Cloud Computing	**22CSL583**	Cloud Computing Lab
22CS544	Advanced Java Programming	**22CSL584**	Advanced Java Programming Lab
Open Elective Course – I			
22CS551	Introduction to Data Structures		
22CS552	Fundamentals of Database Management System		

Ability Enhancement Course-V	
Coursecodeunder22AEC510X	**Course Title**
22ACS510A	Angular and Node JS
22AAD510B	Microsoft Power BI

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 2022-23

[As Per NEP, Outcome Based Education (OBE) and Choice Based Credit System (CBCS) Scheme]

(Effective from the academic year 2022-23)

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	22CS61	Software Engineering	CSE	3			3	50	50	100	03
2	PCC	22CS62	Artificial Intelligence and Machine Learning	CSE	2	1		3	50	50	100	03
3	PEC	22CS63X	Professional Elective Course –II	CSE	3			3	50	50	100	03
4	PEC	22CS64X	Professional Elective Course –III	CSE	3			3	50	50	100	03
5	OEC	22XX65X	Open Elective Course –II	CSE	4			3	50	50	100	04
6	PCC	22CSL66	Software Engineering Lab	CSE			2	3	50	50	100	01
7	PCC	22CSL67	Artificial Intelligence and Machine Learning Lab	CSE			2	3	50	50	100	01
8	PEC	22CSL68X	Professional Elective Course Lab-II Lab	CSE			2	3	50	50	100	01
9	PW	22PRJ69	Project-VI	CSE			2	3	50	50	100	01
10	HSS	22HSM610A	Research Methodology and Intellectual Property Rights	Humanities	1			2	50	50	100	01
11	AEC	22AXX611X	Ability Enhancement Course-VI	CSE			2	3	50	50	100	01
Total					**16**	**1**	**10**	**32**	**550**	**550**	**1100**	**22**

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): A Batch of 4 students (Same Branch or Different Branches with a Guide, May undertake one project.			
Professional Elective Course-II			
Course code under 22XX63X	**Course Title**	**Course code under 22XXL68X**	**Course Title**
22CS631	Full Stack Development	22CSL681	Full Stack Development Lab
22CS632	Cryptography and Network Security	22CSL682	Cryptography and Network Security Lab
22AD633	Natural Language Processing	22ADL683	Natural Language Processing Lab
22CS634	Mobile Application Development	22CSL684	Mobile Application Development Lab
Professional Elective Course-III			
Course code under 22EC64X	**Course Title**		
22CS641	Image Processing		
22CS642	Blockchain Technology		
22CS643	Object Oriented Modelling and Design		
22CS644	Cognitive Science		
Open Elective Course –II			
22CS651	OOPS with C++		
22CS652	Java Programming		
Ability Enhancement Course-VI			
22AAD611A	Tableau		
22ACS611B	Devops		
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..			

MANAGEMENT AND ENTREPRENEURSHIP DEVELOPMENT
[As per NEP Choice Based Credit System (CBCS)scheme]
(Effective from the academic year 2024-2025)
SEMESTER-V

Course Code	**22HSM51**	**CIE Marks**	**50**
Number Lecture Hour/Week	**03**	**SEE Marks**	**50**
Number of Lecture Hours	**40**	**Exam Hours**	**03**

CREDITS-03

Course Objectives The objectives of the course is to enable students to:

1. Understand basic skills of Management.
2. Understand the need for Entrepreneurs and their skills.
3. Identify the Management functions and Social responsibilities.
4. Distinguish between management and administration.
5. Understand Project identification and Selection.

Modules	**Hours**
Module `-I	
Management: Introduction-Meaning-Nature and characteristics of management, Scope and Functional areas of management- Management as art of science, art or profession-Management & Administration-Roles of Management, Levels of Management, Development of Management Thought-Early management approaches-Modern management approaches. **Planning**: Nature, Importance and purpose of planning process objectives-types of plans (meaning only)-decision making, Importance of planning-steps in planning & planning premise- Hierarchy of plans.	08
Module -II	
Organizing and Staffing: Organization-Meaning, Characteristics, Process of Organizing, Principles of Organizing, Span of Management (meaning and importance only), Departmentalization, Committees–Meaning, Types of Committees; Centralization Vs Decentralization of Authority and Responsibility; **Staffing**-Need and Importance, Recruitment and Selection Process. **Directing:** Meaning and Requirements of Effective Direction, Giving Orders; Motivation-Nature of Motivation, Motivation Theories (Maslow's Need-Hierarchy Theory and Herzberg's Two Factor Theory); Communication – Meaning, Importance and Purposes of Communication; Leadership-Meaning, Characteristics, Behavioral Approach of Leadership.	08
Module -III	
Coordination: Coordination-Meaning, Types, Techniques of Coordination; **Controlling** – Meaning, Need for Control System, Benefits of Control, Essentials of Effective Control System, Steps in Control Process. **Authority delegation:** Meaning, advantage of effective delegation, barriers to effective delegation, guidelines for effective delegation. Decentralization: Decentralization of authority meaning, distinction between delegation and decentralization, the trade-off of centralization and decentralization.	08
Module -4	
Entrepreneurship: Definition of Entrepreneur, Importance of Entrepreneurship, concepts of Entrepreneurship, Characteristics of successful Entrepreneur, Classification of Entrepreneurs, Myths of Entrepreneurship, Entrepreneurial Development models, Entrepreneurial development cycle. **Modern Small Business Enterprises:** Role of Small Scale Industries, Impact of Globalization and WTO on SSIs, Concepts and definitions of SSI Enterprises,	08

Government policy and development of the Small Scale sector in India, Growth and Performance of Small Scale Industries in India, Sickness in SSI sector, Problems for Small Scale Industries, Ancillary Industry and Tiny Industry (Definition only) .	
Module -5	
Projects Management: A Project. Search for a Business idea: Introduction, Choosing an Idea, Selection of product, The Adoption process, Product Innovation, Product Planning and Development Strategy, Product Planning and Development Process. Concepts of Projects and Classification: Introduction, Meaning of Projects, Characteristics of a Project, Project Levels, Project Classification, Aspects of a Project, The project Cycle, Features and Phases of Project management, Project Management Processes. Project Identification: Feasibility Report, Project Feasibility Analysis. Project Formulation: Meaning, Steps in Project formulation, Sequential Stages of Project Formulation, Project Evaluation. **Project Design and Network Analysis:** Introduction, Importance of Network Analysis, Origin of PERT and CPM, Network, Network Techniques, Need for Network Techniques, Steps in PERT, CPM, Advantages, Limitations and Differences.	08

Question paper pattern

- The question paper will have TEN questions
- Each full question carries 20 marks
- There will be two full questions (with a maximum of THREE sub questions) from each module
- Each full question will have sub questions covering all topics under a module
- The students will have to answer 5 full questions, selecting one full question from each module

Textbooks:

1.Principles of Management – P.C Tripathi, P.N Reddy, McGraw Hill Education, 6th Edition, 2017. ISBN-13:978-93-5260-535-4.

2.Entrepreneurship Development Small Business Enterprises- Poornima M Charantimath, Pearson Education 2008, ISBN 978-81-7758-260-4.

3.Dynamics of Entrepreneurial Development and Management by Vasant Desai. HPH 2007, ISBN: 978-81-8488-801-2.

4.Robert D. Hisrich, Mathew J. Manimala, Michael P Peters and Dean A. Shepherd, "Entrepreneurship", 8th Edition, Tata Mc-graw Hill Publishing Co.ltd.-new Delhi, 2012

Reference Books:

1.Essentials of Management: An International, Innovation and Leadership perspective by Harold Koontz, Heinz Weihrich McGraw Hill Education, 10th Edition 2016. ISBN- 978-93-392-2286-4.

COURSE OUTCOMES (COS):

CO1	Understand the fundamental concepts of Management and Entrepreneurship and opportunities to set up a business.
CO2	Identify the required organizing committee and staffing.
CO3	Understand the techniques of coordinates and authority delegates.
CO4	Design and develop project management and network analysis.
CO5	Develop a entrepreneurial mindset and leadership skills to drive organizational success

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

SYSTEM SOFTWARE AND COMPILER DESIGN
[As per NEP Choice Based Credit System (CBCS)scheme]
(Effective from the academic year 2024-2025)
SEMESTER – V

Course Code	**22CS52**	**CIE Marks**	**50**
Number Lecture Hour/Week	**03**	**SEE Marks**	**50**
Number of Lecture Hours	**40**	**Exam Hours**	**03**

CREDITS-03

Course Objectives: This course will enable students to:

- Define System Software.
- Familiarize with source file, object file and executable file structures and libraries
- To Teach concepts of language translation and phases of compiler design
- To demonstrate the common forms of parsers
- To demonstrate intermediate code using technique of syntax-directed translation

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, assembler design options. Basic Loader Functions **Text book 1: Chapter 1: 1.1,1.2,1.3.1,1.3.2, Chapter2 : 2.1 to 2.4, Chapter 3 ,3.1**	**08**
Module -II	
Introduction: Language Processors, The structure of a compiler, The evaluation of programming languages, The science of building compiler, Applications of compilertechnology. **Lexical Analysis:** The role of lexical analyzer, Input buffering, Specifications of token,recognition of tokens. **Text book 2: Chapter 1 1.1-1.5 Chapter 3: 3.1 – 3.4**	**08**
Module -III	
Syntax Analysis: Introduction, Context Free Grammars, Writing a grammar, Top-DownParsers, Bottom-Up Parsers **Text book 2: Chapter 4 4.1, 4.2 4.3 4.4 4.5**	**08**
Module-IV	
Lex and Yacc –The Simplest Lex Program, Grammars, Parser-Lexer Communication, AYACC Parser, The Rules Section, Running LEX and YACC, LEX and Hand- Written Lexers, Using LEX - Regular Expression, Examples of Regular Expressions, A WordCounting 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. **Text book 3: Chapter 1,2 and 3**	**08**

Module-V	
Syntax Directed Translation, Intermediate code generation, Code generation **Text book 2: Chapter 5.1, 5.2,** **5.3, 6.1, 6.2, 8.1, 8.2RBT: L1, L2, L3**	**08**

Question paper pattern

- The question paper will have TEN questions
- Each full question carries 20 marks
- There will be two full questions (with a maximum of THREE sub questions) from each module
- Each full question will have sub questions covering all topics under a module
- The students will have to answer 5 full questions, selecting one full question from each module

Textbooks:

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 andTools, 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

COURSE OUTCOMES (COS):

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

COMPUTER NETWORKS
[As per Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2024-2025)
SEMESTER – V

Course Code	**22CS53**	**CIE Marks**	**50**
Number Lecture Hour/Week	**03**	**SEE Marks**	**50**
Number of Lecture Hours	**40**	**Exam Hours**	**03**

CREDITS-03

Course Objectives:

1. Explore basic concepts of data communication
2. Understand the working of Datalink Layer
3. Learn network layer services and IP versions.
4. Discuss transport layer services and understand UDP and TCP protocols.
5. Demonstrate the working of different Application layer protocols.

Modules	**Hours**
Module -I	
Introduction: Data Communications, Networks, Network Types, Networks Models: Protocol Layering, TCP/IP Protocol suite, The OSI model, Introduction to Physical Layer: Transmission media, Guided Media, Unguided Media: Wireless. Switching: Packet Switching and its types.	**08**
Module -II	
Data Link Layer: Error Detection and Correction: Introduction, Block Coding, Cyclic Codes. Data link control: DLC Services: Framing, Flow Control, Error Control, Connectionless and Connection Oriented, Data link layer protocols, High Level Data Link Control. Media Access Control: Random Access, Controlled Access. Check Sum and Point to Point Protocol	**08**
Module -III	
Network Layer: Network layer Services, Packet Switching, IPv4 Address, IPv4 Datagram, IPv6 Datagram, Introduction to Routing Algorithms, Unicast Routing Protocols: DVR, LSR, PVR, Unicast Routing protocols: RIP, OSPF, BGP, Multicasting Routing-MOSPF	**08**
Module-IV	
Transport Layer: Introduction, Transport-Layer Protocols: Introduction, User Datagram Protocol, Transmission Control Protocol: services, features, segments, TCP connections, flow control, Error control, Congestion control	**08**
Module-V	
Application Layer: Introduction, Client-Server Programming, Standard Client Server Protocols: World Wide Web and HTTP, FTP, Electronic Mail, Domain Name System (DNS), TELNET, Secure Shell (SSH)	**08**

Question paper pattern

- The question paper will have TEN questions
- Each full question carries 20 marks
- There will be two full questions (with a maximum of THREE sub questions) from each module
- Each full question will have sub questions covering all topics under a module.
- The students will have to answer 5 full questions, selecting one full question from each module

Text Books:

1.Data Communications and Networking, Behrouz A. Forouzan, Fourth Edition TMH.

Reference Books:

1. Computer Networks, Andrew S Tanenbaum, 6th Edition. Pearson Education.

2. Computer Networking: A Top-Down Approach Featuring the Internet. James F. Kurose & Keith W. Ross, 3rd Edition, Pearson Education

3. Data communications and Computer Networks, P.C Gupta, PHI.

4. An Engineering Approach to Computer Networks, S. Keshav, 2nd Edition, Pearson Education.

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.digimat.in/nptel/courses/video/106105183/L01.html
2. http://www.digimat.in/nptel/courses/video/106105081/L25.html
3. https://nptel.ac.in/courses/10610

Course Outcomes (COs):

CO1	Demonstrate basics of communication and computer networks
CO2	Apply the concepts of error detection, correction and datalink layer protocols to real world scenarios
CO3	Analyze the services provided by network Layer
CO4	Comprehend the functionalities of the transport layer
CO5	Develop and manage networked applications for different 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	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	**22CS541**	**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
- Each full question carries 20 marks
- There will be two full questions (with a maximum of THREE sub questions) from each module
- Each full question will have sub questions covering all topics under a module. The students will have to answer 5 full questions, selecting one full question from each module

Text Book

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

Course Outcomes (COs):

CO1	Understand UNIX architecture, POSIX standards, and file management concepts
CO2	Use UNIX file APIs and understand process environment and memory management.
CO3	Apply process control and manage job execution in UNIX.
CO4	Understand and apply signal handling and daemon process management in UNIX
CO5	Implement IPC methods and client-server communication in UNIX

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	-

Computer Graphics and Fundamentals of Image Processing
[As per NEP Choice Based Credit System (CBCS)scheme]
(Effective from the academic year 2024-2025)
SEMESTER – V

Course Code	**22CS542**	**CIE Marks**	**50**
Number Lecture Hour/Week	**03**	**SEE Marks**	**50**
Number of Lecture Hours	**40**	**Exam Hours**	**03**

CREDITS-03

Course Objectives:

- Overview of Computer Graphics along with its applications.
- Exploring 2D and 3D graphics mathematics along with OpenGL API‘s. CLO
- Use of Computer graphics principles for animation and design of GUI‘s .
- Introduction to Image processing and Open CV.
- 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, Bresenham‘s). Textbook 1: Chapter -1,2,3, 5(1 and 2 only)	**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 transformations function, 3D Geometric Transformations: Translation, rotation, scaling, composite 3D transformations, other 3D transformations, OpenGL geometric transformations functions. Textbook 1: Chapter -6, 8	**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. Textbook 1: Chapter -11, 18	**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. Text book 2: Chapter 3	**08**
Module-V	
Image Segmentation: Introduction, classification, detection of discontinuities, Edge detection (up to canny edge detection(included)). Text Book 2: Chapter 9: 9.1 to 9.4.4.4	**08**

Question paper pattern

- The question paper will have TEN questions
- Each full question carries 20 marks
- There will be two full questions (with a maximum of THREE sub questions) from each module
- Each full question will have sub questions covering all topics under a module.
- The students will have to answer 5 full questions, selecting one full question from each module

Textbooks

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

Web links and Video Lectures (e-Resources):

1. https://nptel.ac.in/courses/106/102/106102063/
2. https://nptel.ac.in/courses/106/103/106103224/
3. https://nptel.ac.in/courses/106/102/106102065/
4. https://www.tutorialspoint.com/opencv/ (Tutorial, Types of Images, Drawing Functions)
5. https://nptel.ac.in/courses/106/106/106106090/

Course Outcomes (COs):

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

Cloud Computing

[As per Choice Based Credit System(CBCS) scheme]
(Effective from the academic year 2024 -2025)
SEMESTER – V

Course Code	22CS543	**CIE Marks**	50
Number of Contact Hours/Week	03	**SEE Marks**	50
Total Number of Contact Hours	40	**Exam Hours**	03

CREDITS –03

Course Objectives: This course will enable students to:

- Describe the security and compliance measures of the AWS Cloud, including AWS Identity and Access Management (IAM)
- Create a virtual private cloud (VPC) by using Amazon Virtual Private Cloud (Amazon VPC)
- Demonstrate when to use Amazon Elastic Compute Cloud (Amazon EC2), AWS Lambda, and AWS Elastic Beanstalk
- Differentiate between Amazon Simple Storage Service (Amazon S3), Amazon Elastic Block Store (Amazon EBS), Amazon Elastic File System (Amazon EFS), and Amazon Simple Storage Service Glacier (Amazon S3 Glacier)
- Demonstrate when to use AWS database services, including Amazon Relational Database Service (Amazon RDS), Amazon DynamoDB, Amazon Redshift, and Amazon Aurora
- Explain the architectural principles of the AWS Cloud
- Explore key concepts related to Elastic Load Balancing, Amazon CloudWatch, and Amazon EC2Auto Scaling

Modules	**Hours**
Module – I	
Cloud Concepts Overview: Introduction to Cloud Computing, Advantages of the Cloud, Introduction to AWS, Moving to the AWS Cloud, **Cloud Economics and Billing**: Introduction Fundamentals of Pricing, Total Cost of Ownership, Simple Monthly Calculator, Delaware North Case Study, AWS Organizations, AWS Billing and Cost Management, Billing Dashboard	**08**
Module – II	
AWS Global Infrastructure Overview: Introduction, AWS Global Infrastructure, AWS Global Infrastructure, AWS Services and Service Categories, AWS Management Console Clickthrough. **Cloud Security:** Introduction, AWS Shared Responsibility Model, AWS Shared Responsibility Model, AWS IAM, AWS IAM Console Demonstration, Securing a New AWS Account, Introduction to AWS IAM, Securing Accounts, Securing Data, Working to Ensure Compliance	**08**
Module – III	
Networking and Content Delivery: Introduction, Networking Basics, Amazon VPC, VPC Networking, Label This Diagram, Amazon VPC Console Demonstration, VPC Security, Design a VPC Build a VPC and Launch a Web Server, Route 53, CloudFront. **Compute:** Introduction, Compute Services Overview, Amazon EC2 Part 1, Amazon EC2 Part 2, Amazon EC2 Part 3, Introduction to Amazon EC2, Amazon EC2 versus Managed Services, Amazon EC2 Part Console Demonstration, Amazon EC2 Cost Optimization, Container Services, Introduction to AWS Lambda, AWS Lambda, Introduction to AWS Elastic Beanstalk, AWS Elastic Beanstalk	**08**
Module – IV	

Storage: Introduction, AWS EBS, Amazon Elastic Block Store Console, Demonstration, Working with EBS, AWS S3, AWS S3 Console Demonstration, AWS EFS, AWS EFS Console Demonstration, AWS S3 Glacier, AWS S3 Glacier Console Demonstration, Storage Technology Selection**. Databases:** Introduction, Amazon RDS, Amazon RDS Console Demonstration, Build a Database Server, Amazon DynamoDB, Amazon DynamoDB Demonstration, Amazon Redshift, Amazon Aurora, Database Case Study	**08**
Module – V	
Cloud Architecture: Introduction, AWS Well-Architected Framework Design, Principles, AWS Well-Architected Framework Design, Principles, Operational Excellence, Security, Reliability Performance Efficiency, Cost Optimization, Reliability & High Availability, AWS Trusted Advisor, Interpret AWS Trusted Advisor Recommendations. **Automatic Scaling and Monitoring:** Introduction, Elastic Load Balancing, Elastic Load Balancing, Amazon CloudWatch, Amazon CloudWatch, Amazon EC2 Auto Scaling, Scale & Load Balance your Architecture	**08**

Question paper pattern

- The question paper will have TEN questions
- Each full question carries 20 marks
- There will be two full questions (with a maximum of THREE sub questions) from each module
- Each full question will have sub questions covering all topics under a module. The students will have to answer 5 full questions, selecting one full question from each module

Textbook and Reference Books

1.Mark Wilkins, "Learning Amazon Web Services (AWS): A Hands-On Guide to the Fundamentals of AWS Cloud", Publisher(s): Addison-Wesley Professional, O'Reilly Media 2019.

2."Mastering AWS Cost Optimization: Real-world technical and operational cost-saving best practices (Second Edition)", by Eli Mansoor and Yair Green 2020

Web links and Video Lectures:

● https://awsacademy.instructure.com/courses/3515/modules

Course Outcomes (COs):

CO1	Analyze the fundamental concepts of cloud computing and its benefits.
CO2	Adapt knowledge of AWS global infrastructure and key AWS services.
CO3	Implement basic networking and content delivery concepts in AWS.
CO4	Collaborate various AWS storage options and manage databases in AWS
CO5	Design well-architected cloud solutions and implement automatic scaling and monitoring using AWS.

CO-PO-PSO Mapping:

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

Advanced Java Programming
[As per NEP Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2024-2025)
SEMESTER – V

Course Code	**22CS544**	**CIE Marks**	**50**
Number Lecture Hour/Week	**03**	**SEE Marks**	**50**
Number of Lecture Hours	**40**	**Exam Hours**	**03**

CREDITS-03

Course Objectives: This course will enable students to

- Understand fundamentals concepts of Java programming
- How to write generic java programs.
- Understand collection and frameworks
- To learn java servlet
- Understand JDBC

Modules	Hours
Module -I	
Enumerations, Autoboxing and Annotations(metadata): Enumerations, Enumeration fundamentals, the values () and valueOf() Methods, java enumerations are class types, enumerations Inherits Enum, example, type wrappers, Autoboxing, Autoboxing and Methods, Autoboxing/Unboxing occurs in Expressions, Autoboxing/Unboxing, Boolean and character values, Autoboxing/Unboxing helps prevent errors, A word of Warning. Annotations, Annotation basics, specifying retention policy, Obtaining Annotations at run time by use of reflection, Annotated element Interface, Using Default values, Marker Annotations, Single Member annotations, Built-In annotations. Textbook 1: Lesson 12	**08**
Module -II	
The collections and Framework: Collections Overview, Recent Changes to Collections, The Collection Interfaces, The Collection Classes, Accessing a collection Via an Iterator, Storing User Defined Classes in Collections, The Random Access Interface, Working With Maps, Comparators, The Collection Algorithms, Why Generic Collections?, The legacy Classes and Interfaces, Parting Thoughts on Collections. Text Book 1: Ch.17	**08**
Module -III	
String Handling: The String Constructors, String Length, Special String Operations, String Literals, String Concatenation, String Concatenation with Other Data Types, String Conversion and toString( ) Character Extraction, charAt( ), getChars( ), getBytes( ) toCharArray(), String Comparison, equals( ) and equalsIgnoreCase( ), regionMatches( ) startsWith( ) and endsWith( ), equals( ) Versus == , compareTo( ) Searching Strings,	**08**

Modifying a String, substring( ), concat( ), replace( ), trim( ), Data Conversion Using valueOf( ), Changing the Case of Characters Within a String, Additional String Methods, StringBuffer , StringBuffer Constructors, length( ) and capacity( ), ensureCapacity( ), setLength( ), charAt( ) and setCharAt( ), getChars( ),append( ), insert( ), reverse( ), delete( ) and deleteCharAt( ), replace( ), substring( ), Additional StringBuffer Methods, StringBuilder Text Book 1: Ch 15	
Module-IV	
Background; The Life Cycle of a Servlet; Using Tomcat for Servlet Development; A simple Servlet; The Servlet API; The Javax.servlet Package; Reading Servlet Parameter; The Javax.servlet.http package; Handling HTTP Requests and Responses; Using Cookies; Session Tracking. Java Server Pages (JSP): JSP, JSP Tags, Tomcat, Request String, User Sessions, Cookies, Session Objects Text Book 1: Ch 31 Text Book 2: Ch 11	**08**
Module-V	
The Concept of JDBC; JDBC Driver Types; JDBC Packages; A Brief Overview of the JDBC process; Database Connection; Associating the JDBC/ODBC Bridge with the Database; Statement Objects; ResultSet; Transaction Processing; Metadata, Data types; Exceptions. Text Book 2: Ch 06	**08**

Question paper pattern

- The question paper will have TEN questions
- Each full question carries 20 marks
- There will be two full questions (with a maximum of THREE sub questions) from each module
- Each full question will have sub questions covering all topics under a module.

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

Textbooks:

1.Herbert Schildt: JAVA the Complete Reference, 7th/9th Edition, Tata McGraw Hill, 2007.
2.Jim Keogh: J2EE-TheCompleteReference, McGraw Hill, 2007.

Reference Books:

1. Y. Daniel Liang: Introduction to JAVA Programming, 7thEdition, Pearson Education, 2007.
2. Stephanie Bodoff et al: The J2EE Tutorial, 2nd Edition, Pearson Education,2004.
3. Uttam K Roy, Advanced JAVA programming, Oxford University press, 2015.

Online Courses and Video Lectures

https://nptel.ac.in/courses/106105191

Course Outcomes (COs):

CO1	Interpret the need for advanced Java concepts like enumerations, Auto Boxing and annotations.
CO2	Demonstrate the concept of Collections, Comparators, Legacy classes and Interfaces.
CO3	Understand the use of string handling functions.
CO4	Develop distributed web application using Servlets and JSP.
CO5	Apply the concepts of JDBC, Transaction processing, statement objects and Resultset to perform operations on Database

CO-PO-PSO Mapping:

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

Introduction to Data Structures
[As per NEP Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2024-2025)
SEMESTER – V

Course Code	**22CS551**	**CIE Marks**	**50**
Number Lecture Hour/Week	**04**	**SEE Marks**	**50**
Number of Lecture Hours	**50**	**Exam Hours**	**03**

CREDITS-04

Course Objectives:

- Introduce elementary data structures.
- Understand pointers for indirection reference
- Analyze Linear Data Structures: Stack, Queues, Lists
- Analyze Non-Linear Data Structures: Trees
- Assess appropriate data structure during program development/Problem Solving

Modules	**Hours**
Module -I	
Introduction:Introduction to arrays: one-dimensional arrays, two dimensional arrays, initializing two dimensional arrays, Multidimensional arrays. Introduction to Pointers: Pointer concepts, accessing variables through pointers, Dynamic memory allocation, pointers applications. Introduction to structures and unions: Declaring structures, Giving values to members, structure initialization, arrays of structures, nested structure, unions, size of structures. Textbook 1: Ch 8.3 to 8.15, Ch 12.3 to 12.19 Textbook 2: Ch 2.1 to2.13,2.51 ,2.80 to 2.98	**10**
Module -II	
Linear Data Structures-Stacks and queues: Introduction, Stack representation in Memory, Stack Operations, Stack Implementation, Applications of Stack. Introduction, Queues-Basic concept, Logical representation of Queues, Queue Operations and its types, Queue Implementation, Applications of Queue. Textbook 2: Ch 6.1 to 6.14 ,Ch 8.1,8.2	**10**
Module -III	
Linear Data Structures-Linked List:Introduction, Linked list Basic concept, Logical representation of Linked list, Self-Referential structure, Singly-linked List Operations and Implementation, Circular Linked List, applications of Linked list. Textbook 1: Ch 15.1 ,15.3,15.4,15.8 Textbook 2: Ch 9.2.9.5	**10**

Module-IV	
Non-Linear Data Structures – Trees:Introduction, Basic concept, Binary Tree and its types, Binary Tree Representation, Binary Tree Traversal, Binary Search tree, Expression Trees. Textbook1: Ch 16.1,16.2 Textbook2:Ch 10.1,10.2,10.4,10.6.3	**10**
Module-V	
Sorting and Searching ,Sorting: Introduction, Bubble sort, Selection sort, Insertion sort Searching: Introduction, Linear search, Binary search. Textbook1: Ch 17.1,17.2.2, 17.2.4, 17.3.1,17.3.2 Textbook2: Ch 11.1.,11.2,11.3,11.7,11.10.1,11.10.2	10

Question paper pattern

- The question paper will have TEN questions
- Each full question carries 20 marks
- There will be two full questions (with a maximum of THREE sub questions) from each module
- Each full question will have sub questions covering all topics under a module.

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

Textbooks

1. C Programming and data structures, E Balaguruswamy 4th Edition, 2007, McGraw Hill

2. Systematic approach to Data structures using C, A M Padma Reddy, 7thEdition 2007, Sri Nandi Publications.

Reference Books

1. Ellis Horowitz and Sartaj Sahni, Fundamentals of Data Structures in C, 2nd Ed, Universities Press, 2014.

2. Seymour Lipschutz, Data Structures Schaum‘s Outlines, Revised 1st Ed, McGraw Hill, 2014

Weblinks and Video Lectures (e-Resources):

1. https://www.youtube.com/watch?v=DFpWCl_49i0
2. https://www.youtube.com/watch?v=x7t_-ULoAZM
3. https://www.youtube.com/watch?v=I37kGX-nZEI
4. https://www.youtube.com/watch?v=XuCbpw6Bj1U
5. https://www.youtube.com/watch?v=R9PTBwOzceo
6. https://www.youtube.com/watch?v=qH6yxkw0u78

Course Outcomes (COs):

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

Fundamentals of Database Management System
[As per NEP Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2024-2025)
SEMESTER – V

Course Code	**22CS552**	**CIE Marks**	**50**
Number Lecture Hour/Week	04	**SEE Marks**	**50**
Number of Lecture Hours	50	**Exam Hours**	**03**

CREDITS:04

Course Objectives:

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

Module I	**Hours**
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 recoverability, Characterizing schedules based on Serializability, Transaction support in SQL	**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. 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

1. SQL and Relational Theory
2. (How to Write Accurate SQL code), C.J. Date, O'REILLY Publication
3. 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

Course Outcomes(COs):

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

SYSTEM SOFTWARE AND COMPILER DESIGN LABORATORY
[As per NEP Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2024-2025)
SEMESTER – V

Course Code	**22CSL56**	**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:

- 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
- To understand the various phases in the design of a compiler.
- To understand the design of top-down and bottom-up parsers.
- 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: $E \rightarrow E + T|TT \rightarrow T * F|F$, F→ $(E)|id$ 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 50 marks

Course Outcomes (COs):

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	Demonstrate theoretical concept of System Software and Compiler Design through series of experiment

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 Laboratory [As per Choice Based Credit System (CBCS) scheme] (Effective from the academic year 2024-2025) SEMESTER – V			
Course Code	**22CSL57**	**CIE Marks**	**50**
Number of Laboratory Hours/Week	**02**	**SEE Marks**	**50**
Total Teaching hours	**30**	**Exam Hours**	**03**
CREDITS – 01			

Course Objectives: This course will enable students

1.Understand and implement the fundamentals of socket programming.
2. To analyze the traffic flow and the contents of protocol frames.
3. Learn and apply cryptographic algorithms.
4. To understand the network simulator environment and visualize a network topology and observe its performance

1. Using TCP/IP Socket programming implement a program to transfer the contents of a requested file from server to the client using TCP/IP sockets.
2. Implement the data link layer farming methods such as character stuffing and bit stuffing.
3. Implement on a data of set of characters the three CRC polynomials-CRC 12, CRC 16 and CRC CCIP.
4. Write a program for frame sorting techniques used in buffers.
5. Write a program for Hamming Code generation for error detection and correction.
6. Take an example subnet graph with weights indicating delay between nodes. Now obtain routing table at each node using distance vector routing algorithm.
7. Using bucket algorithm, design a program to achieve traffic management at flow level by implementing closed loop control technique.
8. Using RSA algorithm encrypt a text data and decrypt the same.
9. a.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.
b.Write NS 3 Program to configure two nodes on an 802.11b physical layer, with 802.11b NICs in Ad hoc 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.
10. a Configure network topology using switch and router (LAN, Internet).
b.Configure network topology to implement VLAN using packet tracer.

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

Course Outcomes (COs):

CO1	Demonstrate understanding of fundamental networking concepts including protocols, error control, routing algorithms, and traffic shaping mechanisms.
CO2	Design and implement network-based solutions using programming, simulation tools like NS-3, and Packet Tracer to analyze and optimize network performance.
CO3	Debug and troubleshoot issues effectively.
CO4	Analyze and interpret the performance of network systems and algorithms using appropriate metrics
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

SYSTEMS PROGRAMMING LABORATORY
[As per NEP Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2024-2025)
SEMESTER– V

Course Code	**22CSL581**	**CIEMarks**	**50**
Number of Lecture Hours/Week	**02**	**SEEMarks**	**50**
Total Number of Lecture Hours	**30**	**Exam Hours**	**03**

CREDITS: 01

Course Objectives:

- To be able to introduce Unix System Programming basics and program design with functions.
- To understand a range of Object-Oriented Programming, as well as in-depth data, file and communication processing techniques.
- 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

Course Outcomes (COs):

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

COMPUTER GRAPHICS AND FUNDAMENTALS OF IMAGE PROCESSING LABORATORY
[As per NEP Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2024-2025)
SEMESTER– V

Course Code	**22CSL582**	**CIEMarks**	**50**
Number of Lecture Hours/Week	**02**	**SEEMarks**	**50**
Total Number of Lecture Hours	**30**	**Exam Hours**	**03**

CREDITS 01

Course Objectives:

- Demonstrate the use of Open GL.
- Demonstrate the different geometric object drawing using openGL
- Demonstration of 2D/3D transformation on simple objects.
- Demonstration of lighting effects on the created objects.
- Demonstration of Image processing operations on image/s.

- Installation of OpenGL /OpenCV/ Python and required headers
- Simple programs using OpenGL (Drawing simple geometric object like line, circle,rectangle, square)
- Simple programs using OpenCV (operation on an image/s)

Programs list

1.	Develop a program to draw a line using Bresenham‘s line drawing technique
2.	Develop a program to demonstrate basic geometric operations on the 2D object
3.	Develop a program to demonstrate basic geometric operations on the 3D object
4.	Develop a program to demonstrate 2D transformation on basic objects
5.	Develop a program to demonstrate 3D transformation on 3D objects
6.	Develop a program to demonstrate Animation effects on simple objects.
7.	Write a Program to read a digital image. Split and display image into 4 quadrants, up, down,right and left.
8.	Write a program to show rotation, scaling, and translation on an image.
9.	Read an image and extract and display low-level features such as edges, textures using filtering techniques.
10.	Write a program to blur and smoothing an image.
11.	Write a program to contour an image.
12.	Write a program to detect a face/s in an image.

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

Course Outcomes (COs):

CO1	Understand the fundamental concepts of computer graphics and image processing, including 2D/3D transformations, rendering algorithms, filtering, and basic computer vision techniques.
CO2	Apply computer graphics and image processing techniques through programming to create, manipulate objects and digital images.
CO3	Debug and troubleshoot issues effectively.
CO4	Analyze 2D/3D objects, digital images 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	2	2		2	-	-	-	-	-	-	2	2	-	3
CO2	3	2	2		2	-	-	-	-	-	-	2	2	-	3
CO3	3	2	2		2	-	-	-	-	-	-	2	2	-	3
CO4	3	2	2		2	-	-	-	-	-	-	2	2	-	3
CO5	3	2	2		2	-	-	-	-	-	-	2	2	-	3

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

Course Code	**22CSL583**	**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: 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	2	2	-	2	2	2	-	-	-	-	2	2	-	3
CO2	3	2	2	-	2	2	2	-	-	-	-	2	2	-	3
CO3	3	2	2	-	2	2	2	-	-	-	-	2	2	-	3
CO4	3	2	2	-	2	2	2	-	-	-	-	2	2	-	3
CO5	3	2	2	-	2	2	2	-	-	-	-	2	2	-	3

ADVANCED JAVA PROGRAMMING LABORATORY
[As per NEP Choice Based Credit System (CBCS) scheme]
(Effective from the academic year 2024-2025)
SEMESTER– V

Course Code	**22CSL584**	**CIEMarks**	**50**
Number of Lecture Hours/Week	**02**	**SEEMarks**	**50**
Total Number of Lecture Hours	**30**	**Exam Hours**	**03**

CREDITS -01

Course Objectives:

- Develop students' skills in building GUI-based applications using Java AWT and Swing components.
- Provide practical exposure to Java applets, event handling, and exception handling mechanisms.
- Enable students to design interactive applications such as calculators, forms, and multi-frame interfaces.

Programs list

1.	a) Write a Java program to create an Applet that displays student information and also set foreground and backgrounds. b)Write a Java Program to create an applet that scrolls a message from left to right c) Write a Java program to create an Applet that reads Employee information using parameters and displays name of employee ,designation ,salary and tax.
2.	a. Write a java program to draw Lines, ovals, filled ovals and arcs, filled arcs? b. Write a java program to draw rectangle, filled rectangle and rounded rectangle and filled rounded rectangle with any two colors. c) . Write a java program to draw a smiley face?
3.	a .Write a Java program to create an Applet that displays 4buttons each represents different colors.if a user click on particular button then that color is set as back ground to applet. b . Write a Java program to create an Applet that displays 2buttons .if a user click on one button then change name of another button and display the button clicked and vice-versa c. Write a Java program to create an Applet that displays 2TextFields and labels User Name and Password . Display Text Field entered and display selected text in username & display password d.Write a Java program that displays 4buttons and also count Number of button clicks on each button
4.	Write a Java program that creates a user interface to perform integer divisions. The user enters two numbers in the text fields, Num1 and Num2. The division of Num1 and Num2 is displayed in the Result field when the Divide button is clicked. If Num1 or Num2 were not an integer, the program would throw a Number Format Exception. If Num2 were Zero, the program would throw an Arithmetic Exception Display the exception in a message dialog box.
5.	Write a Java program that works as a simple calculator. Use a grid layout to arrange buttons for the digits and for the +, -,*, % operations. Add a text field to display the result.
6.	Write a Java Program a simple user form which reads the name of a user and mail id in Text fields, select gender with radio buttons, and selects some Known languages using checkboxes, and also enters an address in a text area. After filling details whenever a user press the “submit” button, then displays all the information about the user input.
7.	Write a Java Program to create multiple frames, which create a Frame2 with a ‘back ‘button, such that when a user click ‘back’ button, Frame 2 is closed and we see the Frame1 only

8.	Write a Java Program to create a student table, which includes name, roll no, branch and age or DOB
9.	Write a Java Program to create a tabbed pane with two tabs. In the first tab sheet, display some push buttons with names of Branches. In second tab sheet, display checkboxes with names of subjects.
10.	A) Write a java program to create a menu with several menu items by implementing JMenu. B)Write a java program to create a combo box with some name of some places. The user can select any one name from the list and the selected country name is displayed in the frame? (Use JComboBox) C)Write a java program to select multiple places and displayed in Frame using JList?

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 the fundamental concepts of Java programming, including applets, AWT, Swing, and exception handling.
CO2	Design and implement Java programs using GUI components and event-driven programming techniques.
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	3	-	3	2	2	-	-	-	-	2	2	-	3
CO2	3	3	3	-	3	2	2	-	-	-	-	2	2	-	3
CO3	3	3	3	-	3	2	2	-	-	-	-	2	2	-	3
CO4	3	3	3	-	3	2	2	-	-	-	-	2	2	-	3
CO5	3	3	3	-	3	2	2	-	-	-	-	2	2	-	3

PROJECT-V [As per Choice Based Credit System (CBCS) scheme] (Effective from the academic year 2024-25) 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 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

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

Course Code	**22ACS510A**	**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 50 marks

CO#	COURSE OUTCOMES
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

Microsoft Power BI

[As per Choice Based Credit System (CBCS) scheme]

(Effective from the academic year 2023-2024)

SEMESTER – VI

Course Code	**22AAD510B**	**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: This course will enable students

1. Understand the fundamentals of Power BI and its role in business intelligence and data visualization.
2. Learn how to prepare and clean data within Power BI Desktop, ensuring data quality for effective analysis.
3. Explore different methods of importing and loading data from various sources into Power BI Desktop.
4. Understand and apply DAX functions to perform powerful data analysis and calculations in Power BI.
5. Learn how to publish reports and dashboards, and share them securely with stakeholders via Power BI Service.

List of Programs

Implement the following using Microsoft Power BI tool.

1. Getting Started With power BI
2. Preparing Data in Power BI Desktop
3. Loading Data in Power BI Desktop
4. Data Modeling in Power BI Desktop
5. Advanced Data Modeling in Power BI Desktop
6. Using simple DAX queries in Power BI Desktop
7. Using complex DAX queries in Power BI Desktop
8. Designing a Report in Power BI Desktop
9. Data Analysis in Power BI Desktop
10. Publishing and Sharing Power BI Content

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	Utilize classical and modern cryptographic algorithms to implement secure data encryption, decryption, and key exchange mechanisms.
CO2	Apply cryptographic hash functions and bitwise operations for secure data processing and integrity verification.
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	3	3	3	-	3	2	2	-	-	-	-	2	2	-	3
CO2	3	3	3	-	3	2	2	-	-	-	-	2	2	-	3
CO3	3	3	3	-	3	2	2	-	-	-	-	2	2	-	3
CO4	3	3	3	-	3	2	2	-	-	-	-	2	2	-	3
CO5	3	3	3	-	3	2	2	-	-	-	-	2	2	-	3

Software Engineering
[As per NEP Choice Based Credit System (CBCS)scheme]
(Effective from the academic year 2024-2025)
SEMESTER – VI

Course Code	**22CS61**	**CIE Marks**	**50**
Number of Lecture Hours/Week	**03**	**SEE Marks**	**50**
Number of Lecture Hours	**40**	**Exam Hours**	**03**

CREDITS-03

Course Objectives:

- Understand fundamental concepts of software engineering.
- Provide comprehensive understanding of the requirements of engineering process and system modeling techniques.
- Impart knowledge of architectural and object-oriented design principles.
- Understand the basics of software project management for effective project delivery.
- Learn modern software development practices and testing techniques.

Modules	**Hours**
Module -I	
Introduction: Need for software engineering, Professional and ethical responsibility, case studies, Software Process models, Process Iteration, Process Iteration continued, Process Activities, Software requirements: Functional and Non-functional requirements, User requirements, System requirements, Interface specification, The software requirements document.	**08**
Module -II	
Requirements engineering process: Feasibility studies, Requirement's elicitation and analysis, Requirements validation, Requirements management. **System models:** Context models, Behavioral models, Data models, Object models, Structured methods.	**08**
Module -III	
Architectural Design: Architectural Design Decisions, Desing and implementation System organization, Modular Decomposition styles, Control styles. **Object oriented design**: Objects and Object Classes, An object-oriented design process Design evolution.	**08**
Module-IV	
Software Project Management: The Management Spectrum, Product, process and project, The W5HH principle, Critical practices, Estimation for Software Project: Software Project estimation, Decomposition Techniques, Empirical Estimation models **Project Scheduling:** Basic Concepts, Project Scheduling, Defining Task set and Task network, Scheduling **Risk Management**: Reactive versus proactive strategies, Software Risks, Risk identification, Risk mitigation, monitoring and management, The RMMM plan.	**08**
Module-V	
Rapid software development: Agile methods, Extreme programming, Rapid application development **Software evolution**: Legacy system evolution **Verification and Validation**: Planning verification and validation, Software inspections, Automated static analysis, Verification and formal methods	**08**

Software testing: System testing, Component testing, Test case design, Test automation.	

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. Software Engineering- Sommerville, 10th edition, Pearson Education.
2. Software Engineering, A practitioner's Approach- Roger S. Pressman, 6th edition, Mc Graw Hill International Edition.

REFERENCE BOOKS:

1. Software Engineering theory and Practice Shari Lawrence Pflieger, Joanne M Atlec 3rd edition Pearson Education.
2. Software Engineering Principles and Practice Waman S Javadekar 1st edition Tata McGraw Hill.

CO#	COURSE OUTCOMES
CO1	Apply software engineering concepts for software systems.
CO2	Demonstrate understanding of engineering requirements and apply system modeling techniques to analyze, specify, and represent software system requirements.
CO3	Utilize architectural and object-oriented design principles to create well-structured software systems.
CO4	Apply software project management skills to deliver projects effectively.
CO5	Implement software development strategies, and testing techniques to ensure quality and adaptability in software projects.

CO-PO-PSO Mapping:

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

Artificial Intelligence and Machine Learning
[As per NEP Choice Based Credit System (CBCS)scheme]
(Effective from the academic year 2024-2025)
SEMESTER – VI

Course Code	**22CS62**	**CIE Marks**	**50**
Number of Lecture Hours/Week	**03**	**SEE Marks**	**50**
Number of Lecture Hours	**40**	**Exam Hours**	**03**

CREDITS-03

Course Objectives:

- To study the concept of Artificial Intelligence and problem solving.
- Get to know approaches of inference, perception, Uncertain Knowledge and Reasoning
- To figure out advanced problem-solving paradigms and knowledge representation.
- To enable students to evaluate machine learning models
- To explore neural networks, build neural networks to solve various classification problems.

Modules	**Hours**
Module-I	
Introduction, Problem Solving: state space search and control strategies: Introduction, General problem solving, Characteristics of problem, Exhaustive Searches, Heuristic Search Techniques, Interative Deepening, Constant satisfaction	**08**
Module-II	
Problem reduction and Game playing, Logic concepts and logic programming: Introduction, Problem reduction, Game playing, Bounded look ahead strategy and Use of, Alpha-Beta Pruning, Two –player perfect information games. Propositional calculus, Propositional logic, Natural Deduction system, Axiomatic system, Semantic tableau system in propositional logic, resolution refutation in propositional logic, Predicate logic, Logic programming	**08**
Module-III	
Advanced problem-solving paradigm: planning- types of planning systems, Block world problem, logic based planning, Linear planning using a goal stack, Means-ends analysis, Non –linear planning strategies. Knowledge representation: Approaches to knowledge representation, knowledge representation	08
Module-IV	
Uncertainty Measure: Probability Theory, Bayesian Belief Networks, Machine Learning Paradigms: Machine learning system, supervised and unsupervised learnings, Inductive, deductive learning, Clustering	**08**
Module-V	
Support vector Machine, case-based reasoning and learning. ANN: Single Layer, Multilayer. RBF, Design issues in ANN, Recurrent Network	**08**

Question paper pattern

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

- Each full question carries 20 marks.

Each full question will have sub questions covering all topics under a module.

Textbooks

1. Artificial Intelligence, Saroj Kaushik Cengage Learning 2014 Edition.

Reference:

1 .Artificial Intelligence: Structures and Strategies for Complex Problem Solving, George F Luger Pearson Addison Wesley 6 th Ed, 2008.
2. Artificial Intelligence, E Rich, K Knight, and S B Nair Tata Mc-Graw Hill 3rd Ed, 2009.
3. Artificial Intelligence: A Modern Approach, Stuart Russell and Peter Norvig Prentice Hall 3rd, 2009.

CO#	COURSE OUTCOMES
CO1	Understand the fundamentals of problem-solving in AI, including state space Search.
CO2	Apply problem reduction techniques and logic concepts.
CO3	Apply predicate logic to represent complex real-world problems& distinguish between different machine learning paradigms.
CO4	Analyze the concept learning and decision tree learning approaches.
CO5	Understand the theory and architecture of artificial neural networks.

CO-PO-PSO Mapping:

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

Full Stack Development
[As per NEP Choice Based Credit System (CBCS)scheme]
(Effective from the academic year 2024-2025)
SEMESTER – VI

Course Code	**22CS631**	**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:

- Explain the use of learning full stack web development.
- Make use of rapid application development in the design of responsive web pages.
- Illustrate Models, Views and Templates with their connectivity in Django for full stack web development.
- Demonstrate the use of state management and admin interfaces automation in Django.
- 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 andLoose Coupling, Errors in Django, Wild Card patterns in URLS. Textbook 1: Chapter 1 and Chapter 3	**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 Textbook 1: Chapter 4 and Chapter 5	**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. Textbook 1: Chapters 6, 7 and 8	**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, Usersand Authentication. Textbook 1: Chapters 9, 11 and 12	**08**
Module V	
jQuery and AJAX Integration in Django: Ajax Solution, Java Script, XHTML Http Request and Response, HTML, CSS, JSON, iFrames, Settings of Java Script in Django, jQuery and Basic AJAX, jQuery AJAX Facilities, Using jQuery UI Autocomplete in Django Textbook 2: Chapters 1, 2 and 7	**08**
Question paper pattern	

- The question paper will have TEN questions, Two questions from each module.
- The students will have to answer 5 full questions, selecting one full question from each module.
- Each full question carries 20 marks.

Each full question will have sub questions covering all topics under a module.

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 and trying to implement in coding way.
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	-

Cryptography and Network Security
[As per NEP Choice Based Credit System (CBCS)scheme]
(Effective from the academic year 2024-2025)
SEMESTER – VI

Course Code	**22CS632**	**CIE Marks**	**50**
Number of Lecture Hours/Week	**03**	**SEE Marks**	**50**
Number of Lecture Hours	**40**	**Exam Hours**	**03**

CREDITS-03

Course Objectives:

- To understand basics of Cryptography and Network Security.
- To be able to secure a message over insecure channel by various means.
- To learn about how to maintain Confidentiality, Integrity and Availability of data.
- To understand various protocols for network security to protect against the threats in the networks.

Modules	**Hours**
Module -I	
Security Concepts: Introduction, The need for security, Security approaches, Principles of security, Types of Security attacks, Security services, Security Mechanisms, A model for Network Security. **Cryptography Concepts and Techniques:** Introduction, plain text and cipher text, substitution techniques, transposition techniques, encryption and decryption, symmetric and asymmetric key cryptography, steganography, key range and key size, possible types of attacks.	**08**
Module -II	
Symmetric key Ciphers: Block Cipher principles, DES, AES, Blowfish, RC5, IDEA, Block cipher operation, Stream ciphers, RC4. **Asymmetric key Ciphers:** Principles of public key cryptosystems, RSA algorithm, Elgamal Cryptography, Diffie-Hellman Key Exchange, Knapsack Algorithm.	**08**
Module -III	
Cryptographic Hash Functions: Message Authentication, Secure Hash Algorithm (SHA-512), **Message authentication codes:** Authentication requirements, HMAC, CMAC, Digital signatures, Elgamal Digital Signature Scheme. **Key Management and Distribution:** Symmetric Key Distribution Using Symmetric & Asymmetric Encryption, Distribution of Public Keys, Kerberos, X.509 Authentication Service, Public – Key Infrastructure.	**08**
Module-IV	
Transport-level Security: Web security considerations, Secure Socket Layer and Transport Layer Security, HTTPS, Secure Shell (SSH). **Wireless Network Security:** Wireless Security, Mobile Device Security, IEEE 802.11 Wireless LAN, IEEE 802.11i Wireless LAN Security.	**08**
Module-V	

E-Mail Security: Pretty Good Privacy, S/MIME **IP Security:** IP Security overview, IP Security architecture, Authentication Header, Encapsulating security payload, Combining security associations, Internet Key Exchange. **Case Studies on Cryptography and security:** Secure Multiparty Calculation, Virtual Elections, Single sign On, Secure Inter-branch Payment Transactions, Cross site Scripting Vulnerability.	**08**

TEXTBOOKS

1. Cryptography and Network Security - Principles and Practice: William Stallings, Pearson Education, 6th Edition.
2. Cryptography and Network Security: Atul Kahate, Mc Graw Hill, 3rd Edition.

REFERENCE BOOK:

1. Cryptography and Network Security: C K Shyamala, N Harini, Dr T R Padmanabhan, Wiley
2. India, 1st Edition.
3. Cryptography and Network Security: Forouzan Mukhopadhyay, Mc Graw Hill, 3rd Edition.
4. Information Security, Principles, and Practice: Mark Stamp, Wiley India.
5. Principles of Computer Security: WM. Arthur Conklin, Greg White, TMH.
6. Introduction to Network Security: Neal Krawetz, CENGAGE Learning.
7. Network Security and Cryptography: Bernard Menezes, CENGAGE Learning.

Question paper pattern

- The question paper will have TEN questions, Two questions from each module.
- The students will have to answer 5 full questions, selecting one full question from each module.
- Each full question carries 20 marks
- Each full question will have sub questions covering all topics under a module.

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	-

Natural Language Processing
[As per NEP Choice Based Credit System (CBCS)scheme]
(Effective from the academic year 2024-2025)
SEMESTER– VI

Course Code	**22AD633**	**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: This course will enable students.

- Will be able to understand the wide spectrum of problem statements, tasks, and solution approaches 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 NLP product.
- Understand best practices, opportunities, and the roadmap for NLP from a business and product leader'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 for Relation Extraction and Experimental Evaluation. **Mining Diagnostic Text Reports by Learning to Annotate Knowledge Roles:** Introduction, Domain Knowledge and Knowledge Roles, Frame Semantics and Semantic Role 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 Semantic Analysis, 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, Non classical, 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, Two questions from each module.
- The students will have to answer 5 full questions, selecting one full question from each module.
- Each full question carries 20 marks.
- Each full question will have sub questions covering all topics under a module.

Textbooks:

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 Natural Language 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	-

Mobile Application Development
[As per NEP Choice Based Credit System (CBCS)scheme]
(Effective from the academic year 2024-2025)
SEMESTER– VI

Course Code	**22CS634**	**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:

- Learn to set up an Android application development environment.
- Illustrate user interfaces for interacting with apps and triggering actions.
- Interpret tasks used in handling multiple activities.
- Identify options to save persistent application data.
- Appraise the role of security and performance in Android applications

Modules	Hours
Module I	
Get started: Build your first app: Introduction to Android, Create Your First Android App, Layouts, Views and Resources, Text and Scrolling Views, Resources to Help You Learn, Activities: Understanding Activities and Intents, Activities and Implicit Intents, Testing, Debugging and using support libraries: The Android Studio Debugger, Testing your App, The Android Support Library.	**08**
Module II	
User Interaction: User Input Controls, Menus, Screen Navigation, Recycler View, Delightful user experience: Drawable, Styles, and Themes, Material Design, Providing Resources for Adaptive Layouts, Testing your UI: Testing the User Interface.	**08**
Module III	
Background Tasks: AsyncTask and AsyncTaskLoader, Connect to the Internet, Broadcast Receivers, Services, Triggering, scheduling and optimizing background tasks: Notifications, Scheduling Alarms, Transferring Data Efficiently.	**08**
Module IV	
All about data, Preferences and Settings: Storing Data, Shared Preferences, App Settings, Storing data using SQLite: SQLite Primer, SQLite Database, Sharing data with content providers: Share Data Through Content Providers, Loading data using Loaders: Loaders.	**08**
Module V	
Permissions, Performance and Security, Firebase and AdMob, Publish	**08**

Question paper pattern

- The question paper will have TEN questions, Two questions from each module.
- The students will have to answer 5 full questions, selecting one full question from each module.
- Each full question carries 20 marks.

Each full question will have sub questions covering all topics under a module.

Textbooks:

1. Google Developer Training, "Android Developer Fundamentals Course – Concept Reference", Google Developer Training Team, 2017.

Reference Books:

1. Erik Hellman, "Android Programming – Pushing the Limits", 1st Edition, Wiley India Pvt Ltd, 2014.
2. Dawn Griffiths and David Griffiths, "Headfirst Android Development", 1st Edition, O'Reilly SPD Publishers, 2015.

3. J F DiMarzio, “Beginning Android Programming with Android Studio”, 4th Edition, Wiley India Pvt Ltd, 2016. ISBN-13: 978-8126565580 4. 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	-

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	**22CS641**	**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	-

Blockchain Technology
[As per NEP Choice Based Credit System (CBCS)scheme]
(Effective from the academic year 2024-2025)
SEMESTER – VI

Course Code	**22CS642**	**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:

- Introduction to blockchain technology, covering its fundamentals, consensus mechanisms, and distributed system challenges. It explores blockchain types, applications, benefits, and limitations.
- Explores decentralization through blockchain, covering decentralized organizations, DAOs, and full ecosystem decentralization. It also examines cryptographic foundations, including symmetric and asymmetric encryption, digital signatures, and key management.
- Bitcoin's fundamentals, including transactions, blockchain, and network structure. It also examines alternative coins, their theoretical foundations of alternative coins.
- Introduce smart contracts, including their definition and Ricardian contracts. It also explores Ethereum, its blockchain structure, key components, and precompiled contracts.
- Students get the opportunity to know the implementation of this technology in various fields.

Modules	**Hours**
Module I	
Blockchain 101: Distributed systems, CAP theorem, Byzantine Generals problem, Consensus, History of blockchain, Introduction to blockchain, Generic elements of blockchain, Features of blockchain, application of blockchain, Tiers of blockchain, Types of blockchain, and blockchain, Benefits and limitations of blockchain	**08**
Module II	
Decentralization and Cryptography: Decentralization using blockchain, Methods of decentralization, Routes to decentralization, Blockchain and full ecosystem decentralization. Decentralized organizations, Decentralized autonomous organizations, Decentralized autonomous corporations, Decentralized autonomous societies. Cryptography and Technical Foundations: Cryptographic primitives- symmetric cryptography-stream ciphers, block ciphers, AES, DES, SHAs, Merkle Tree, Patricia trees, DHTs, Digital Signature, ECDSA, Asymmetric cryptography, Public and private keys- RSA, ECC.	**08**
Module III	
Bitcoin and Alternative Coins A: Bitcoin, Transactions, Blockchain, Bitcoin Network. Alternative Coins, Theoretical foundations, Bitcoin limitations, Namecoin, Litecoin, Primecoin, Zcash	**08**
Module IV	
Smart Contracts and Ethereum 101: Smart Contracts: Definition, Ricardian contracts. Ethereum 101: Introduction, Ethereum blockchain, Elements of the Ethereum blockchain, Precompiled contracts.	**08**
Module V	
Alternative Blockchains: Blockchains Blockchain-Outside of Currencies: Internet of Things, Government, Health, Finance, Media.	**08**

Question paper pattern

- The question paper will have TEN questions, Two questions from each module.
- The students will have to answer 5 full questions, selecting one full question from each module.
- Each full question carries 20 marks.

Each full question will have sub questions covering all topics under a module.

Text Books:

1.Mastering Blockchain - Distributed ledgers, decentralization and smart contracts explained, Imran Bashir, Packt Publishing Ltd, Second Edition, ISBN 978-1-78712-544-5, 2017

Reference Books:

1. Bitcoin and Cryptocurrency Technologies, Arvind Narayanan, Joseph Bonneau, Edward Felten, 2016
2. Blockchain Basics: A Non-Technical Introduction in 25 Steps, Daniel Drescher, Apress, First Edition, 2017
3. Mastering Bitcoin: Unlocking Digital Cryptocurrencies, Andreas M. Antonopoulos, O'Reilly Media, First Edition, 2014

CO#	COURSE OUTCOMES
CO1	Describe the foundational concepts of blockchain, including its evolution, types, CAP theorem implications, and the benefits and limitations of blockchain in distributed systems.
CO2	Apply decentralization methods and demonstrate the use of cryptographic primitives such as public and private keys in blockchain systems.
CO3	Analyze the architecture and transaction flow of Bitcoin and evaluate the limitations and innovations introduced by alternative coins
CO4	Develop and test smart contracts using Ethereum by understanding its blockchain components and contract models.
CO5	Evaluate the application of blockchain technology beyond cryptocurrencies in sectors such as IoT, government, finance, and healthcare

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	-

Object Oriented Modeling and Design
[As per NEP Choice Based Credit System (CBCS)scheme]
(Effective from the academic year 2024-2025)
SEMESTER – VI

Course Code	**22CS643**	**CIE Marks**	**50**
Number of Lecture Hours/Week	**03**	**SEE Marks**	**50**
Number of Lecture Hours	**40**	**Exam Hours**	**03**

CREDITS-03

Course Objectives:

- Describe the concepts involved in Object-Oriented modelling and their benefits.
- Demonstrate the concept of use-case model, sequence model and state chart model for a given problem.
- Explain the facets of the unified process approach to design and build a Software system.
- Translate the requirements into implementation for Object Oriented design.
- Choose an appropriate design pattern to facilitate development procedure.

Modules	**Hours**
Module-I	
Advanced object and class concepts; Association ends; N-ary associations; Aggregation; Abstract classes; Multiple inheritance; Metadata; Reification; Constraints; Derived Data; Packages. State Modeling: Events, States, Transistions and Conditions, State Diagrams, State diagram behaviour. Textbook-1: 4, 5	08
Module-II	
Use Case Modelling and Detailed Requirements: Overview; Detailed object-oriented Requirements definitions; System Processes-A use case/Scenario view; Identifying Input and outputs-The System sequence diagram; Identifying Object Behaviour-The state chart Diagram; Integrated Object-oriented Models. Textbook-2:Chapter- 6:Page 210 to 250	**08**
Module-III	
Process Overview, System Conception and Domain Analysis: Process Overview: Development stages; Development life Cycle; System Conception: Devising a system concept; elaborating a concept; preparing a problem statement. Domain Analysis: Overview of analysis; Domain Class model: Domain state model; Domain interaction model; Iterating the analysis. Textbook-1:Chapter- 10,11,and 12	**08**
Module-IV	
Use case Realization: The Design Discipline within up iterations: Object Oriented Design-The Bridge between Requirements and Implementation; Design Classes and Design within Class Diagrams; Interaction Diagrams-Realizing Use Case and defining methods; Designing with Communication Diagrams; Updating the Design Class Diagram; Package Diagrams-Structuring the Major Components; Implementation Issues for Three-Layer Design. Textbook-2: Chapter 8: page 292 to 346	**08**
Module-V	
Design Patterns: Introduction; what is a design pattern? Describing design patterns, the catalogue of design patterns, Organizing the catalogue, How design patterns solve design problems, how to select a design patterns, how to use a design pattern; Creational patterns: prototype and singleton (only); structural patterns adaptor and proxy (only). Textbook-3: Ch-1: 1.1, 1.3, 1.4, 1.5, 1.6, 1.7, 1.8,Ch-3,Ch-4.	**08**

Question paper pattern

- The question paper will have TEN questions, Two questions from each module.
- The students will have to answer 5 full questions, selecting one full question from each module.
- Each full question carries 20 marks.

Each full question will have sub questions covering all topics under a module.

Textbooks

1. Michael Blaha, James Rumbaugh: Object Oriented Modelling and Design with UML,2nd Edition, Pearson Education,2005

2. Satzinger, Jackson and Burd: Object-Oriented Analysis & Design with the Unified Process, Cengage Learning, 2005.

3. Erich Gamma, Richard Helm, Ralph Johnson and john Vlissides: Design Patterns –Elements of Reusable Object-Oriented Software, Pearson Education,2007.

Reference:

1. Grady Booch et. al.: Object-Oriented Analysis and Design with Applications,3rd Edition,Pearson Education,2007.

2. Frank Buschmann, RegineMeunier, Hans Rohnert, Peter Sommerlad, Michel Stal: Pattern – Oriented Software Architecture. A system of patterns , Volume 1, John Wiley and Sons.2007.

3. Booch, Jacobson, Rambaugh : Object-Oriented Analysis and Design with Applications, 3rd edition, pearson, Reprint 2013

CO#	COURSE OUTCOMES
CO1	Understand and apply advanced object-oriented modeling to represent system behavior effectively using UML.
CO2	Develop detailed object-oriented requirements models using use cases, sequence diagrams, and state charts to capture system behavior and interactions.
CO3	Analyze and define system requirements through domain analysis, class modeling, and interaction modeling to prepare foundational models for object-oriented development.
CO4	Design object-oriented solutions using interaction and class diagrams.
CO5	Apply design patterns to solve common software design problems

CO-PO-PSO Mapping:

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

Cognitive Science
[As per NEP Choice Based Credit System (CBCS)scheme]
(Effective from the academic year 2024-2025)
SEMESTER – VI

Course Code	**22CS644**	**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:

- Explores the foundations of cognitive science, including its historical roots, computational theories, and linguistic analysis.
- Examine cognitive systems, brain anatomy, and computational modeling in neural processing.
- Explores information-processing models, including physical symbol systems, the language of thought, and machine learning.
- Explores neural networks and distributed information processing in cognition, covering single-layer, multilayer models, and key processing features.
- Examine the organization of cognitive systems, intelligent agent architectures, and modularity theories.

Modules	**Hours**
Module I	
Introduction : The prehistory of cognitive science, The reaction against behaviorism in psychology, The theory of computation and the idea of an algorithm, Linguistics and the formal analysis of language, Information-processing models in psychology, The discipline matures: Three milestones: Language and micro-worlds, How do mental images represent? , An interdisciplinary model of vision.	**08**
Module II	
The turn to the brain : Cognitive systems as functional systems , The anatomy of the brain and the primary visual pathway , Extending computational modeling to the brain , Mapping the stages of lexical processing , THE INTEGRATION CHALLENGE, Cognitive science and the integration challenge; Cognitive science: An interdisciplinary endeavor, Levels of explanation: The contrast between psychology and neuroscience, The integration challenge.	**08**
Module III	
Information-Processing Models of the Mind : Physical symbol systems and the language of thought ; The physical symbol system hypothesis , From physical symbol systems to the language of thought , The Chinese room argument , Applying the symbolic paradigm ;Expert systems, machine learning, and the heuristic search hypothesis , ID3: An algorithm for machine learning , WHISPER: Predicting stability in a block world.	**08**
Module IV	
Neural networks and distributed information processing : Neural inspired models of information processing , Single-layer networks and Boolean functions , Multilayer networks, Information processing in neural networks: Key features, Neural network models of cognitive processes; Language and rules: The challenge for information processing models, Language learning in neural networks, Object permanence and physical reasoning in infancy, Neural network models of children's physical reasoning.	**08**

Module V	
The Organization of The Mind : How are cognitive systems organized?; Architectures for intelligent agents , Fodor on the modularity of mind , The massive modularity hypothesis , Strategies for brain mapping ; Structure and function in the brain , , Studying cognitive functioning: Techniques from neuroscience.	**08**

Question paper pattern

- The question paper will have TEN questions, Two questions from each module.
- The students will have to answer 5 full questions, selecting one full question from each module.
- Each full question carries 20 marks.

Each full question will have sub questions covering all topics under a module.

Textbooks

1.Cognitive Science :An Introduction to the Science of the Mind, Jose Luis Bermudez ,Cambridge University Press, Second Edition 2020.

Reference Books:

1.José Luis Bermúdez, Cognitive Science: An Introduction to the Science of the Mind, Cambridge University Press

2. Michael R. W. Dawson , Mind, Body, World: Foundations of Cognitive Science, UBC Press

3. Daniel Kolak, William Hirstein, Peter Mandik, Jonathan Waskan, Cognitive Science, An Introduction to Mind and Brain, Routledge Taylor and Francis Group

4. Amit Konar – Artificial Intelligence and Soft computing: Behavioral and Cognitive Modeling of the Human Brain, CRC Press

CO#	COURSE OUTCOMES
CO1	Describe the foundational developments that led to the emergence of cognitive science, including key theories in psychology, linguistics, and computation
CO2	Explain how brain anatomy and function relate to cognitive processes and analyze the interdisciplinary challenges in integrating psychology and neuroscience
CO3	Apply symbolic models and learning algorithms to represent cognitive tasks.
CO4	Compare neural and symbolic models in simulating cognitive processes.
CO5	Identify cognitive system architectures and basic brain mapping techniques.

CO-PO-PSO Mapping:

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

OOPS with C++
[As per NEP Choice Based Credit System (CBCS)scheme]
(Effective from the academic year 2024-2025)
SEMESTER – VI

Course Code	**22CS651**	**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:

- Covers C++ programming basics, including OOP, C comparison, console I/O, variables, and function concepts.
- Introduces classes and objects in C++, covering member functions, data, and interactions with functions and arrays. It also explores dynamic memory management, including allocation and deallocation techniques.
- Covers basics such as operator overloading, type conversion, constructors, destructors, and inheritance concepts, including function overriding and access control in C++.
- Covers stream handling in C++, including text and binary I/O, file operations using the fstream class, and file pointers. It also explores random access file handling and stream class hierarchy.
- Introduces exception handling in C++, covering error handling using C-style and C++-style try/throw/catch constructs. It also explores the limitations of exception handling mechanisms.

Modules	**Hours**
Module I	
Introduction to C++: Procedure-Oriented Programming System, Object-Oriented Programming System, Comparison of C++ and C, Console Input/Output in C++, Variables in C++, Reference Variable in C++, Function Prototyping, Function Overloading.	**10**
Module II	
Classes and Objects: Introduction to classes and objects, member functions, and member data, objects and functions, objects and arrays. **Dynamic memory management**: introduction, dynamic memory allocation, dynamic memory deallocation.	**10**
Module III	
Operator overloading, Type Conversion, New Style Casts, and RTTI: operator overloading, overloading the various operators, type conversion . **Constructors and Destructors:** constructors, destructors. **Inheritance:** Introduction to Inheritance, Base class and Derived class pointers, function overriding, base class initialization, protected access specifier.	**10**
Module IV	
Stream Handling: streams, class hierarchy for handling systems, text and binary input/output, opening and closing files, files as objects of fstream class, file pointers , random access files.	**10**
Module V	
Exception Handling: Introduction , C-Style handling of Error-generation code, C++ style solution-try/throw/catch construct, limitations of exception handling.	**10**

Question paper pattern

- The question paper will have TEN questions, Two questions from each module.
- The students will have to answer 5 full questions, selecting one full question from each module.
- Each full question carries 20 marks.

Each full question will have sub questions covering all topics under a module.

Textbook/ Textbooks:

Object Oriented Programming with C++, Sourav Sahay.

Reference Books:

Object Oriented Programming with C++, E. Balaguruswamy, TMH,6th Edition,2013.

CO#	COURSE OUTCOMES
CO1	Understand the fundamentals of C++ programming and apply concepts like function overloading and reference variables.
CO2	Implement classes, objects, and dynamic memory management to build modular and reusable programs.
CO3	Apply operator overloading, constructors, destructors, and inheritance to enhance code flexibility and reusability.
CO4	Perform file input and output operations using C++ stream classes and manage file pointers for data access.
CO5	Use exception handling constructs in C++ to manage runtime errors effectively

CO-PO-PSO Mapping:

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

Java Programming
[As per NEP Choice Based Credit System (CBCS)scheme]
(Effective from the academic year 2024-2025)
SEMESTER – VI

Course Code	**22CS652**	**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: This course will enable students

- To impart the basic concepts of data structures and algorithms.
- To understand concepts about searching and sorting techniques.
- To understand basic concepts about stacks, queues, lists, trees and graphs.
- To enable them to write algorithms for solving problems with the help of fundamental data structures

Modules	**Hours**
Module I	
Introduction to Java History of Java, Bytecode, Features of Java, Java Applications, Differences between C , C++ and Java, Principles of Object Oriented Programming, Building and Running Java Program, Java Tokens, Data Types, Variables, Operators, Classes and Objects: Class Fundamentals, Object creation; Type Conversion and Casting, Arrays, Access Specifiers.	**10**
Module II	
Control Statements, Methods, Constructors **Control Statements:** Java Selection Statements, Iterative Statements, Jump Statements; **Methods:** Method Definition, Method accessing, Method That Takes Parameters, Method overloading; **Constructors:** Constructor Definition, Parameterized Constructors, Constructor overloading;Using this keyword, Instance variable hiding; Java static keyword.	**10**
Module III	
Inheritance, Polymorphism Inheritance : Inheritance basics, Types of Inheritance, Simple and Multi-level Inheritance, Using super keyword, the call of Constructors ; **Polymorphism :** Run-time Polymorphism- Method overriding, Dynamic Method Dispatch; Using abstract class and final keyword	**10**
Module IV	
Packages, Interfaces, String Handling Packages : Defining Package, Access Protection, Importing Packages; **Interfaces** : Defining and Implementing Interfaces, Nested Interfaces, **String Handling** : String Constructors, String Operations: Concatenation, Conversion; CharacterExtraction	**10**
Module V	
Exception Handling, Multithreading Exception Handling : Exception Handling Fundamentals, Handling Exceptions using keywords,Nested try statements, **Multithreading :** Thread Creation, Creating Multiple Threads, Thread Priorities, Synchronization, Interthread Communication: Producer Consumer Problem	**10**

Question paper pattern

- The question paper will have TEN questions, Two questions from each module.
- The students will have to answer 5 full questions, selecting one full question from each module.
- Each full question carries 20 marks.

Each full question will have sub questions covering all topics under a module.

Textbooks:

1. Herbert Schildt, Java The Complete Reference, 7th Edition, Tata McGraw Hill

Reference Books:

1. Mahesh Bhave and Sunil Patekar, "Programming with Java", First Edition, Pearson Education,2008, ISBN:9788131720806.
2. Rajkumar Buyya,S Thamarasi selvi, xingchen chu, Object oriented Programming with java, Tata McGraw Hill education private limited.
3. E Balagurusamy, Programming with Java A primer, Tata McGraw Hill companies. Anita Seth and B L Juneja, JAVA One step Ahead, Oxford University Press, 2017..

CO#	COURSE OUTCOMES
CO1	Differentiate structured programming and object oriented programming and know the concepts of classes, objects, members of a class.
CO2	Apply the concept of classes, control statements, constructor and develop Simple Java Programs.
CO3	Develop Simple Java Programs using inheritance and Polymorphism
CO4	Apply the concepts of packages, interfaces, and string handling in Java to design and implement solutions for real-world programming problems
CO5	Illustrates the concepts of Multi Threaded Programming and Exception handling

CO-PO-PSO Mapping:

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

Software Engineering Laboratory [As per NEP Choice Based Credit System (CBCS)scheme] (Effective from the academic year 2024-2025) SEMESTER – VI			
Course Code	**22CSL66**	**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:

- To understand the software engineering methodologies for project development
- To design and implement complex software systems
- To apply Agile software development methodologies in real-world projects
- To integrate automated testing into the development lifecycle
- To develop secure, scalable, and efficient applications
- To deploy, maintain, and scale software systems.

Experiments:

1. Online Course Registration System, and Student marks analyzing system.
2. Health Care (Expert system to prescribe medicines for given symptoms, Remote Diagnostics, Patient/Hospital Management System).
3. Banking System (ATM/Net Banking,).
4. E-Commerce (Various online shopping portals like Flipkart /Amazon/Myntra) system.
5. Logistics (Postal /Courier) System.
6. Tourism Management System.
7. Social Networking System.
8. Customer Support System.
9. Booking/Ticketing System.
10. Develop basic calculators using Agile Method. Test each feature using automated unit tests

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	Identify and formulate research or societal problems
CO2	Design and develop solution to the problem
CO3	Analyse for optimization solutions
CO4	Implement and execute
CO5	Write effective technical report and demonstrate through presentation

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 and Machine Learning Laboratory [As per NEP Choice Based Credit System (CBCS)scheme] (Effective from the academic year 2024-2025) SEMESTER – VI			
Course Code	**22CSL67**	**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:

- Able to understand the concepts of Searching algorithms and implementing them with a code
- Able to understand the ANN and back propagation algorithms by experimenting on datasets
- Able to understand the implementation of plotting graphs by importing certain packages

No.	List of Experiments:
1.	Implement AO* search algorithm.
2.	Implement and Demonstrate the Travelling Salesman Problem
3.	Write a program to iimplement Alpha-Beta Pruning using Python
4.	Write a program to implement Tic-Tac-Toe game using python
5.	write a program to implement Hill Climbing algorithm
6.	Implement Resolution principle on FOPL related Problem.
7.	Build an Artificial Neural Network by implementing the Back propagation algorithm and test the same using appropriate data sets.
8.	Write a Program to implement 8-Queens Problem using Python.
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 50 marks

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

Full Stack Development Laboratory
[As per NEP Choice Based Credit System (CBCS)scheme]
(Effective from the academic year 2024-2025)
SEMESTER – VI

Course Code	**22CSL681**	**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:

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

No.	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 Many to Many 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.

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

Cryptography and Network Security Laboratory [As per NEP Choice Based Credit System (CBCS)scheme] (Effective from the academic year 2024-2025) SEMESTER – VI			
Course Code	**22CSL682**	**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:

- Able to understand the concept of XOR , AND operators experimenting with the character strings.
- Able to know the algorithms of encryption and decryption and implementing them
- Able to understand the security factors such as message digests algorithms.

No.	List of Experiments:
1.	Write a C program that contains a string (char pointer) with a value 'Hello world'. The program should XOR each character in this string with 0 and displays the result.
2.	Write a C program that contains a string (char pointer) with a value 'Hello world'. The program should AND or and XOR each character in this string with 127 and display the result.
3.	Write a Java program to perform encryption and decryption using the following algorithms a. Ceaser cipher b. Substitution cipher c. Hill Cipher
4.	Write a C/JAVA program to implement the DES algorithm logic.
5.	Write a C/JAVA program to implement the Blowfish algorithm logic.
6.	Write a C/JAVA program to implement the Rijndael algorithm logic.
7.	Write the RC4 logic in Java Using Java cryptography; encrypt the text "Hello world" using Blowfish. Create your own key using Java key tool.
8.	Implement the Diffie-Hellman Key Exchange mechanism using HTML and JavaScript.
9.	Calculate the message digest of a text using the SHA-1 algorithm in JAVA.
10.	Calculate the message digest of a text using the MD5 algorithm in JAVA.

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.	Utilize classical and modern cryptographic algorithms to implement secure data encryption, decryption, and key exchange mechanisms.
CO2	Apply cryptographic hash functions and bitwise operations for secure data processing and integrity verification.
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	-	-	-	-	-	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

Natural Language Processing Laboratory
[As per NEP Choice Based Credit System (CBCS)scheme]
(Effective from the academic year 2024-2025)
SEMESTER – VI

Course Code	**22ADL683**	**CIE Marks**	**50**
Number of Lecture Hours/Week	**02**	**SEE Marks**	**50**
Total Number of LectureHours	**30**	**Exam Hours**	**03**

CREDITS – 01

Course Objectives:

- Be able to discuss the current and likely future performance of several NLP applications.
- Be able to describe briefly a fundamental technique for processing language for several subtasks, such as morphological processing.
- Implement parsing, word sense disambiguation etc.
- Understand how these techniques draw on and relate to other areas of computer science.
- Understand the basic principles of designing and running an NLP experiment.

No.	List of Experiments:
1.	a. Write a python program to perform tokenization by word and sentence using nltk. b. Write a python program to eliminate stop words using nltk. c. Write a python program to perform stemming using nltk.
2.	a. Write a python program to perform Parts of Speech tagging using nltk. b. Write a python program to perform lemmatization using nltk.
3.	a. Write a python program for chunking using nltk. b. Write a python program to perform Named Entity Recognition using nltk.
4.	a. Write a python program to find Term Frequency and Inverse Document Frequency (TF-IDF). b. Write a python program for CYK parsing **(Cocke- Younger-Kasami Parsing) or Chart Parsing.**
5.	a. Write a python program to find all unigrams, bigrams and trigrams present in the given corpus. b. Write a python program to find the probability of the given statement "This is my cat" by taking the an exmple corpus into consideration.
6.	Use the Stanford named Entity recognizer to extract entities from the documents. Use it programmatically and output for each document which named entities it contains and of which type.
7.	Choose any corpus available on the internet freely. For the corpus, for each document, count how many times each stop word occurs and find out which are the most frequently occurring stop words. Further, calculate the term frequency and inverse document frequency as The motivation behind this is basically to find out how important a document is to a given query. For e.g.: If the query is say: "The brown crow". "The" is less important. "Brown" and "crow" are relatively more important. Since "the" is a more common word, its tf will be high. Hence we multiply it by idf, by knowing how common it is to reduce its weight.
8.	Write the python code to perform sentiment analysis using NLP.
9.	Write the python code to develop Spam Filter using NLP
10.	Write the python code to detect Fake News using NLP

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 and apply foundational NLP techniques such as tokenization, stemming, lemmatization, POS tagging, and chunking using Python and NLTK.
CO2	Design and implement NLP applications including sentiment analysis, spam filtering, fake news detection, and information extraction using machine learning and text processing tools.
CO3	Debug and troubleshoot issues effectively.
CO4	Analyze the data and interpret the results.
CO5	Prepare a well-organizedlaboratory 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

Mobile Application Development Laboratory
[As per NEP Choice Based Credit System (CBCS)scheme]
(Effective from the academic year 2024-2025)
SEMESTER VI

Course Code	**22CSL684**	**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:

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

PART- A

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 TextView 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. Devlop a simple application with one EditText 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 dialer 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

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 MkSDcard. 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 farward, 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 bannr. On pressing the Start task button, the banner message shuld stop. Let the banner messaga 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 EditText 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)

Where

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 EditText to read the Principal amount, Down payment, Interest rate, loan term(in months) and a button named as " CalculateMonthly EMI". On click of this button, the result should be shown in a TextView. Also, calculate the EMI by varying the Loan term and interest rate values.

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 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	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 2024-25) SEMESTER – VI			
Course Code	**22PRJ69**	**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 and Intellectual Property Rights [As per NEP Choice Based Credit System (CBCS)scheme] (Effective from the academic year 2024-2025) SEMESTER – VI			
Course Code	**22HSM610A**	**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:

- To Understand the knowledge on basics of research and its types.
- To Learn the concept of Literature Review, Technical Reading, Attributions and Citations.
- To learn Ethics in Engineering Research.
- To Discuss the concepts of Intellectual Property Rights in engineering

Modules	
Module I	**Hours**
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 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	**04**

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**

Textbooks:

1. Dipankar Deb • Rajeeb Dey, Valentina E. Balas "Engineering Research Methodology", ISSN 1868-4394 ISSN 1868-4408 (electronic), Intelligent Systems Reference Library, ISBN 978-981-13- 2946-3 ISBN 978-981-13-2947-0 (eBook), https://doi.org/10.1007/978-981-13-2947-0

2. Intellectual Property A Primer for Academia by Prof. Rupinder Tewari Ms. Mamta Bhardwa

Reference Book:

1. David V. Thiel "Research Methods for Engineers" Cambridge University Press, 978-1-107-03488-4 – 2. Intellectual Property Rights by N.K.Acharya Asia Law House 6th Edition. ISBN: 978-93- 81849-30-9

Question paper pattern

- The question paper will have TEN questions, Two questions from each module.
- The students will have to answer 5 full questions, selecting one full question from each module.
- Each full question carries 20 marks.
- Each full question will have sub questions covering all topics under a module.

CO#	COURSE OUTCOMES
CO1	Students get exposure to the research objectives, motivations, ethics, to recognize and define the research problem and steps involved in finding the solutions.
CO2	Steps involved in the Process of conducting a Literature Review and engaging in Technical Reading and also regarding the publications and citations provided for the work performed.
CO3	Gain knowledge of the fundamental principles of patent laws and the drafting process. Understand the key procedures involved in patent filing and documentation.
CO4	Gain knowledge of copyright laws and their relevance to course content, creative works, and designs.
CO5	Learn the fundamental principles of design rights and their legal protection. Understand how they safeguard creative and innovative designs.

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	-

Tableau
[As per NEP Choice Based Credit System (CBCS)scheme]
(Effective from the academic year 2024-2025)
SEMESTER – VI

Course Code	**22AAD611A**	**CIE Marks**	**50**
Number of Lecture Hours/Week	**02**	**SEE Marks**	**50**
Total Number of LectureHours	**30**	**Exam Hours**	**03**

CREDITS – 01

Course Objectives:

- Develop practical skills in data visualization using Tableau software.
- Understand the Tableau interface and its basic functionalities.
- Learn about creating different types of advanced charts.
- Learn to integrate multiple views into a single interactive dashboard.
- Learn how story telling in data visualization using Tableau.

List of Experiments:

1. Introduction to Tableau and installation.
2. Data connections in tableau interface
3. Calculations using Tableau.
4. Organizing and Simplifying Data
5. Preparing data for visualization in Tableau.
6. Creating Dashboard
7. Building Different Chart Types.
8. Advanced Chart Types
10. Storytelling in Tableau.

Textbooks:

1. "Learning Tableau" by Joshua N. Milligan: Great for beginners, it covers how to create data visualizations, dashboards, and more.
2. "Tableau Your Data": by Dan Murray: Ideal for people who already have some data analysis background and want to take their Tableau skills to the next level.

Online Resources:

1. https://youtube.com/playlist?list=PLO9LeSU_vHCU_DHaLzEvsLxFdmB3Qcao_&si=AV8QQMRCCZrmF5Xs
2. https://youtube.com/playlist?list=PLWPirh4EWFpGXTBu8ldLZGJCUeTMBpJFK&si=tzJVKC3r2DfEnuSx

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.	Utilize Tableau to connect, import, and prepare various data sources for effective data visualization..
CO2	Apply Tableau features to create interactive worksheets, dashboards, and stories for insightful data communication.
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	-	-	-	-	-	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

Devops
[As per NEP Choice Based Credit System (CBCS)scheme]
(Effective from the academic year 2024-2025)
SEMESTER – VI

Course Code	**22ACS611B**	**CIE Marks**	**50**
Number of Lecture Hours/Week	**02**	**SEE Marks**	**50**
Total Number of LectureHours	**30**	**Exam Hours**	**03**

CREDITS – 01

Course Objectives:

- Understand the agile relationship between development and IT operations.
- Study the different Version control tools like Git and Git hub.
- Enable students to implement DevOps practices and tools for modern software development.
- Enable students to automate system updates and manage the full DevOps lifecycle for efficient, reliable, and scalable software delivery.

List of Experiments:

1. Write code for a simple user registration form for an event.
2. Explore Git and GitHub commands.
3. Practice Source code management on GitHub. Experiment with the source code written in exercise 1.
4. Jenkins installation and setup, explore the environment.
5. Demonstrate continuous integration and development using Jenkins.
6. Explore Docker commands for content management.
7. Develop a simple containerized application using Docker.
8. Integrate Kubernetes and Docker
9. Automate the process of running containerized application developed in exercise 7 using Kubernetes.
10. Install and Explore Selenium for automated testing.
11. Write a simple program in JavaScript and perform testing using Selenium.
12. Develop test cases for the above containerized application using selenium.

Textbooks:

1. Joakim Verona. Practical Devops, Second Edition. Ingram short title; 2nd edition (2018). ISBN-10: 1788392574
2. Deepak Gaikwad, Viral Thakkar. DevOps Tools from Practitioner's Viewpoint. Wiley publications. ISBN: 9788126579952

Online Resources:

- Free Code Camp DevOps Course: https://www.freecodecamp.org/news/devops-tutorial
- Docker Official Docs - https://docs.docker.com

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.	Utilize Devops Tools to Automate Software Development, Integration, And Deployment Processes.
CO2	Apply Automated Testing Techniques to Ensure The Reliability And Functionality Of Web Applications In A Devops Pipeline.
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	-	-	-	-	-	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