# CURRICULUM AND DETAILED SYLLABI

FOR

## B. TECH DEGREE PROGRAMME

IN

## COMPUTER SCIENCE AND ENGINEERING

## SEMESTER VII & VIII

## 2023 SCHEME

## (AUTONOMOUS)

**MAR BASELIOS COLLEGE OF ENGINEERING AND TECHNOLOGY**
(Approved by AICTE, Autonomous Institution Affiliated to APJ Abdul Kalam Technological University)
MAR IVANIOS VIDYANAGAR, NALANCHIRA, THIRUVANANTHAPURAM – 695015, KERALA.
Phone: 0471 2545866
Fax: 0471 2545869
Web: www.mbcet.ac.in
email: hodcs@mbcet.ac.in

## MAR BASELIOS COLLEGE OF ENGINEERING AND TECHNOLOGY

### Vision and Mission of the Institution

**Vision:**
To be an Institution moulding globally competent professionals as epitomes of Noble Values.

**Mission:**
To transform the Youth as technically competent, ethically sound and socially committed professionals, by providing a vibrant learning ambience for the welfare of humanity.

## DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

### Vision and Mission of the Department

**Vision:**
To be a Centre of Excellence in Computer Science and Engineering providing quality education and research for the betterment of the society.

**Mission:**
To impart sound knowledge in theoretical and applied foundations of Computer Science and Engineering, and to train the students to solve real life issues to effectively define and shape life.

### PROGRAMME EDUCATIONAL OBJECTIVES (PEOs)

1. Graduates will be successful professionals in Industries of core or interdisciplinary nature or entrepreneurs, demonstrating effective leadership and excellent team work.
2. Graduates will expand the horizon of knowledge through higher education orresearch, leading to self-directed professional development.
3. Graduates will demonstrate professional attitude and ethics while providing solutions in societal and environmental contexts.

## PROGRAMME OUTCOMES (POs)

Engineering graduates will be able to:

1. **Engineering knowledge**: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
2. **Problem analysis**: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.
3. **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 the public health and safety, and the cultural, societal, and environmental considerations.
4. **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.
5. **Modern tool usage**: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modelling to complex engineering activities with an understanding of the limitations.
6. **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.
7. **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.
8. **Ethics**: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
9. **Individual and team work**: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.
10. **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 reports and design documentation, make effective presentations, and give and receive clear instructions.
11. **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.
12. **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.

## PROGRAMME SPECIFIC OUTCOMES (PSOs)

. Engineering Graduates will have the ability to:

1. Apply Algorithmic Principles, Programming Skills and Software Engineering Principles to design, develop and evaluate Software Systems of varying complexities.
2. Apply knowledge of System Integration to design and implement computer-based systems.
3. Solve real world and socially relevant problems with the knowledge in recent and advanced Computing Technologies.

# DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

*For the students admitted from 2023*

## Scheduling of Courses

### i) Knowledge Segments and Credits

Every course of B. Tech Programme is placed in one of the nine categories as listed in the following table. No semester shall have more than six lecture-based courses and two laboratory courses, and/or drawing/seminar/project courses in the curriculum.

| Sl. No. | Category | Category Code | 2020(REVISED) | 2023(REVISED) |
|---|---|---|---|---|
| 1 | Humanities and Social Sciences including Management Courses | HSC | 5 | 6 |
| 2 | Basic Science Courses | BSC | 26 | 26 |
| 3 | Engineering Science Courses | ESC | 22 | 24 |
| 4 | Programme Core Courses, Comprehensive Course Work and Viva Voce | PCC | 79 | 72 |
| 5 | Programme Elective Courses | PEC | 15 | 18 |
| 6 | Institute Elective Courses | OEC/IEC | 3 | 6 |
| 7 | Project Work and Seminar | PWS | 10 | 15 |
| 8 | Mandatory Non-credit Courses (P/F) with Grade | MNC | --- | --- |
| 9 | Mandatory Student Activities (P/F) | MSA | 2 | 3 |
| **Total Mandatory Credits** | | | **162** | **170** |
| | Value Added Courses (Optional) – Honours/Minor | VAC | 15 | 15 |

### ii) Semester-wise Credit Distribution

| ***Semester*** | **I** | **II** | **III** | **IV** | **V** | **VI** | **VII** | **VIII** | ***Total Credits*** |
|---|---|---|---|---|---|---|---|---|---|
| *Credits for Courses* | 19 | 21 | 23 | 22 | 21 | 26 | 21 | 14 | **167** |
| | 40 | | 45 | | 47 | | 35 | | **167** |
| *Credits for Activities* | 3 | | | | | | | | **3** |
| *Total Credits* | | | | | | | | | **170** |
| *Value Added Courses (Optional) – Honours / Minor* | | | | | | | | | **15** |
| ***Total Credits*** | | | | | | | | | **185** |

| SEMESTER VII | | | | | | | |
|---|---|---|---|---|---|---|---|
| **Slot** | **Category Code** | **Course Number** | **Courses** | **L-T-P-J** | **SS Hours** | **Hours** | **Credit** |
| A | PCC | 23CSB40A | Compiler Design | 3-1-2-0 | 6 | 6 | 5 |
| B | PCC | 23CSB40B | Web Technology | 3-0-2-0 | 5.5 | 5 | 4 |
| C | PEC | 23CSL43X | Programme Elective Course 3 | 2-1-0-0 | 3.5 | 3 | 3 |
| E | IEC | 23IEL42X | Institute Elective 2 | 3-0-0-0 | 4.5 | 3 | 3 |
| T | PWS | 23CSV48A | Comprehensive Viva Voce | 0-0-2-0 | 1 | 2 | 1 |
| U | PWS | 23CSJ48A | Project | 0-0-10-0 | 10 | 10 | 5 |
| | | 23CSI48A | Internship* | | | | |
| M/H | VAC | | Minor/Honours Course | 0-0-6-0/ 3-0-0-0 | 3/4.5 | 6/3 | 3 |
| **TOTAL** | | | | | **30.5/33 .5/ 35** | **29/35/32** | **21/24** |

| SEMESTER VIII | | | | | | | |
|---|---|---|---|---|---|---|---|
| **Slot** | **Category Code** | **Course Number** | **Courses** | **L-T-P-J** | **SS Hours** | **Hours** | **Credit** |
| A | PEC | 23CSL44X | Programme Elective Course 4 | 2-1-0-0 | 3.5 | 3 | 3 |
| B | PEC | 23CSL45X | Programme Elective Course 5 | 2-1-0-0 | 3.5 | 3 | 3 |
| C | PEC | 23CSL46X | Programme Elective Course 6 | 2-1-0-0 | 3.5 | 3 | 3 |
| U | PWS | 23CSJ48B | Project | 0-0-10-0 | 10 | 10 | 5 |
| | | 23CSI48A | Internship* | | | | |
| M/H | VAC | | Minor/Honours Course | 0-0-6-0 | 3 | 6 | 3 |
| **TOTAL** | | | | | **20.5/23.5** | **19/25** | **14/17** |

*Students can opt for Internship either in S7 or S8. However, in S7, the internship can be permitted only if there are no pending Programme/Course requirements in the semester, that need to be completed in College in the offline mode, such as laboratory sessions.

**PROGRAMME ELECTIVE I**

| Slot | Category Code | Course Number | Courses | L-T-P-J | Hours | Credit |
|---|---|---|---|---|---|---|
| D | PEC | 23CSL31A | Parallel Computer Architecture | 2-1-0-0 | 3 | 3 |
| | | 23CSL31B | Introduction to Data Science | 2-1-0-0 | 3 | 3 |
| | | 23CSL31C | Computer Graphics and Multimedia | 2-1-0-0 | 3 | 3 |
| | | 23CSL31D | Fundamentals of Cryptography | 2-1-0-0 | 3 | 3 |

**PROGRAMME ELECTIVE II**

| Slot | Category Code | Course Number | Courses | L-T-P-J | Hours | Credit |
|---|---|---|---|---|---|---|
| D | PEC | 23CSL32A | Cloud Computing | 2-1-0-0 | 3 | 3 |
| | | 23CSL32B | Applied Data Science Using Python | 2-1-0-0 | 3 | 3 |
| | | 23CSL32C | Robotics and Intelligent Systems | 2-1-0-0 | 3 | 3 |
| | | 23CSL32D | Virtual and Augmented Reality Systems | 2-1-0-0 | 3 | 3 |

**PROGRAMME ELECTIVE III**

| Slot | Category Code | Course Number | Courses | L-T-P-J | Hours | Credit |
|---|---|---|---|---|---|---|
| C | PEC | 23CSL43A | Internet of Things | 2-1-0-0 | 3 | 3 |
| | | 23CSL43B | Computational Complexity | 2-1-0-0 | 3 | 3 |
| | | 23CSL43C | Deep Learning | 2-1-0-0 | 3 | 3 |
| | | 23CSL43D | Cloud Security | 2-1-0-0 | 3 | 3 |
| | | 23CSL43E | Human Computer Interaction | 2-1-0-0 | 3 | 3 |
| | | 23CTL43A | Computer Vision | 2-1-0-0 | 3 | 3 |
| | | 23CTL43C | AI for Cyber Security | 2-1-0-0 | 3 | 3 |

**PROGRAMME ELECTIVE IV**

| Slot | Category Code | Course Number | Courses | L-T-P-J | Hours | Credit |
|---|---|---|---|---|---|---|
| A | PEC | 23CSL44A | Virtualization | 2-1-0-0 | 3 | 3 |
| | | 23CSL44B | Approximation Algorithms | 2-1-0-0 | 3 | 3 |
| | | 23CSL44C | Natural Language Processing | 2-1-0-0 | 3 | 3 |
| | | 23CSL44D | Digital Forensics | 2-1-0-0 | 3 | 3 |

**PROGRAMME ELECTIVE V**

| Slot | Category Code | Course Number | Courses | L-T-P-J | Hours | Credit |
|---|---|---|---|---|---|---|
| B | PEC | 23CSL45A | Algorithmic Game Theory | 2-1-0-0 | 3 | 3 |
| | | 23CSL45B | Large Language Models | 2-1-0-0 | 3 | 3 |
| | | 23CSL45C | Hardware Security | 2-1-0-0 | 3 | 3 |
| | | 23CSL45D | Social Media Analysis | 2-1-0-0 | 3 | 3 |
| | | 23CSL45E | Quantum Computing and Programming | 2-1-0-0 | 3 | 3 |

**PROGRAMME ELECTIVE VI**

| Slot | Category Code | Course Number | Courses | L-T-P-J | Hours | Credit |
|---|---|---|---|---|---|---|
| C | PEC | 23CSL46A | Parallel Algorithms | 2-1-0-0 | 3 | 3 |
| | | 23CSL46B | Big Data Analytics | 2-1-0-0 | 3 | 3 |
| | | 23CSL46C | Block Chain and Crypto Currencies | 2-1-0-0 | 3 | 3 |
| | | 23CSL46D | Vibe Coding | 2-1-0-0 | 3 | 3 |

**INSTITUTE ELECTIVE I**

| Slot | Category Code | Course Number | Courses | L-T-P-J | Hours | Credit |
|---|---|---|---|---|---|---|
| E | IEC | 23IEL31E | Data Science for Engineers | 3-0-0-0 | 3 | 3 |
| | | 23IEL31F | Introduction to Mobile Application | 3-0-0-0 | 3 | 3 |
| | | 23IEL31G | Introduction to Cyber Security and Ethical Hacking | 3-0-0-0 | 3 | 3 |
| | | 23IEL31H | Digital Marketing and E-commerce | 3-0-0-0 | 3 | 3 |

**INSTITUTE ELECTIVE II**

| Slot | Category Code | Course Number | Courses | L-T-P-J | Hours | Credit |
|---|---|---|---|---|---|---|
| E | IEC | 23IEL42E | Responsible AI | 3-0-0-0 | 3 | 3 |
| | | 23IEL42F | Prompt Engineering | 3-0-0-0 | 3 | 3 |
| | | 23IEL42G | Business Intelligence and Analytics | 3-0-0-0 | 3 | 3 |
| | | 23IEL42H | Game Development | 3-0-0-0 | 3 | 3 |

## MINOR

| Semester | BASKET I Specialization: SOFTWARE ENGINEERING | | | | BASKET II Specialization: MACHINE LEARNING | | | | BASKET III Specialization: NETWORKING | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Course Number | Course | L-T-P-J | Credit | Course Number | Course | L-T-P-J | Credit | Course Number | Course | L-T-P-J | Credit |
| S3 | 23CSL2MA | Object Oriented Programming | 3-0-0-0 | 3 | 23CSL2MC | Mathematics for Machine Learning | 3-0-0-0 | 3 | 23CSL2ME | Data Communication | 3-0-0-0 | 3 |
| S4 | 23CSL2MB | Programming Methodologies | 3-0-0-0 | 3 | 23CSL2MD | Concepts in Machine Learning | 3-0-0-0 | 3 | 23CSL2MF | Introduction to Computer Networks | 3-0-0-0 | 3 |
| S5 | 23CSL3MA | Concepts in Software Engineering | 3-0-0-0 | 3 | 23CSL3MC | Concepts in Deep Learning | 3-0-0-0 | 3 | 23CSL3ME | Client Server Systems | 3-0-0-0 | 3 |
| S6 | 23CSL3MB | Introduction to Software Testing | 3-0-0-0 | 3 | 23CSL3MD | Reinforcement Learning | 3-0-0-0 | 3 | 23CSL3MF | Wireless Networks and IoT Applications | 3-0-0-0 | 3 |
| S7/ S8 | 23CSJ4MA | Mini Project | 0-0-6-0 | 3 | 23CSJ4MC | Mini Project | 0-0-6-0 | 3 | 23CSJ4ME | Mini Project | 0-0-6-0 | 3 |

| Semester | Basket IV Specialization: Data Science | | | | Basket V Specialization: Network Security | | | |
|---|---|---|---|---|---|---|---|---|
| | Course Number | Course | L-T-P-J | Credit | Course Number | Course | L-T-P-J | Credit |
| **S3** | 23CSL2MG | Statistics for Data Science and Time Forecasting | 3-0-0-0 | 3 | 23CSL2MI | Basics of Computer Systems | 3-0-0-0 | 3 |
| **S4** | 23CSL2MH | Data Visualization & ML | 3-0-0-0 | 3 | 23CSL2MJ | Cyber Security | 3-0-0-0 | 3 |
| **S5** | 23CSL3MG | Natural Language Processing | 3-0-0-0 | 3 | 23CSL3MI | Introduction to Blockchain technologies | 3-0-0-0 | 3 |
| **S6** | 23CSL3MH | Deep Learning | 3-0-0-0 | 3 | 23CSL3MJ | Privacy and security in IoT | 3-0-0-0 | 3 |
| **S7/S8** | 23CSJ4MG | Mini Project | 0-0-6-0 | 3 | 23CSJ4MI | Mini Project | 0-0-6-0 | 3 |

## HONOURS

| Semester | BASKET I Specialization: SECURITY IN COMPUTING | | | | BASKET II Specialization: MACHINE LEARNING | | | | BASKET III Specialization: FORMAL METHODS | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Course Number | Course | L-T-P-J | Credit | Course Number | Course | L-T-P-J | Credit | Course Number | Course | L-T-P-J | Credit |
| **S4** | 23CSL 2HB | Number Theory | 3-0-0-0 | 3 | 23CSL 2HD | Computational Fundamentals of Machine Learning | 3-0-0-0 | 3 | 23CSL 2HF | Principles of Program Analysis and Verification | 3-0-0-0 | 3 |
| **S5** | 23CSL 3HA | Cryptographic Algorithms | 3-0-0-0 | 3 | 23CSL 3HC | Neural Networks and Deep Learning | 3-0-0-0 | 3 | 23CSL 3HE | Principles of Model Checking | 3-0-0-0 | 3 |
| **S6** | 23CSL 3HB | Network Security | 3-0-0-0 | 3 | 23CSL 3HD | Advanced Topics in Machine Learning | 3-0-0-0 | 3 | 23CSL 3HF | Theory of Computability and Complexity | 3-0-0-0 | 3 |
| **S7** | 23CSL 4HA | Cyber Forensics | 3-0-0-0 | 3 | 23CSL 4HC | Reinforcement Learning | 3-0-0-0 | 3 | 23CSL 4HE | Logic for Computer Science | 3-0-0-0 | 3 |
| **S8** | 23CSJ 4HB | Mini Project | 0-0-6-0 | 3 | 23CSJ4 HD | Mini Project | 0-0-6-0 | 3 | 23CSJ 4HF | Mini Project | 0-0-6-0 | 3 |

# SEMESTER VII

| Course Code | Course Name | Category | L | T | P | J | Credit | Year of Introduction |
|---|---|---|---|---|---|---|---|---|
| 23CSB40A | COMPILER DESIGN | PCC | 3 | 1 | 2 | 0 | 5 | 2023 |

**i. COURSE OVERVIEW**:

This course aims to provide a comprehensive understanding of the compiler construction process through its various phases such as lexical analysis, parsing, semantic analysis, code generation, optimization and target code generation.

**ii. COURSE OUTCOMES**

After the completion of the course, the student will be able to:

| Course Outcomes | Description | Level |
|---|---|---|
| CO1 | Explain the phases of a compiler | Understand |
| CO2 | Model a lexical analyser | Apply |
| CO3 | Construction of top-down and bottom-up parsers | Apply |
| CO4 | Build Syntax Directed Translation and intermediate representations | Apply |
| CO5 | Implement a code optimizer and code generator. | Apply |

**iii. SYLLABUS**

Review of Regular and Context free Grammar, left recursion, Elimination of Left recursion, Left factoring, Elimination of Left factoring.

Phases of Compiler, Front end of compiler, Backend of compiler, Bootstrapping, Compiler construction Tools.

Lexical analysis, Role of Lexical Analyser, Input Buffering, Specification of Tokens, Recognition of Tokens.

Role of the Syntax Analyser, Top-Down Parsing - Recursive Descent parsing, Predictive Parsing, LL(1) Grammars, Bottom-Up Parsing-Shift Reduce parsing, Operator precedence parsing, LR Parsing.

Syntax directed translation, Run-Time Environments, Intermediate Code Generation, Code Optimization, Code Generation.

**iv. a) TEXT BOOKS**

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

**b) REFERENCES**

1. D.M.Dhamdhere, System Programming and Operating Systems, Tata McGraw Hill & Company, 1996.
2. Kenneth C. Louden, Compiler Construction – Principles and Practice, Cengage Learning Indian Edition, 2006.
3. Tremblay and Sorenson, The Theory and Practice of Compiler Writing, Tata McGraw Hill & Company,1984.

**v. COURSE PLAN**

| Module | Contents | No. of hours |
|---|---|---|
| **I** | Analysis of the source program - Analysis and synthesis phases, Phases of a compiler. Bootstrapping. Lexical Analysis - Role of Lexical Analyser, Input Buffering- Buffer pairs, input buffering with sentinels, Specification of Tokens, Recognition of Tokens. Error handling in lexical analyser.<br>Lexical analysis using Lex tool. | 10 |
| **II** | Role of the Syntax Analyser – Syntax error handling. Error recovery strategies. Review of Context Free Grammars, Basic parsing approaches - Eliminating left recursion, left factoring.<br>Top-Down Parsing - Recursive Descent parsing and its algorithm, First and follow, Predictive Parsing, Construction of predictive parsing table, non-recursive predictive parsing. LL(1) Grammars. | 12 |

| | | |
|---|---|---|
| | The Parser Generator Yacc. | |
| **III** | Bottom-Up parsing – Handle Pruning. Shift Reduce parsing, conflicts. Introduction to LR parsing – LR(0) automaton, LR parsing, Constructing SLR, Canonical LR(1) items. LALR and canonical LR parsing tables. | 13 |
| **IV** | Syntax directed translation - Syntax directed definitions, S-attributed definitions, L-attributed definitions, Evaluation order of SDD's – Dependency graphs. Bottom-up evaluation of S-attributed definitions. Application of SDD- Construction of parse tree,<br>Intermediate Code Generation - Variants of Syntax Trees Directed Acyclic Graphs for Expressions, Constructing DAG's, Three-Address code, Quadruples, Triples. | 13 |
| **V** | Code Optimization - Machine dependent and machine independent optimizations - Principal sources of optimization.<br>Basic blocks and flow graphs, Optimization of basic block, DAG representation, finding local common expression, Dead code elimination.<br>Code generation - Issues in the design of a code generator, Target Language, A simple code generator. | 12 |
| | **Total Hours** | **60** |

**COMPILER DESIGN LAB**

| No | Experiment | Hours |
|---|---|---|
| **1** | Design a DFA for a language. | 6 |
| **2** | Implementing a lexical analyser using Lex Tool | 3 |
| **3** | Implement a predictive parser | 6 |
| **4** | Yacc | 3 |
| **5** | Implement a shift reduce parser. | 6 |

| | | |
|---|---|---|
| **6** | Design a three-address code generator and perform machine independent optimization. | 6 |
| | **Total** | **30** |

**vii. ASSESSMENT PATTERN**

**Continuous Assessment: End Semester Examination – 60: 40**

| **Continuous Assessment** | | |
|---|---|---|
| Attendance | : | 5 marks |
| Assignments | : | 15 marks |
| Assessment through Tests | : | 20 marks |
| Lab Work | : | 10 marks |
| Lab Exam | : | 10 marks |
| Total Continuous Assessment | : | 60 marks |
| **End Semester Examination** | **:** | **40 marks** |
| **TOTAL** | **:** | **100 marks** |

**viii. CONTINUOUS ASSESSMENT TEST**

- No. of Tests: 02
- Maximum Marks: 30
- Test Duration: 1 ½ hours
- Topics: 2 ½ modules

**ix. END SEMESTER EXAMINATION**

- Maximum Marks: 40
- Exam Duration: 2 hours

| Course Code | Course Name | Category | L | T | P | J | Credit | Year of Introduction |
|---|---|---|---|---|---|---|---|---|
| **23CSB40B** | **WEB TECHNOLOGY** | **PCC** | **3** | **0** | **2** | **0** | **4** | **2023** |

## i. COURSE OVERVIEW

This course provides a comprehensive foundation in full-stack web development through a lab-integrated approach. It covers frontend technologies including HTML5, CSS3, JavaScript, and React for building responsive and interactive single-page applications. The course extends to backend development using Node.js, Express.js, and MongoDB for designing RESTful APIs and database-driven applications. Emphasis is placed on integration of frontend and backend systems, secure coding practices, authentication mechanisms, performance optimization, DevOps workflows, and cloud deployment. Through continuous hands-on laboratory activities, students develop scalable, secure, and production-ready web applications using the MERN stack.

## ii. COURSE OUTCOMES

After the completion of the course, the student will be able to:

| Course Outcomes | Description | Level |
|---|---|---|
| CO1 | Develop semantic, responsive, and accessible web interfaces using HTML5 and CSS3. | Apply |
| CO2 | Implement interactive client-side applications using JavaScript, DOM manipulation, and modern ES6+ features. | Apply |
| CO3 | Design and develop single-page applications using React, including state management, routing, and API integration. | Apply |
| CO4 | Build and integrate backend services using Node.js, Express.js, and MongoDB to create RESTful web applications. | Apply |

| | | |
|---|---|---|
| CO5 | Apply security best practices, performance optimization techniques, version control, CI/CD workflows, and cloud deployment strategies for scalable web applications. | Apply |

**iii. SYLLABUS**

Web fundamentals including Internet concepts, HTTP/HTTPS, and client–server architecture; frontend development using HTML5, CSS3, responsive design, Flexbox, and Grid; JavaScript programming with ES6+, DOM manipulation, event handling, and browser storage; single-page application development using React including components, props, state, hooks, routing, and Context API; API integration using Fetch, Axios, and JSON; backend development with Node.js and Express.js including REST API design; database management using MongoDB and Mongoose with CRUD operations; authentication and security concepts including cookies, sessions, JWT, and OWASP basics; version control and DevOps practices using Git and CI/CD; cloud deployment, performance optimization, and monitoring tools.

**iv (a) TEXTBOOKS**

1 Philip Ackermann, Full stack web development – The comprehensive guide, SAP Press, 2022.

2 Ethan Brown. (2019). Web development with Node and Express. O'Reilly Media,2019

**(b) REFERENCES**

1 Marijn Haverbeke, Eloquent JavaScript, 4th ed., No Starch Press, 2024.

2 Carl Rippon, Beginning React with Hooks, Apress, 2020.

3 Kyle Simpson, You Don't Know JS Yet: Get Started. 2nd ed., Independently published, 2023.

**v. COURSE PLAN**

| Module | Contents | Hours |
|---|---|---|
| **I** | **Frontend Foundations** – Internet, WWW, URL, HTTP/HTTPS, Client–Server Model, Static vs Dynamic Websites, SPA vs MPA (Intro), HTML5 Structure (DOCTYPE, Head, Body), Semantic Elements, Hyperlinks, Lists, Tables, Forms, Media Elements, CSS3 Selectors, Box Model, Positioning, Flexbox, Grid, Responsive Design. | **9** |
| | ● Create semantic webpage using HTML5 and CSS3.<br>● Design responsive layout using Flexbox and Grid.<br>● Develop styled form with validation and media elements. | **6** |
| **II** | **JavaScript Programming and DOM Manipulation** – Variables (var, let, const), Data Types, Operators, Control Flow, Functions, Arrays, Objects, JSON, DOM Selection and Manipulation, Event Handling, Form Validation using Regex, Browser Storage (LocalStorage, SessionStorage), ES6+ Features. | **9** |
| | ● Implement dynamic form validation using DOM.<br>● Develop To-Do List using localStorage.<br>● Create interactive image gallery with event handling. | **6** |
| **III** | **React and SPA Development** – SPA vs MPA Architecture, Virtual DOM, React Setup (Vite/npm), JSX, Functional Components, Props, State (useState), useEffect, Conditional Rendering, Controlled Forms, List Rendering with Keys, React Router, Context API, API Integration (Fetch/Axios). | **9** |
| | ● Build multi-page SPA using React Router.<br>● Fetch and display API data using useEffect.<br>● Implement Theme Switcher using Context API. | **6** |
| **IV** | **Backend Development and Full Stack Integration** – Node.js Architecture, NPM, Express.js Routing and Middleware, HTTP Methods, Template Engines (Intro), JSON Handling, Authentication (Cookies, Sessions, JWT – Intro), MongoDB, Mongoose, CRUD Operations, REST API Design, AJAX/Fetch Integration, MERN Stack Overview. | **9** |

| | | |
|---|---|---|
| | • Develop CRUD REST API using Express and MongoDB.<br>• Implement login system with encrypted passwords.<br>• Connect React frontend with Express backend. | **6** |
| **V** | **Security, DevOps and Deployment** – HTTPS, CSP, OWASP Top 10 (SQL Injection, XSS, CSRF), Input Sanitization, Secure Cookies, Git Workflow, Branching and Merging, CI/CD Concepts, Cloud Deployment (Netlify, Vercel, Render, MongoDB Atlas), Performance Optimization (Caching, Minification, Lazy Loading), Monitoring Tools (Lighthouse, DevTools), Scalability and Maintainability. | **9** |
| | ● Deploy MERN application with cloud database.<br>● Configure CI/CD pipeline.<br>● Analyze and improve performance using Lighthouse. | **6** |
| | Total Hours | **75** |

## vi. ASSESSMENT PATTERN

Continuous Assessment: End Semester Examination – 60: 40

| **Continuous Assessment** | | |
|---|---|---|
| Attendance | : | 5 marks |
| Assignments | : | 15 marks |
| Assessment through Tests | : | 20 marks |
| Lab Work | : | 10 marks |
| Lab Exam | : | 10 marks |
| **Total Continuous Assessment** | : | **60 marks** |
| **End Semester Examination** | : | **40 marks** |
| **TOTAL** | : | **100 marks** |

## vii. CONTINUOUS ASSESSMENT TEST

- No. of Tests: 02
- Maximum Marks: 30

- Test Duration: 1 ½ hours
- Topics: 2 ½ modules

**viii. END SEMESTER EXAMINATION**

- Maximum Marks: 40
- Exam Duration: 2 hours

| Course Code | Course Name | Category | L | T | P | J | Credit | Year of Introduction |
|---|---|---|---|---|---|---|---|---|
| **23CSL43A** | **INTERNET OF THINGS** | **PEC** | **2** | **1** | **0** | **0** | **3** | **2023** |

**i. COURSE OVERVIEW**

This course equips the learners with fundamental of the Internet of Things (IoT) and the IoT ecosystem. It covers the architecture of IoT, communication mechanisms, protocols, hardware, software, data analytics, and the cloud platforms for IoT. This course enables the students to design smart IoT applications for real world problems using Raspberry Pi.

**ii. COURSE OUTCOMES**

After the completion of the course, the student will be able to:

| Course Outcomes | Description | Level |
|---|---|---|
| CO1 | Explain the fundamentals of IoT and its underlying physical and logical architecture. | Understand |
| CO2 | Illustrate the hardware architectures for IoT. | Understand |
| CO3 | Explain the Network architectures for IoT. | Understand |
| CO4 | Implement data analytics on the IoT platforms. | Apply |
| CO5 | Interpret the security considerations in IoT. | Understand |
| CO6 | Implement IoT applications using the available hardware and software. | Apply |

**iii) SYLLABUS**

**Module 1: IoT Architecture**

Introduction to IoT – Genesis, Digitization, and Impact. Convergence of IT and IoT. IoT challenges and network architectures. Core IoT functional stack. IoT data management and compute stack.

**Module 2: Engineering IoT Networks**
Smart objects and IoT components – sensors, actuators, and networks. Connecting smart objects. Communication criteria and IoT access technologies.

**Module 3: IoT Network Layer**
Role of IP in IoT networks. IP optimization and compliance. IoT transport and application protocols. IoT application transport methods.

**Module 4: Data Analytics for IoT**
IoT data analytics and machine learning. Big data and edge analytics. Network analytics and IoT security. IT vs. OT security practices. Risk analysis frameworks – OCTAVE, FAIR.

**Module 5: Developing IoT Systems**
IoT system design using Python. Raspberry Pi interfaces and programming. IoT devices and cloud integration. Cloud storage models, WAMP, Django, and RESTful APIs for IoT applications.

**iv (a) TEXT BOOKS**

1. David Hanes, Gonzalo Salgueiro, Patrick Grossetete, Robert Barton, Jerome Henry, "IoT Fundamentals: Networking Technologies, Protocols, and Use Cases for the Internet of Things", 1st Edition, Pearson Education (Cisco Press Indian Reprint).
2. Arshadeep Bahga, Vijay Madisetti, "Internet of Things: A hands-on approach", University Press, 2015 (First edition)

**(b) REFERENCES**

1. Rajkamal, "Internet of Things: Architecture and Design Principles", McGraw Hill (India) Private Limited
2. Dieter Uckelmann, Mark Harrison, Michahelles Florian (Ed.), Architecting the internet of things, Springer, 2011
3. Dr. Ovidiu Vermesan, Dr. Peter Friess, Internet of Things: Converging Technologies for Smart Environments and Integrated Ecosystems, River Publishers, 2013
4. Simon Monk, "Programming Arduino: Getting Started with Sketches", McGraw Hill Publications.

**v. COURSE PLAN**

| Module | Contents | Hours |
|---|---|---|
| I | **IoT Architecture**<br>What is IoT, Genesis of IoT, IoT and Digitization, IoT Impact, Convergence of IT and IoT, IoT Challenges, IoT Network Architecture and Design, Drivers Behind New Network Architectures, Comparing IoT Architectures, A Simplified IoT Architecture, The Core IoT Functional Stack, IoT Data Management and Compute Stack. | **9** |
| II | **Engineering IoT Networks**<br>Smart Objects: The "Things" in IoT, Sensors, Actuators, and Smart Objects, Sensor Networks, Connecting Smart Objects, IoT Access Technologies – IEEE 802.15.4 (g/e), 1901.2a, IoT Access Technologies - 802.11ah, LoRaWAN, IoT Access Technologies – LoRaWAN, NBIoT, LTE. | **9** |
| III | **IoT Network Layer**<br>IP as the IoT Network Layer, The Business Case for IP, the need for Optimizing IP for IoT, Optimizing IP for IoT, Profiles, and Compliance, Application Protocols for IoT – CoAP, Application Protocols for IoT – MQTT, The Transport Layer, IoT Application Transport Methods, The Transport Layer, IoT Application Transport Methods. | **9** |
| IV | **Data Analytics for IoT**<br>An Introduction to Data Analytics for IoT, Machine Learning, Big Data Analytics Tools and Technology, Edge Streaming Analytics, Network Analytics, A Brief History of OT Security, Common Challenges in OT Security, Differences between IT and OT Security Practices and Systems, Formal Risk Analysis Structures: OCTAVE and FAIR. | **9** |
| V | **Developing IoT Systems**<br>IoT Logical Design using Python, IoT Physical Devices and Endpoints, Raspberry Pi interfaces, Programming Raspberry Pi using Python, Other IoT devices, Cloud Storage Models, WAMP-Autobahn for IoT, Django, Designing RESTful Web API, Cloud Web Services for IoT | **9** |
| | **Total Hours** | **45** |

**vi. ASSESSMENT PATTERN**

Continuous Assessment: End Semester Examination – 40: 60

| Continuous Assessment | | |
|---|---|---|
| Attendance | : | 5 marks |
| Assignments | : | 15 marks |
| Assessment through Tests | : | 20 marks |

| | | |
|---|---|---|
| **Total Continuous Assessment** | : | **40 marks** |
| **End Semester Examination** | : | **60 marks** |
| **TOTAL** | : | **100 marks** |

**vii. CONTINUOUS ASSESSMENT TEST**

- No. of Tests: 02
- Maximum Marks: 30
- Test Duration: 1 ½ hours
- Topics: 2 ½ modules

**viii. END SEMESTER EXAMINATION**

- Maximum Marks: 60
- Exam Duration: 3 hours

| Course Code | Course Name | Category | L | T | P | J | Credit | Year of Introduction |
|---|---|---|---|---|---|---|---|---|
| 23CSL43B | COMPUTATIONAL COMPLEXITY | PEC | 2 | 1 | 0 | 0 | 3 | 2023 |

## i. COURSE OVERVIEW

Computational Complexity introduces the theoretical foundations for classifying computational problems based on resource requirements.

## ii. COURSE OUTCOMES

After the completion of the course, the student will be able to:

| Course Outcomes | Description | Level |
|---|---|---|
| CO1 | Explain the complexity measures and classes. | Understand |
| CO2 | Analyze time and space complexity classes. | Analyze |
| CO3 | Apply reductions to prove NP-completeness. | Apply |
| CO4 | Compare randomized and deterministic models. | Analyze |
| CO5 | Explain circuit and parallel complexity. | Understand |

## iii. SYLLABUS

Introduction to computational complexity, motivation, models of computation (Turing machines and RAM model), time and space complexity, asymptotic notations, decision problems, and complexity hierarchies. Study of deterministic and nondeterministic complexity classes including P, NP, co-NP, EXP, and PSPACE, along with hierarchy theorems and class relationships. NP-completeness theory with polynomial-time reductions and classical NP-complete problems. Circuit complexity including Boolean circuits, $AC^0$, $NC^1$, $TC^0$ classes, and lower bound results. Advanced topics such as

randomized complexity classes, parallel complexity, approximation classes, and interactive proof systems.

**iv(a)TEXTBOOKS**

1. Michael Sipser – *Introduction to the Theory of Computation*, 3rd Edition, Cengage, 2013

2. Steven Homer & Alan L. Selman – *Computability and Complexity Theory*, 2nd Edition, Springer, 2011

3. Ding-Zhu Du & Ker-I Ko – *Theory of Computational Complexity*, 2nd Edition, Wiley, 2014

4. Sanjeev Arora & Boaz Barak – *Computational Complexity: A Modern Approach*, Cambridge University Press, 2009

**(b) REFERENCES**

1. Oded Goldreich – *Computational Complexity: A Conceptual Perspective*, Cambridge University Press, 2008

2. Lane A. Hemaspaandra & Mitsunori Ogihara – *The Complexity Theory Companion*, Springer, 2002/republished 2010

3. Robert A. Meyers (Ed.) – *Computational Complexity: Theory, Techniques, and Applications*, Springer, 2012

4. Michael R. Garey & David S. Johnson – *Computers and Intractability: A Guide to the Theory of NP-Completeness*, W.H. Freeman, 1979.

**v) COURSE PLAN**

| Module | Contents | Hours |
|---|---|---|
| **I** | Introduction to Computational Complexity - Motivation and scope of complexity theory, Models of computation: Turing Machines and RAM model, Time and space complexity: Asymptotic notations: O, Ω, Θ, Complexity measures and decision problems, Complexity hierarchy. | **8** |
| **II** | Time and Space Complexity Classes - Deterministic and nondeterministic Turing machines, Classes: P, NP, co-NP, EXP, PSPACE. Time and Space Hierarchy Theorems: Savitch's Theorem, Relationship between classes. | **10** |
| **III** | NP-Completeness and Reductions - Polynomial-time reductions: Cook-Levin Theorem, NP-complete problems: SAT, 3-SAT, CLIQUE, VERTEX COVER, SUBSET SUM, HAMILTONIAN CYCLE. NP-hard vs NP-complete. | **10** |
| **IV** | Circuit Complexity & Lower Bounds - Boolean circuits and families: Circuit size and depth, Classes: $AC^0$, $NC^1$, $TC^0$. Lower bounds for $AC^0$ (PARITY not in $AC^0$), Uniform vs non-uniform circuits. | **9** |
| **V** | Advanced Complexity Classes - Randomized classes: RP, BPP, ZPP. Probabilistic algorithms, Parallel complexity: NC, PRAM model. Approximation classes: APX, PTAS. Interactive proof systems and IP = PSPACE. | **8** |
| | Total Hours | **45** |

**vi. ASSESSMENT PATTERN**

Continuous Assessment: End Semester Examination – 40: 60

| Continuous Assessment | | |
|---|---|---|
| Attendance | : | 5 marks |
| Assignments | : | 15 marks |

| | | |
|---|---|---|
| Assessment through Tests | : | 20 marks |
| **Total Continuous Assessment** | : | **40 marks** |
| **End Semester Examination** | : | **60 marks** |
| **TOTAL** | : | **100 marks** |

**vii. CONTINUOUS ASSESSMENT TEST**

- No. of Tests: 02
- Maximum Marks: 30
- Test Duration: 1 ½ hours
- Topics: 2 ½ modules

**viii. END SEMESTER EXAMINATION**

- Maximum Marks: 60
- Exam Duration: 3 hours

| Course Code | Course Name | Category | L | T | P | J | Credit | Year of Introduction |
|---|---|---|---|---|---|---|---|---|
| 23CSL43C | DEEP LEARNING | PEC | 2 | 1 | 0 | 0 | 3 | 2023 |

## i. COURSE OVERVIEW

This course provides the learners an overview of the concepts and algorithms involved in deep learning. The course covers the basic concepts in neural networks, deep learning, optimization techniques, regularization techniques, convolutional neural networks, recurrent neural networks, autoencoders, and generative models. The students will be able to implement deep learning algorithms to solve real-world problems.

## ii. COURSE OUTCOMES

After the completion of the course, the student will be able to:

| Course Outcomes | Description | Level |
|---|---|---|
| CO1 | Explain the basic concepts of deep learning. | Understand |
| CO2 | Explain the deep learning techniques for structured/tabular data. | Understand |
| CO3 | Apply convolutional Neural Network (CNN) models for different use cases. | Apply |
| CO4 | Apply sequence-to-sequence architectures, auto encoder and generative models to solve real world problems | Apply |
| CO5 | Make use of the concept of transformers and generative text models in natural language processing tasks. | Apply |

## iii. SYLLABUS

**Foundations & Training Optimization** Deep Feedforward Networks, Weight Initialization, Vanishing and Exploding Gradients, Gradient Descent (GD, SGD, Momentum, Nesterov, AdaGrad, RMSProp, Adam), L1 and L2 Regularization, Early Stopping, Dataset Augmentation, Dropout, Batch Normalization.

**Deep Learning for Structured Data** Nature and characteristics of Tabular Data, Feature Scaling, Entity Embeddings, Model Optimization, Evaluation Metrics, Overfitting, Decision Trees vs. XGBoost vs. Deep Learning, Hybrid ML Models.

**Computer Vision & CNNs** Convolution Operation, 3D Convolution, Padding, Stride, Pooling, Sparse Connections, Weight Sharing, Transfer Learning, VGG, ResNet, GoogleNet, EfficientNet, Object Detection (R-CNN, YOLO).

**Sequential Modeling & GAN** RNN, LSTM, GRU, Encoder-Decoder Models, Attention Mechanism, Autoencoders (Variational, Under-complete, Denoising), Generative Adversarial Networks (GANs)

**NLP & Advanced Architectures** Transformers, Word2Vec, BERT Embeddings, Named Entity Recognition (NER), Text Classification, Generative Text Models

**iv (a) TEXT BOOKS**

1. Ian Goodfellow, Yoshua Bengio, and Aaron Courville**,** *Deep Learning*, MIT Press, 2016**.**
2. Charu C. Aggarwal, *Neural Networks and Deep Learning*, Springer International Publishing, part of Springer Nature**,** 2018**.**
3. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I., *Attention Is All You Need*, NeurIPS, 2017

**(b) REFERENCES**

1. Nikhil Buduma and Nicholas Locascio, *Fundamentals of Deep Learning: Designing Next-Generation Machine Intelligence Algorithms*, 1st Edition, O'Reilly Media, Inc., 2017.
2. M. Gopal**,** *Deep Learning: Core Concepts, Methods and Applications*, Pearson Education, 2022**.**

**v. COURSE PLAN**

| Module | Contents | Hours |
|---|---|---|
| **I** | **Introduction to Deep learning and training Deep models**<br>Introduction to deep learning. Artificial Neurons and Deep Feedforward Networks: Perceptron, multilayer perceptron (MLP), activation functions. Network Architecture and Training Setup: Loss functions, weight initialization strategies, bias initialization. Training Challenges: Vanishing | **10** |

| | | |
|---|---|---|
| | and exploding gradient problems. Optimization Techniques: Gradient Descent (GD), Stochastic Gradient Descent (SGD)<br><br>Regularization Techniques: L1 and L2 regularization, early stopping, dropout, batch normalization. | |
| **II** | **Deep Learning for Structured Data**<br><br>Nature and characteristics of tabular datasets, challenges in applying deep learning. Neural Network Models for Structured Data.<br><br>Data Preprocessing Techniques: Feature scaling and normalization, handling missing values, encoding and embeddings for categorical variables.<br><br>Model Training and Optimization: Hyperparameter tuning, regularization techniques, handling class imbalance.<br><br>Evaluation of Tabular Models: Performance metrics, cross-validation, overfitting and generalization issues. | **7** |
| **III** | **Convolutional Neural Network-CNN**<br><br>Convolution Operations: Convolution, padding, stride, pooling operations, and 3D convolution. CNN Design Principles: Weight sharing, sparse connectivity, and variants of convolution functions. Transfer Learning: Fine-tuning and feature extraction using pre-trained CNN models. Pré-trained CNN Architectures: VGG, GoogLeNet, ResNet, EfficientNet.<br><br>Applications of CNNs: Image classification, object detection. Object Detection Models: R-CNN family, YOLO. | **11** |
| **IV** | **Sequential Models & GAN**<br><br>Architectures- Recurrent neural networks, LSTM, GRU. Encoder-Decoder Models ,Attention Mechanism. Autoencoders, Types of autoencoders: Variational Auto-Encoder, stochastic encoder, denoising encoder.<br>Generative models: generative adversarial networks (GAN). Applications of GAN. | **9** |
| **V** | **NLP & Advanced Architectures**<br><br>NLP-Language Modeling (GPT, BERT), Transformer models, Text Classification— Named Entity Recognition (NER), Text Embedding & | **8** |

| | | |
|---|---|---|
| | Representation Learning - Word2Vec, BERT Embedding. Generative Text Models , LLM, SLM. | |
| | **Total Hours** | **45** |

## vii. ASSESSMENT PATTERN

Continuous Assessment: End Semester Examination – 40: 60

| Continuous Assessment | | |
|---|---|---|
| Attendance | : | 5 marks |
| Assignments | : | 15 marks |
| Assessment through Tests | : | 20 marks |
| **Total Continuous Assessment** | : | **40 marks** |
| **End Semester Examination** | : | **60 marks** |
| **TOTAL** | : | **100 marks** |

## viii. CONTINUOUS ASSESSMENT TEST

- No. of Tests: 02
- Maximum Marks: 30
- Test Duration: 1 ½ hours
- Topics: 2 ½ modules

## ix. END SEMESTER EXAMINATION

- Maximum Marks: 60
- Exam Duration: 3 hours

| Course Code | Course Name | Category | L | T | P | J | Credit | Year of Introduction |
|---|---|---|---|---|---|---|---|---|
| 23CSL43D | CLOUD SECURITY | PEC | 2 | 1 | 0 | 0 | 3 | 2023 |

## i. COURSE OVERVIEW

This course provides an understanding of cloud computing fundamentals, architecture, and resource optimization techniques along with essential cloud security principles. It covers various cloud security measures to ensure confidentiality, integrity, and availability of data. The course will enable students to analyze various cloud environments and apply effective security measures.

## ii. COURSE OUTCOMES

After the completion of the course, the student will be able to:

| Course Outcomes | Description | Level |
|---|---|---|
| CO1 | Explain cloud computing fundamentals | Understand |
| CO2 | Explain cloud reference model and virtualized environments | Understand |
| CO3 | Apply cloud security measures effectively, including authentication, authorization, and auditing, to ensure confidentiality, integrity, and availability of data in cloud computing environments. | Apply |
| CO4 | Identify cloud computing risks and security challenges, CIA triad and privacy regulations. | Understand |
| CO5 | Explain cloud security principles, design secure environments and communications, integrate robust identity management, and enforce access controls for autonomic security. | Understand |

## iii. SYLLABUS

Cloud basics, architecture, and deployment models to improve how your organization manages its IT infrastructure and operations - Principles and practices of securing cloud-based systems to safeguard data confidentiality, integrity, and availability- Security risks in cloud computing environments- Secure cloud computing environments through principles of architecture, trusted methodologies, identity management, and access control.

**iv (a) TEXT BOOKS**

1. Rajkumar Buyya, Christian Vecchiola, S Tamarai Selvi "Mastering Cloud Computing Foundations And Applications Programming" , McGraw Hill Education, 2016.
2. Ronald L. Krutz, Russell Dean Vines “Cloud Security A Comprehensive Guide to Secure Cloud Computing” Wiley Publishing. INC.
3. Kai Hwang, Geoffrey C Fox, Jack J Dongarra, "Distributed and Cloud Computing - From Parallel Processing to the Internet of Things", Morgan Kaufman Publishing, 2012

**(b) REFERENCES**

1. Sirisha Potluri et.al. “Cloud Security Techniques and Applications (Smart Computing Applications)”.
2. Chris Dotson, “Practical Cloud Security A Guide for Secure Design and Deployment”. O’Riley Publishing

**v. COURSE PLAN**

| Module | Contents | Hours |
|---|---|---|
| **I** | **Introduction & Cloud Computing Architecture**: Cloud computing at a glance: Defining a cloud, A closer look, The cloud computing reference model, Characteristics and benefits. Historical developments: Distributed systems, Virtualization, Web 2.0, Service oriented computing, Utility-oriented computing Building cloud computing environments: Application development. | **9** |

| | | |
|---|---|---|
| **II** | **The cloud reference model:** Architecture, Infrastructure-and hardware-as-a security service, Platform as a service, Software as a service. Types of clouds: Public, Private, Hybrid, Open Challenges: Cloud definition, Cloud interoperability and standards, Scalability and fault tolerance, Security, trust, and privacy, Organizational aspects.<br>**Virtualization:** Introduction, Characteristics of Virtualized Environments, Taxonomy of Virtualization Techniques - Execution Virtualization , Pros and Cons of Virtualization, Technology Examples – Xen, VMware, Microsoft Hyper- V | **10** |
| **III** | **Secure Cloud Software Requirement**-Cloud Computing Software Security Fundamentals: Cloud Information Security Objectives- Confidentiality, Integrity, Availability, Cloud Security services- Authentication, Authorization, Auditing, Accountability. Secure Development Practices, Approaches to Cloud Software Requirements Engineering. | **10** |
| **IV** | **Cloud Computing Risk Issues and Security Challenges**: Privacy and Compliance Risks, Threats to Infrastructure, Data, and Access Control, Cloud Service Provider Risks, Cloud Computing | **8** |
| | Security Challenges-Security Policy Implementation, Computer Security Incident Response Team (CSIRT). | |
| **V** | **Cloud Computing Security Architecture:**<br>Introduction, Architectural Considerations- General Issues, Trusted Cloud Computing, Secure Execution Environments and Communications, Microarchitectures, Identity Management and Access Control- Identity Management, Access Control, Autonomic Security | **8** |
| | **Total Hours** | **45** |

## vi. ASSESSMENT PATTERN

Continuous Assessment: End Semester Examination – 40: 60

| Continuous Assessment | | |
|---|---|---|
| Attendance | : | 5 marks |
| Assignments | : | 15 marks |
| Assessment through Tests | : | 20 marks |
| **Total Continuous Assessment** | : | **40 marks** |
| **End Semester Examination** | : | **60 marks** |
| **TOTAL** | : | **100 marks** |

**vii. CONTINUOUS ASSESSMENT TEST**

- No. of Tests: 02
- Maximum Marks: 30
- Test Duration: 1 ½ hours
- Topics: 2 ½ modules

**viii. END SEMESTER EXAMINATION**

- Maximum Marks: 60
- Exam Duration: 3 hours

| Course Code | Course Name | Category | L | T | P | J | Credit | Year of Introduction |
|---|---|---|---|---|---|---|---|---|
| 23CSL43E | HUMAN COMPUTER INTERACTION | PEC | 2 | 1 | 0 | 0 | 3 | 2023 |

**i. COURSE OVERVIEW**

This course provides an overview of Human-Computer Interaction (HCI), with an understanding of user interface design in general. The course covers topics which include user-centered design, human cognitive and physical abilities, prototyping and evaluation techniques, graphical design fundamentals and emerging areas of HCI research including mobile interaction, augmented-reality and ubiquitous computing. This course helps the learners to design and evaluate interactive systems by following the fundamental principles of human-computer interaction.

**ii. COURSE OUTCOMES**

After the completion of the course, the student will be able to:

| Course Outcomes | Description | Level |
|---|---|---|
| CO1 | Demonstrate the usability based on a variety of classic universal user-centric models. | Understand |
| CO2 | Summarize the different interaction styles and the methodologies for designing interactive systems. | Understand |
| CO3 | Interpret the core and complex user experience design issues. | Understand |
| CO4 | Identify the evaluation methodologies of interactive system design. | Apply |
| CO5 | Make use of the different contexts and suggest suitable designs for applications related to web, mobile and wearable computing. | Apply |

**iii. SYLLABUS**

Human-Computer Interaction (HCI) focuses on designing interactive systems that enhance user experience through understanding user cognition, ergonomic principles, and user-centered

approaches. It emphasizes usability goals, universal usability for diverse users—including older adults, children, and those with disabilities—and addresses cultural and personality differences. HCI integrates guidelines, principles, and design theories alongside frameworks, prototyping methods, and interaction styles like direct manipulation, immersive environments, and expressive human-command languages. It also explores models of collaboration, system response times, frustrating experiences, and information search strategies. Cognitive models such as GOMS help in understanding user behavior, while various evaluation methods, including usability testing and heuristic evaluation, ensure design effectiveness. Furthermore, HCI encompasses the development of apps and websites, considering information architecture and social media integration, and extends into collaborative environments, AI interface agents, and pervasive technologies like mobile and wearable computing, emphasizing smart materials and material design for seamless user interaction.

**iv (a) TEXT BOOKS**

1. Ben Shneiderman, Catherine Plaisant, Maxine Cohen, Steven Jacobs, NiklasElmqvist *Designing the User Interface: Strategies for Effective HumanComputer Interaction*, Sixth Edition, Pearson Education, 2017.
2. Preece, J., Sharp, H., Rogers, Y., *Interaction Design: Beyond Human-Computer Interactio*, Fifth Edition, Wiley, 2019.
3. David Benyon, *Designing User Experience: A guide to HCI, UX and interaction design*, 4th Edition, Pearson, 2018.

**(b) REFERENCES**

1. Alan Dix, Janet Finlay, Gregory Abowd, Russell Beale, *Human Computer Interaction*, Third Edition, Prentice Hall, 2004.
2. The essential guide to user interface design, Wilbert O Galitz, Wiley Dream Tech 3.
3. Jonathan Lazar Jinjuan Heidi Feng, Harry Hochheiser, *Research Methods in Human Computer Interaction*, Wiley, 2010.
4. Samit Bhattacharya, *Human-Computer Interaction: User-Centric Computing for Design*, McGraw-Hill India, 1st Edition, 2019.

| **v. COURSE PLAN** | | |
|---|---|---|
| **Module** | **Contents** | **Hours** |
| **I** | **Introduction to HCI and Usability**<br><br>Introduction- - Components of Interaction – Ergonomics Designing Interactive systems – Understanding Users cognition and cognitive frameworks, User Centered approaches, Usability goals and measures, Universal Usability-Diverse Cognitive and Perceptual abilities, Personality differences, Cultural and International diversity, Users with disabilities- Older Adult users and Children. Guidelines, Principles and Theories. | **8** |
| **II** | **Design Process and Interaction Styles**<br><br>HCI patterns, Design frameworks, Design methods, Prototyping. Understanding interaction styles - Direct Manipulation and Immersive environments, Fluid navigation - Navigation by Selection, Small Displays, Content Organization, Expressive Human and Command Languages-Speech Recognition, Traditional Command Languages, Communication and Collaboration-Models of Collaboration, Design considerations. | **8** |
| **III** | **User Experience Design**<br><br>Frameworks for User Centric Computing, Computational models of users, Advancing the User Experience- Display Design, View (Window) Management, Animation, Webpage Design, Color. Timely user Experience-Models of System Response Time (SRT) Impacts, Frustrating Experiences, Information Search- Five Stage Search Framework, Data Visualization-Tasks in Data Visualization, Challenges | **9** |
| **IV** | **Cognitive Systems and Evaluation of HCI**<br><br>Cognitive Models- Goal and task hierarchies, GOMS Model. Introducing Evaluation Types of Evaluation, Other Issues to Consider When Doing Evaluation. Conducting Experiments. Usability testing – Heuristic evaluation and walkthroughs, Analytics and predictive models. | **10** |
| **V** | **Contexts for Designing UX**<br><br>Designing apps and websites – Website and app development, The information architecture of apps and websites. Social media -Social Networking, Sharing with others. Collaborative environments- Issues for | **10** |

| | | |
|---|---|---|
| | cooperative working, Technologies to support cooperative working, AI and Interface Agents, Ubiquitous computing -Blended Spaces. Mobile Computing – Designing for Mobiles. Wearable Computing- Smart Materials, Material Design. | |
| | **Total Hours** | **45** |

## vi. ASSESSMENT PATTERN

Continuous Assessment: End Semester Examination – 40: 60

| Continuous Assessment | | |
|---|---|---|
| Attendance | : | 5 marks |
| Assignments | : | 15 marks |
| Assessment through Tests | : | 20 marks |
| **Total Continuous Assessment** | : | **40 marks** |
| **End Semester Examination** | : | **60 marks** |
| **TOTAL** | : | **100 marks** |

## vii. CONTINUOUS ASSESSMENT TEST

- No. of Tests: 02
- Maximum Marks: 30
- Test Duration: 1 ½ hours
- Topics: 2 ½ modules

## viii. END SEMESTER EXAMINATION

- Maximum Marks: 60
- Exam Duration: 3 hours

| Course Code | Course Name | Category | L | T | P | J | Credit | Year of introduction |
|---|---|---|---|---|---|---|---|---|
| 23CTL43A | Computer Vision | PEC | 2 | 1 | 0 | 0 | 3 | 2023 |

## i.COURSE OVERVIEW

This course introduces the foundational principles of computer vision, covering how images are formed, represented, filtered, and analysed. Students learn techniques for feature detection, segmentation, and object recognition, along with essential methods for motion analysis in dynamic scenes. The course then advances to deep learning models used in modern vision tasks, including classification, detection, and segmentation architectures. Learners explore practical applications such as image editing, super-resolution, 3D object generation, tracking, and automated captioning. The course concludes with emerging topics like zero-shot learning, few-shot learning, and self-supervised approaches that shape the future of vision systems.

## ii. COURSE OUTCOMES

After the completion of the course, the student will be able to:

| Course Outcomes | Description | Level |
|---|---|---|
| CO1 | Explain the fundamental concepts, principles, and working mechanisms of Computer Vision systems. | Understand |
| CO2 | Apply feature extraction, segmentation, and object detection techniques for image analysis. | Apply |
| CO3 | Apply motion patterns using optical flow, tracking, background modeling, and other motion analysis methods. | Apply |
| CO4 | Develop and evaluate deep learning models tailored for various computer vision applications. | Apply |
| CO5 | Develop complete computer vision solutions for real-world applications and explore recent trends such as zero-shot and self-supervised learning. | Apply |

**iii.SYLLABUS**

Basics of Computer Vision, Image Formation-Image Representation.
Image Analysis, Features & Object Detection, Image Segmentation, Feature matching. Motion Analysis - Background Subtraction and Modeling-Optical Flow- KLT- Spatio- Temporal Analysis-Dynamic Stereo- Motion, Deep Learning models for Computer Vision Applications and recent trends in Computer Vision

**iv a) TEXTBOOKS**

1. Ian Goodfellow, Yoshua Bengio, Aaron Courville, "Deep Learning", 2016.
2. Richard Szeliski, "Computer Vision: Algorithms and Applications", 2010.
3. Simon Prince, "Computer Vision: Models, Learning, and Inference", 2012.
4. Adrian Kaehler, Gary Bradski, "Learning OpenCV 3", 2017

**b) REFERENCES**

1. Yoshua Bengio, Learning Deep Architectures for AI, 2009.
2. David Forsyth, Jean Ponce, Computer Vision: A Modern Approach, 2002.
3. Michael Nielsen, Neural Networks and Deep Learning, 2016.
4. Richard Hartley and Andrew Zisserman, Multiple View Geometry in Computer Vision, Second Edition, Cambridge University Press, March 2004
5. Fei-Fei Li – CS231n Lecture Notes (CNNs & Deep Learning for Vision)

**v.COURSE PLAN**

| Module | Contents | No. of hours |
|---|---|---|
| **I** | Basics of Computer Vision<br>Introduction to Computer Vision: Importance, applications, challenges, Computer Vision vs Image Processing vs Graphics - Image Formation-Image Representation-Linear Filtering-Image in frequency domain- Image Sampling. | **7** |
| **II** | Image Analysis, Features & Object Detection<br>Feature detection - Feature descriptors - SIFT and its variants- Feature matching - Edge detection - Image Segmentation: Graph-based segmentation, Mean shift, and Normalized cuts. | **9** |
| **III** | Motion Analysis<br>Background Subtraction and Modeling-Optical Flow- KLT- Spatio-Temporal Analysis-Dynamic Stereo- Motion | **9** |

| | | |
|---|---|---|
| **IV** | Deep Learning models for Computer Vision<br>Object Classification-VGGNET, RESNET, ALEXNET, DENSENET, EFFICIENT NET,<br>MOBILENET, INCEPTION V3, Object Detection-R-CNN, F-RCN, SSD, Retinanet, YOLO, CornerNet, Image Segmentation- UNet, SegNet, Mask-RCNN, Attention Models-Transformers | **11** |
| **V** | Applications and recent trends in Computer Vision<br>Applications- Image Editing, Inpainting, Superresolution, 3D Object Generation, Security, Surveillance-Object Tracking-Automatic Image Captioning.<br>Recent Trends- Zero-shot, One-shot, Few-shot Learning-Self-supervised Learning and Reinforcement Learning in Vision | **9** |
| | **Total** | **45** |

## vi. ASSESSMENT PATTERN

Continuous Assessment: End Semester Examination – 40: 60

| Continuous Assessment | | |
|---|---|---|
| Attendance | : | 5 marks |
| Assignments | : | 15 marks |
| Assessment through Tests | : | 20 marks |
| **Total Continuous Assessment** | : | **40 marks** |
| **End Semester Examination** | : | **60 marks** |
| **TOTAL** | : | **100 marks** |

## vii. CONTINUOUS ASSESSMENT TEST

- No. of Tests: 02
- Maximum Marks: 30

## viii. END SEMESTER EXAMINATION

- Maximum Marks: 60
- Exam Duration: 3 hours

| Course Code | Course Name | Category | L | T | P | J | Credit | Year of Introduction |
|---|---|---|---|---|---|---|---|---|
| **23CTL43C** | **AI FOR CYBERSECURITY** | **PEC** | **2** | **1** | **0** | **0** | **3** | **2023** |

## i. COURSEOVERVIEW

This course introduces the integration of Artificial Intelligence (AI) and Machine Learning (ML) techniques in strengthening cybersecurity mechanisms. It focuses on how AI models can enhance threat detection, automate response systems, and predict potential security breaches. The course also emphasizes the ethical and legal considerations surrounding the use of AI in cybersecurity, encouraging responsible and transparent technology adoption. By the end of this course, students will be able to identify and apply different AI-based security solutions for modern cybersecurity challenges.

## ii.COURSE OUTCOMES

After the completion of the course, the student will be able to:

| Course Outcomes | Description | Level |
|---|---|---|
| CO 1 | Explain the fundamental concepts of Artificial Intelligence, Machine Learning, and their relevance in cybersecurity. | Understand |
| CO 2 | Apply machine learning and anomaly detection techniques to identify and predict cybersecurity threats. | Apply |
| CO 3 | Summarize AI-based intrusion detection and automated threat response mechanisms for enhanced network security. | Understand |
| CO 4 | Identify AI-specific security risks such as adversarial attacks and propose mitigation strategies. | Understand |
| CO 5 | Describe ethical, legal, and future implications of AI adoption in cybersecurity. | Understand |

## iii.SYLLABUS

**Introduction to AI and Cybersecurity** – Definition, scope, and types- Historical evolution of AI in technology and security domains- Basics of Machine Learning, Algorithms, Models, Training, Overfitting & Underfitting- Applications , Case Studies, Benefits and Limitations.

**AI Techniques for Security** – Anomaly Detection Techniques, Threat Intelligence and Prediction, Predictive Models for Threat Forecasting, Behavioral Analysis.

**Addressing AI Security Risks** – Designing AI-based Intrusion Detection Systems (IDS)-, Automated Threat Response, Integrating AI Solutions with Existing Security Tools, -Case Studies.

**AI Security Threats and Risks** – AI Security Threats and Risks, Mitigation Strategies, AI Governance.

**Future Trends and Emerging Technologies** - Emerging Technologies in AI-driven Cybersecurity, Future Research Directions in AI for Cybersecurity Case Studies on Next-Generation Security Frameworks, Discussion: The Future of Human–AI Collaboration in Cyber Defense.

**iv (a) TEXT BOOKS**

1.Melanie Mitchell, "Artificial Intelligence: A Guide for Thinking Humans",Penguin Books,2019.

2.Noura Al Moubayed, Pardeep Kumar , "AI in Cybersecurity: Applications, Risks and Challenges" CRC Press, 1/e, 2020.

3.Andreas C. Müller, Sarah Guido Introduction to Machine Learning with Python, O'Reilly Media, 1/e, 2016.

4.Vincent C. Müller , "Ethics of Artificial Intelligence and Robotics", Springer, 1/e, 2020.

5.David M. Hawkins , "AnomalyDetection for Monitoring Systems: A Practical Guide", Wiley, 1/e, 2019.

6.Michael L. Santarcangelo , " Threat Intelligence: A Practical Guide", Syngress, 1/e, 2020.

**(b) REFERENCES**

1. Ian Goodfellow, Jonathon Shlens, Christian Szegedy, Adversarial Machine Learning, MIT Press 1/e, 2021.
2. Ben Goertzel, Cassio Pennachin, Artificial Intelligence: The Next Generation, Wiley, 1/e, 2020.
3. Chris L. Bader, Patrick W. O'Leary, Behavioral Cybersecurity: The Psychology of Cybersecurity, CRC Press, 1/e, 2021.

## v. COURSE PLAN

| Module | Contents | Hours |
|---|---|---|
| **I** | **Introduction to AI and Cybersecurity** - Introduction to Artificial Intelligence – Definition, scope, and types (Narrow AI vs. General AI) Historical evolution of AI in technology and security domains-Basics of Machine Learning – Supervised, Unsupervised, and Reinforcement Learning-Key AI Terminologies: Algorithms, Models, Training, Overfitting & Underfitting-Applications of AI in Cybersecurity. Case Studies: Real-world applications of AI for threat detection, malware analysis, and phishing prevention-Benefits and Limitations of AI in Security Context. | **9** |
| **II** | **AI Techniques for Security** -Anomaly Detection Techniques-Statistical Methods for anomaly detection-Machine Learning Models: Isolation Forest, One-Class SVM, etc. Threat Intelligence and Prediction-Data Collection and Preprocessing-Feature Extraction and Engineering-Predictive Analytics-Building and Evaluating Predictive Models for Threat Forecasting. Case Studies on Threat Intelligence Systems. Behavioral Analysis-Behavioral Profiling and Pattern Recognition-Detecting Anomalies in User/System Behavior. | **8** |
| **III** | **Building AI Security Solutions** -Designing AI-based Intrusion Detection Systems (IDS)-Types of IDS: Network-based and Host-based-Enhancing Detection with AI Models. Automated Threat Response-Response Strategies and Automation Workflows-Integration with Threat Detection Systems. Integrating AI Solutions with Existing Security Tools-Compatibility and Effectiveness of Legacy Systems-Case Studies: Implementation of AI-Driven IDS and Threat Response. | **9** |
| **IV** | **Addressing AI Security Risks** -AI Security Threats and Risks-Adversarial Attacks on AI Models-Model Poisoning and Data Privacy Issues. Mitigation Strategies-Securing AI Models from Adversarial Manipulation-Data Protection and Model Validation Techniques. AI Governance-Ethical and<br><br>Legal Implications of AI in Security-Responsible AI Practices and Compliance Requirements. | **9** |

| | | |
|---|---|---|
| **V** | **Future Trends and Emerging Technologies :** Emerging Technologies in AI- driven Cybersecurity-Deep Learning and Reinforcement Learning in Security-AI in Cloud, IoT, and Edge Security. Future Research Directions in AI for Cybersecurity-Explainable AI (XAI) for Transparency-Quantum Computing and its Security Implications. Case Studies on Next-Generation Security Frameworks. Discussion: The Future of Human–AI Collaboration in Cyber Defense | **10** |
| | Total Hours | **45** |

## vi. ASSESSMENT PATTERN

Continuous Assessment: End Semester Examination – 40: 60

| Continuous Assessment | | |
|---|---|---|
| Attendance | : | 5 marks |
| Assignments | : | 15 marks |
| Assessment through Tests | : | 20 marks |
| **Total Continuous Assessment** | : | **40 marks** |
| **End Semester Examination** | : | **60 marks** |
| **TOTAL** | : | **100 marks** |

## vii. CONTINUOUS ASSESSMENT TEST

- No. of Tests: 02
- Maximum Marks: 30

## viii. END SEMESTER EXAMINATION

- Maximum Marks: 60
- Exam Duration: 3 hours

# INSTITUTE ELECTIVE II

| Course Code | Course Name | Category | L | T | P | J | Credit | Year of Introduction |
|---|---|---|---|---|---|---|---|---|
| 23IEL42E | RESPONSIBLE AI | IEC | 3 | 0 | 0 | 0 | 3 | 2023 |

## i. COURSE OVERVIEW

The objective of the course is to know about the responsibility of artificial intelligence (AI) to make AI more useful for society and humanity. The course will also teach principles and practices to perform responsible AI.

## ii. COURSE OUTCOMES

After the completion of the course, the student will be able to:

| Course Outcomes | Description | Level |
|---|---|---|
| 1 | Explain the sources of bias in AI systems and their impact on fairness. | Understand |
| 2 | Develop frameworks for ethical reasoning in AI decision-making processes. | Apply |
| 3 | Explain the importance of interpretability in AI models. | Understand |
| 4 | Develop real-world case studies on privacy preservation. | Apply |
| 5 | Explain the ethical status of AI systems and levels of ethical behaviour. | Understand |

## iii. SYLLABUS

Fairness, Bias, and Ethics in AI, Bias sources, fairness (group/individual/counterfactual), AI harms, risks, and case studies.
Ethical Decision Making, Seven Principles of Responsible AI, ethical theories, values, and ART (Accountability, Responsibility, Transparency).
Interpretability and Explainability, Interpretability importance, methods, scope, evaluation, and interpretable models (linear/logistic regression, decision trees).

Privacy Preservation Data privacy, protection methods, privacy-utility balance, differential privacy, federated learning, and case studies.
Responsible AI Ethical reasoning, artificial moral agents, governance, codes of conduct, inclusion, and diversity.

**iv. a TEXTBOOKS**

1. Virginia Dignum, "Responsible Artificial Intelligence: How to Develop and Use AI in a Responsible Way" Springer Nature, 2019.

**b REFERENCES**

1. Adnan Masood, Heather Dawe, Dr. Ehsan Adeli, " Responsible AI in the Enterprise", Packt Publishing, 2023.
2. Beena Ammanath, " Trustworthy AI", Wiley, 2022.
3. Christoph Molnar "Interpretable Machine Learning".Lulu, 1st edition, 2019.

**v. COURSE PLAN**

| Module | Contents | Hours |
|---|---|---|
| **I** | Autonomy – Adaptability – Interaction – Need for Ethics in AI - Fairness and Bias: Sources of Biases – Exploratory data analysis, limitations of a dataset – Group fairness and individual fairness – Counterfactual fairness - AI harms – AI risks: Case Study | **11** |
| **II** | Seven Principles of Responsible AI - Ethical theories – Values - Ethics in practice – Implementing Ethical Reasoning – The ART of AI : Accountability, Responsibility, Transparency | **8** |
| **III** | Importance of Interpretability and explainability – Taxonomy of Interpretability Methods – Scope of Interpretability – Evaluation of Interpretability – Model Transparency Techniques – Local and Global Explanation Methods: feature importance, SHAP, LIME) – Interpretable Models: Linear Regression – Logistic Regression – Decision Tree. | **10** |
| **IV** | Introduction to data privacy - Methods of protecting data - Importance of balancing data privacy and utility - Attack model – Privacy Preserving Learning - Differential Privacy – Federated Learning – Case Study | **8** |

| | | |
|---|---|---|
| **V** | Approaches to Ethical Reasoning by AI – Designing Artificial Moral Agents – Implementing Ethical Deliberation – Levels of Ethical Behaviour – The ethical status of AI system – Governance for Responsible AI – Codes of Conduct – Inclusion and Diversity | **8** |
| | **Total Hours** | **45** |

**vi. ASSESSMENT PATTERN**

Continuous Assessment: End Semester Examination – 40: 60

| Continuous Assessment | | |
|---|---|---|
| Attendance | : | 5 marks |
| Assignments | : | 15 marks |
| Assessment through Tests | : | 20 marks |
| **Total Continuous Assessment** | : | **40 marks** |
| **End Semester Examination** | : | **60 marks** |
| **TOTAL** | : | **100 marks** |

**vii. CONTINUOUS ASSESSMENT TEST**

- No. of Tests: 02
- Maximum Marks: 30
- Test Duration: 1 ½ hours
- Topics: 2 ½ modules

**viii. END SEMESTER EXAMINATION**

- Maximum Marks: 60
- Exam Duration: 3 hours

| Course Code | Course Name | Category | L | T | P | J | Credit | Year of Introduction |
|---|---|---|---|---|---|---|---|---|
| 23IEL42F | PROMPT ENGINEERING | IEC | 3 | 0 | 0 | 0 | 3 | 2023 |

## i. COURSE OVERVIEW

This course introduces students to the fundamentals of prompt engineering, focusing on designing effective prompts to leverage large language models (LLMs) such as OpenAI's GPT and other generative AI tools. It covers techniques to elicit accurate, creative, and domain-specific responses, manage limitations of LLMs, and apply prompting techniques across various domains, including natural language processing (NLP), creative writing, and problem-solving. By the end of the course, students will have a comprehensive understanding of prompt engineering principles and the ability to design, test, and refine prompts for real-world applications.

## ii. COURSE OUTCOMES

After the completion of the course, the student will be able to:

| Course Outcomes | Description | Level |
|---|---|---|
| CO1 | Explain the fundamental concepts of Generative AI and prompt engineering. | Understand |
| CO2 | Develop well-structured prompts to improve response quality and mitigate bias, ambiguity, and redundancy. | Apply |
| CO3 | Explain advanced prompt engineering techniques to enhance LLM performance and scalability. | Understand |
| CO4 | Explain ethical challenges, best practices, and emerging trends in prompt engineering to design and deploy responsible and effective AI-driven prompts for real-world applications. | Understand |
| CO5 | Utilize prompt engineering techniques for domain-specific applications. | Apply |

## iii. SYLLABUS

Introduction to Generative AI and Prompt Engineering, Introduction to Large Language Models: Architecture, Types of prompts – zero-shot, one-shot, and few-shot prompts. Evaluation Metrics for Prompts: Designing Basic Prompts. Common Pitfalls in Prompt Design. Advanced Prompt Engineering: Dynamic Prompting, Applications of Prompt Engineering: Domain-Specific Prompting: Debugging and Improving Model Responses: Identifying and mitigating model errors. Integrating Prompt Engineering with APIs: Using OpenAI API and other LLM services. Ethical Considerations and Future Directions: Ethical Challenges in Prompt Engineering: Bias mitigation, content filtering, and responsible AI usage. Best Practices for Safe and Reliable Prompt Design.

**iv (a) TEXT BOOKS**

1. Smith, John. Prompt Engineering: A Guide to Conversing with AI Models. AI Press, 2024.
2. Johnson, Alice. Generative AI and You: Harnessing the Power of Language Models. FutureTech Publications, 2023.

**(b) REFERENCES**

1. OpenAI Team. ChatGPT Prompt Engineering for Developers. Online Guide, 2023.
2. Brown, Tom, et al. Language Models are Few-Shot Learners. NeurIPS Proceedings, 2020.
3. Patil, Surya, and Henshall, Emily. Practical Prompting for AI Models. Digital Insights Publishing,2023

**v. COURSE PLAN**

| Module | Contents | Hours |
|---|---|---|
| I | **Introduction to Generative AI and Prompt Engineering:** Overview of Generative AI: Key concepts, types, and applications. Introduction to Large Language Models: Architecture, capabilities, and limitations. Fundamentals of Prompt Engineering: Types of prompts – zero-shot, one-shot, and few-shot prompts. Evaluation Metrics for Prompts: Accuracy, relevance, and user satisfaction. | **9** |
| II | **Designing Basic Prompts:** Structuring Prompts: Clarity, context, and specificity. Techniques for Improving Prompt Responses: Iterative refinement and phrasing. Experimenting with Prompt Styles: Open-ended vs. | **9** |

| | | |
|---|---|---|
| | directive, creative, and instructional prompts. Common Pitfalls in Prompt Design: Bias, ambiguity, and redundancy. | |
| **III** | **Advanced Prompt Engineering:** Chain-of-Thought Prompts: Step-by-step reasoning for complex queries. Multi-turn Conversations: Maintaining context and coherence over multiple interactions. Role-based Prompts: Using roles to guide LLM behavior.<br><br>**Dynamic Prompting:** Using placeholders and templates for scalable applications. | **9** |
| **IV** | **Ethical Considerations and Future Directions:** Ethical Challenges in Prompt Engineering: Bias mitigation, content filtering, and responsible AI usage. Best Practices for Safe and Reliable Prompt Design. Innovations in Prompt Engineering: Emerging techniques and trends. | **9** |
| **V** | **Domain-Specific Prompting:** Applications in healthcare, education, customer support, and creative industries. Prompting for Creative Writing: Story generation, poetry, and scriptwriting. Debugging and Improving Model Responses: Identifying and mitigating model errors. Integrating Prompt Engineering with APIs: Using OpenAI API and other LLM services<br><br>**Project: Designing and deploying a prompt for a real-world use case.** | **9** |
| | **Total Hours** | **45** |

## vi. ASSESSMENT PATTERN

Continuous Assessment: End Semester Examination – 40: 60

| Continuous Assessment | | |
|---|---|---|
| Attendance | : | 5 marks |
| Assignments | : | 15 marks |
| Assessment through Tests | : | 20 marks |
| **Total Continuous Assessment** | : | **40 marks** |
| **End Semester Examination** | : | **60 marks** |
| **TOTAL** | : | **100 marks** |

**vii. CONTINUOUS ASSESSMENT TEST**

- No. of Tests: 02
- Maximum Marks: 30
- Test Duration: 1 ½ hours
- Topics: 2 ½ modules

**viii. END SEMESTER EXAMINATION**

- Maximum Marks: 60
- Exam Duration: 3 hours

| Course Code | Course Name | Category | L | T | P | J | Credit | Year of Introduction |
|---|---|---|---|---|---|---|---|---|
| **23IEL42G** | **BUISNESS INTELLIGENCE AND ANALYTICS** | **IEC** | **3** | **0** | **0** | **0** | **3** | **2023** |

## i. COURSE OVERVIEW

This course provides a comprehensive understanding of Business Intelligence and Analytics, covering essential concepts, methodologies, and tools used to transform raw data into actionable insights for business decision-making. The course covers both technical aspects and practical application of analytics techniques, including data mining, predictive modeling, and data visualization. By applying these concepts to real-world business scenarios, students will learn how to leverage data for informed decision-making across various domains like marketing, and finance.

## ii. COURSE OUTCOMES

After the completion of the course, the student will be able to:

| Course Outcomes | Description | Level |
|---|---|---|
| CO1 | Summarize Business Intelligence concepts and the various types of analytics. | Understand |
| CO2 | Explain fundamental concepts and techniques of data management, including OLTP, relational databases, data warehousing, OLAP, and data mining, and their application in real-world business scenarios. | Understand |
| CO3 | Utilize data pre-processing, OLAP, and data warehousing methods to design and implement effective data management systems and analytical models for business intelligence. | Apply |
| CO4 | Explain the principles of classification, clustering, and machine learning techniques and recognize their role in advanced analytics and predictive modeling. | Understand |
| CO5 | Apply data mining techniques to solve business problems. | Apply |

**iii. SYLLABUS**

Overview of BIA- drivers of BIA, types of analytics, Technical architecture of BIA, Fundamentals of data management-Online Transaction Processing (OLTP), design process of databases, Relational databases, data warehousing, Online Analytical Processing (OLAP), data cube, Data Warehousing to Data Mining, Data Mining Concepts and Applications, Architecture of typical data mining system, Data Mining Functionalities, Data Mining Issues, Data pre-processing, overview of data mining techniques

Descriptive analytics and visualization- customer analytics, survival analysis, customer lifetime value, Classification- classification techniques, scoring models, classifier performance, ROC and PR curves, ensemble methods, Cluster analysis- clustering algorithms, cluster quality, Artificial Neural Networks (ANN), Text mining

**iv (a) TEXT BOOKS**

1. Ramesh Sharda, Dursun Delen and Efraim Turban, Business Intelligence and Analytics: System for Decision Support, 10th Edition, Pearson Global Edition, 2013
2. Gareth James, Daniela Witten, Trevor Hastie, Robert Tibshirani, An Introduction to Statistical Learning with Applications in R, Springer, 2013
3. Jiawei Han, Jian Pei, Hanghang Tong, Data Mining Concepts and Techniques, 4th edition, Morgan Kaufmann Publishers, 2022.

**(b) REFERENCES**

1. Ramesh Sharda, Dursun Delen and Efraim TurbanR. Sharda, Business Intelligence: A Managerial Perspective on Analytics, 4th Edition, Pearson, 2017.
2. Wayne Winston and S. Albright, Business Analytics: Data Analysis & Decision Making, 5th Edition, South-Western College Publishing, 2014.
3. Foster Provost and Tom Fawcett, Data Science for Business: What You Need to Know about Data Mining and Data-Analytic Thinking, First Edition, O'Reilly Media.
4. Rajkumar Buyya, Rodrigo N. Calheiros and Amir Vahid Dastjerdi, Big Data: Principles and Paradigms, Morgan Kaufmann Publishers, 2016

**v. COURSE PLAN**

| Module | Contents | Hours |
|---|---|---|
| **I** | Introduction to Business Intelligence & Analytics (BIA)-Overview of BIA, drivers of BIA, types of analytics: descriptive to prescriptive, Technical architecture of BIA, Case study | 8 |
| **II** | Data Management Fundamentals-Fundamentals of data management, OnLine Transaction Processing (OLTP), design process of databases, Relational databases, data warehousing, OnLine Analytical Processing (OLAP), data cube | **9** |
| **III** | Data Warehousing to Data Mining, Data Mining Concepts and Applications, Architecture of typical data mining system, Data Mining Functionalities, Data Mining Issues, Data pre-processing, overview of data mining techniques, case study<br>Descriptive analytics and visualization, customer analytics, survival analysis, customer lifetime value, case study | **9** |
| **IV** | Classification- classification techniques, scoring models, classifier performance, ROC and PR curves, Introduction to decision trees, tree induction, measures of purity, tree algorithms, pruning, ensemble methods<br>Clustering- measures of distance, clustering algorithms, K-means and other techniques, cluster quality | **10** |
| **V** | Artificial Neural Networks (ANN)- topology and training algorithms, back propagation, financial time series modelling using ANN<br>Text mining- process, key concepts, sentiment scoring, text mining using R, summary | **9** |
| | **Total Hours** | **45** |

## vi. ASSESSMENT PATTERN

Continuous Assessment: End Semester Examination – 40: 60

| Continuous Assessment | | |
|---|---|---|
| Attendance | : | 5 marks |
| Assignments | : | 15 marks |
| Assessment through Tests | : | 20 marks |

| | | |
|---|---|---|
| **Total Continuous Assessment** | : | **40 marks** |
| **End Semester Examination** | : | **60 marks** |
| **TOTAL** | : | **100 marks** |

**vii. CONTINUOUS ASSESSMENT TEST**

- No. of Tests: 02
- Maximum Marks: 30
- Test Duration: 1 ½ hours
- Topics: 2 ½ modules

**viii. END SEMESTER EXAMINATION**

- Maximum Marks: 60
- Exam Duration: 3 hours

| Course Code | Course Name | Category | L | T | P | J | Credit | Year of Introduction |
|---|---|---|---|---|---|---|---|---|
| 23IEL42H | GAME DEVELOPMENT | IEC | 3 | 0 | 0 | 0 | 3 | 2023 |

## i. COURSE OVERVIEW

This course introduces the fundamentals of game engines and explores the intricate art of game development. It emphasizes using the Unity platform, enabling students to gain a deeper understanding of programming concepts and the tools used while developing the games.

## ii. COURSE OUTCOMES

After the completion of the course, the student will be able to:

| Course Outcomes | Description | Level |
|---|---|---|
| CO1 | Explain the fundamentals of game engines and features of Unity IDE required to develop and deploy interactive video games | Understand |
| CO2 | Develop Scripting code with C# programming language | Apply |
| CO3 | Develop features of 2D games by using Unity IDE: Simple Movement and Input, Decisions and Flow Control, Object-Oriented Concepts and Functions, Exceptions and Debugging, Loops and Arrays | Apply |
| CO4 | Apply techniques to create dynamic audiovisual experiences that enhance gameplay using Unity IDE | Apply |
| CO5 | Explain the processes to successfully publish video games | Understand |

## iii. SYLLABUS

This course covers fundamental game design and C# coding skills such as:

Game Engines, Using the Unity framework, C# scripting concepts, Simple Movement and Input, Decisions and Flow Control, Object-Oriented Concepts and Functions, Exceptions and Debugging, Loops and Arrays, Animation and sound effects, User Interfaces, Publishing Games

## iv (a) TEXT BOOKS

1. Hands-On Unity 2021 Game Development, 2nd Edition

2. Learning C# by Developing Games with Unity 2021

**(b) REFERENCES**

1. Unity 2021 Shaders and Effects Cookbook, 4th Edition

2. https://docs.unity3d.com/Manual/UnityManual.html

**v. COURSE PLAN**

| Module | Contents | Hours |
|---|---|---|
| **I** | **Game Engines** – Engine Concepts, Development Tools, Installation of Unity Software, IDE basics, Unity Concepts, Sprites. **Introduction to Scripting** – C# Language Concepts, Creating Scripts, C# coding Fundamentals, Game Loops and Functions. Data types and Variables, Mathematical Operations, Variable Scope and Access, Displaying Data<br><br>(Develop Pinball Scoring) | **8** |
| **II** | **Simple Movement and Input** – Simple Movement of objects, Rotation and Scaling, Easy Input Handling in Unity (Develop the game Alien Dance Squad). **Decisions and Flow Control** – Logical Expressions, If/Else statements, switch statements. Organizing Game Objects – Parent-Child Objects, Sorting Layers, Tagging Game Objects, Collision Layers.<br><br>(Develop the game Thunder Road/Mower Dodgeball) | **10** |
| **III** | **Object-Oriented Concepts and Functions-** Defining Classes, Creating and Using Classes, Defining Functions, Accessing Game Objects, Constructor and Property Functions. (Develop the game Deep Space). **Exceptions and Debugging-** Run-Time Exceptions, Finding Run-time Errors, Using the Debugger. | **10** |

| | | |
|---|---|---|
| | (Develop the game Bug Hunt) | |
| **IV** | **Loops and Arrays** -Arrays, for, foreach and while loops. **Animation and Sound Effects** -Simple Unity Animation, Animator States, Scripting Animations, Animations and Colliders, Adding Sounds to Game Objects, Scripting Sounds.<br><br>(Develop games Banana Breakout / RoboDash Animation) | **9** |
| **V** | **User Interfaces -** Unity Buttons, Other UI Controls, UI Design Concepts.<br><br>(Develop the game Space Creeps Settings)<br><br>**Publishing Games** - Splash Screens, Credit Scenes and Icons, Publishing to PC, Mac and Linux Computers, Publishing to Smartphones, Publishing to Game Consoles. | **8** |
| | **Total Hours** | **45** |

## vi. ASSESSMENT PATTERN

Continuous Assessment: End Semester Examination – 40: 60

| Continuous Assessment | | |
|---|---|---|
| Attendance | : | 5 marks |
| Assignments | : | 15 marks |
| Assessment through Tests | : | 20 marks |
| **Total Continuous Assessment** | : | **40 marks** |
| **End Semester Examination** | : | **60 marks** |
| **TOTAL** | : | **100 marks** |

## vii. CONTINUOUS ASSESSMENT TEST

- No. of Tests: 02
- Maximum Marks: 30
- Test Duration: 1 ½ hours

- Topics: 2 ½ modules

**viii. END SEMESTER EXAMINATION**

- Maximum Marks: 60
- Exam Duration: 3 hours

| Course Code | Course Name | Category | L | T | P | J | Credit | Year of Introduction |
|---|---|---|---|---|---|---|---|---|
| **23CSV48A** | **COMPREHENSIVE VIVA VOCE** | **PWS** | **0** | **0** | **2** | **0** | **1** | **2023** |

### i. COURSE OVERVIEW

The objective of this course is to evaluate the students' basic understanding and application capability in the core domains of their respective engineering branch.

### ii. COURSE OUTCOMES

After the completion of the course, the student will be able to:

| Course Outcomes | Description | Level |
|---|---|---|
| CO1 | Apply fundamental concepts from core engineering courses to analyze and solve basic engineering problems relevant to the branch of study. | Apply |
| CO2 | Demonstrate comprehensive technical understanding by articulating engineering concepts, reasoning, and solutions effectively during viva voce examination. | Understand |

### iii. SYLLABUS

**Formal Languages and Automata Theory:** Regular Grammar, Context Free Grammar, Context Sensitive Grammar (CSG), Unrestricted Grammar, Chomsky classification of formal languages.

**Computer Organization and Architecture:** Basic Structure of computers, Register transfer logic, Arithmetic algorithms, Control Logic Design, I/O organization: accessing of I/O devices.

**Operating Systems:** Introduction to Operating Systems, Process Management, Inter Process Communication, Process Synchronization, Memory Management, Storage Management, File System Interface, Virtual machines.

**Database Management Systems:** Introduction to Database and Entity Relationship (ER) Model, Relational Model, Normalization, Physical Data Organization, Transactions, Concurrency Control, Recovery and Recent Topics.

**Data Structures:** Basic Concepts of Data Structures, Arrays and Searching, Linked List and Memory Management, Trees and Graphs, Sorting and Hashing.

**Algorithm Analysis and Design:** Introduction to algorithm analysis, Advanced data structures and graph algorithms, Various algorithm design techniques, Introduction to complexity theory and approximation algorithms.

**Computer Networks:** Introduction to Computer Networks-Reference models- Application Layer - Transport Layer-User Datagram Protocol (UDP)- Transmission Control Protocol (TCP) - Network layer -Routing algorithms - Congestion Control Algorithms-Quality of Service (QoS)- The Link Layer- Wireless and Mobile Networks - Wi-Fi: 802.11 Wireless LANs. Mobile IP.

**iv (a) TEXT BOOKS**

1. Prescribed textbooks for the core courses from the first to the sixth semester.

**(b) REFERENCES**

1. Prescribed textbooks for the core courses from the first to the sixth semester.

**v. COURSE PLAN**

| | Contents | Hours |
|---|---|---|
| | Formal Languages and Automata Theory | 5 |
| | Computer Organization and Architecture | 4 |
| | Operating Systems | 4 |
| | Database Management Systems | 4 |
| | Data Structures | 4 |
| | Algorithm Analysis and Design | 5 |
| | Computer Networks | 4 |
| | **Total Hours** | **30** |

**vi) COURSE GUIDELINES AND EVALUATION**

i) The course should be mapped with a faculty and classes shall be arranged for practicing questions based on the core courses listed in the curriculum. Mock tests, mock viva voce sessions, technical discussions may be conducted by the faculty in charge during the hours assigned for the course.

ii) **Total marks: 50**

**Continuous internal evaluation-25 marks**

Attendance – 5 marks

Multiple choice questions test and comprehensive viva (minimum 2 each) - 20 marks

**Final viva voce examination-25 marks**

The mark will be treated as internal and should be uploaded along with internal marks of other courses.

The final viva voce examination shall be conducted by a panel of two evaluators. The panel shall consist of one senior faculty member from the Department and an expert from Industry/research institute/academia or two senior faculty members from the Department.

| Course Code | Course Name | Category | L | T | P | J | Credit | Year of Introduction |
|---|---|---|---|---|---|---|---|---|
| 23CSJ48A<br>23CSJ48B | PROJECT | PWS | 0 | 0 | 10 | 0 | 5 | 2023 |

**i. COURSE OVERVIEW** The aim of this course is to apply engineering knowledge in solving practical problems, to foster innovation in design of products, processes or systems, and to develop creative thinking in finding viable solutions to engineering problems. The course is mainly intended to evoke the innovation and invention skills of a student. The course will provide an opportunity to synthesize and apply the knowledge and analytical skills learned, to be developed as a prototype or simulation.

**Desirable:** The project outcome should be published in a peer-reviewed journal or presented at a conference, or a patent application should be filed.

It is recommended that the projects may be finalized in the thrust areas of the respective engineering stream or as interdisciplinary projects. Importance should be given to address societal problems and developing indigenous technologies.

**ii. COURSE OUTCOMES** After the completion of the course, the student will be able to:

| Course Outcomes | Description | Level |
|---|---|---|
| CO1 | Apply multidisciplinary knowledge to model and solve real world problems. | Apply |
| CO2 | Apply innovative and creative problem-solving to develop sustainable and socially relevant products, processes, or technologies. | Create |
| CO3 | Exhibit effective teamwork and leadership skills in diverse environments, with the ability to comprehend and carry out designated responsibilities. | Apply |
| CO4 | Execute tasks by planning effectively and utilizing available resources to meet deadlines, while adhering to ethical and professional standards. | Apply |

| | | |
|---|---|---|
| CO5 | Effectively document, present, and communicate technical and scientific outcomes in professional written and verbal formats. | Apply |

**iii. Evaluation Guidelines**

Total: 100 marks (Minimum required to pass: **50** Marks).

- Project progress evaluation by guide: **20** Marks.
- Two interim evaluations by the Evaluation Committee: **30** Marks (15 marks for each evaluation).
- Final evaluation by the Final Evaluation committee: **30** Marks
- Quality of the report evaluated by the evaluation committee: **20** Marks

**Interim Evaluation:**

The Evaluation committee comprises of HoD or a senior faculty member, Project coordinator and Project supervisor.

**Final Evaluation:**

The final evaluation committee comprises of Project coordinator, domain expert from the Department and expert from Industry/research/academic Institute / senior faculty from a sister department.

| Course Code | Course Name | Category | L | T | P | J | Credit | Year of introduction |
|---|---|---|---|---|---|---|---|---|
| 23CSI48A | INTERNSHIP | PWS | 0 | 0 | - | 0 | 5 | 2023 |

## COURSE OBJECTIVES

The internship is an integral component of the undergraduate programme aimed at providing students with immersive learning through practical exposure in industry, research organizations, or academic institutions.

## BENEFITS OF INTERNSHIP

### Benefits to Students

- Provides practical exposure to real-world industrial and organizational environments, enabling the application of theoretical knowledge gained in classrooms.
- Enhances professional competencies such as communication, teamwork, time management, interpersonal skills, and workplace ethics.
- Facilitates the development of technical and domain-specific skills, thereby strengthening overall professional readiness.
- Assists students in assessing their career interests and determining the suitability of a particular industry or profession.
- Improves employability by strengthening profiles for placements, higher education, and potential recruitment by the host organization.
- Encourages professional networking and relationship - building with industry experts and peers.
- Offers an opportunity to evaluate the organization and work culture before committing to full-time employment.

### Benefits to the Institute

- Strengthens industry - academia collaboration.
- Facilitates smoother and more effective placement processes.
- Enhances institutional credibility and brand value.
- Supports student engagement and retention.
- Enables curriculum updates based on industry and student feedback.
- Improves the overall teaching - learning process through industry relevance.

### Benefits to the Industry

- Provides access to a pool of motivated, job-ready students who can contribute immediately.
- Offers a cost-effective mechanism to evaluate and recruit potential employees.
- Enables the availability of a flexible workforce for temporary, project-based, or seasonal requirements.

- Brings fresh perspectives and innovative approaches to problem-solving.
- Enhances organizational visibility and employer branding within academic institutions.
- Strengthens corporate image by contributing to education and skill development.

**TYPES OF INTERNSHIPS**

- Industry Internship with/without Stipend
- Government / PSU Internship (BARC/Railway/ISRO etc.)
- Internship with prominent education/ Research Institutes
- Internship with Incubation centres /Start-ups

**COURSE OVERVIEW**

Students may undertake an internship in an industry, research organization, or reputed academic institution with prior approval from the respective Head of the Department.

The internship is designed to promote meaningful skill development through structured, outcome-driven experiential learning. It shall focus on clearly defined technical competencies, such as domain-specific skills, software proficiency, exposure to industry-standard tools, and structured engineering problem-solving tasks. The learning objectives of the internship, along with the expectations of the host organization and the academic requirements of the institution, shall be clearly defined in consultation with the host organization and mutually agreed upon prior to the approval of the internship.

Each student shall be assigned a faculty guide/supervisor for monitoring and evaluation. The internship shall be relevant to the student's stream of study and can be carried out in Semester **VII** or Semester **VIII**, as specified by the Department, for a minimum duration of **three months.**

A student shall be permitted to undertake the internship only after the respective semester registration. During the internship period, any other courses or academic activities shall be pursued in online mode or as specified by the Department, to ensure timely fulfillment of all academic requirements.

**COURSE OUTCOMES**

After the completion of the course, the student will be able to:

| Course Outcomes | Description | Level |
|---|---|---|
| CO1 | Apply theoretical knowledge and engineering principles to practical problems encountered in professional practice. | Apply |

| | | |
|---|---|---|
| CO2 | Demonstrate technical competence and understanding of tools, techniques, and processes relevant to the chosen field of specialization. | Apply |
| CO3 | Analyse social, environmental, economic, safety, and administrative factors influencing industrial operations and decision-making processes. | Analyze |
| CO4 | Communicate effectively through technical reports, presentations, and professional interactions in an industrial or research environment. | Apply |
| CO5 | Apply professional ethics, teamwork, and adaptability while performing assigned tasks in a multidisciplinary work setting. | Apply |

**iv) GUIDELINES FOR STUDENTS**

- Duration of internship is three to six months (One semester).
- Students may undertake mini projects, case studies, or related technical tasks during the internship with the prior approval of the competent authority at the host organisation.
- Students shall strictly adhere to the rules, regulations, code of conduct, and working hours prescribed by the host organisation.
- Prior permission shall be obtained from the host organisation before using or reproducing any data, documents, drawings, photographs, or proprietary information for academic purposes.
- Students shall follow all ethical practices, confidentiality requirements, and Standard Operating Procedures (SOPs) of the host organisation.
- Students shall comply with all health, safety, and environmental guidelines prescribed by the host organisation during the internship period.
- Students shall maintain regular contact with the assigned faculty guide/supervisor and submit weekly progress updates on the work carried out.
- Each student shall maintain a diary/logbook recording daily activities, learning outcomes, and progress throughout the internship period.
- On completion of the internship, students shall submit the following documents to the Department:
  - ✓ Internship report detailing the work carried out and learning outcomes
  - ✓ Internship Completion Certificate issued by the host organisation
  - ✓ Feedback from the employer
  - ✓ Proof of stipend received, if applicable

**v) ASSESSMENT PATTERN**

The marks awarded for the Internship will be based on the following:

(i) Evaluation done by the industry,
(ii) Student's diary,

(iii) Internship report, and
(iv) Internship viva voce.

| | | |
|---|---|---|
| **Continuous Assessment** | | |
| Student's diary/ Daily Log | : | 25 marks |
| Evaluation done by the industry | : | 25 marks |
| **Total Continuous Assessment** | : | **50 marks** |
| **Final Assessment** | | |
| Internship Report | : | 25 Marks |
| Internship Viva Voce | : | 25 marks |
| **Total Final Assessment** | : | **50 marks** |
| **TOTAL** | : | **100 marks** |

**Student's Diary/ Daily Log**

The purpose of maintaining a daily diary is to cultivate systematic documentation habits and to encourage students to record observations, impressions, information gathered and suggestions, if any, during the internship period. The diary shall contain a day-to-day record of activities, learning experiences, technical details, and relevant sketches or drawings related to the work carried out. The daily diary shall be signed regularly by the industry supervisor and shall be verified and ratified by the faculty guide during the interim review.

The interim review shall be conducted midway through the internship by the Internship Review Committee, comprising the internship coordinator, faculty supervisor, and a senior faculty member of the Department. The review may be conducted in online or offline mode, based on the feasibility of the student's physical presence on campus.

**Internship Report**

On completion of the internship, each student shall prepare and submit a comprehensive internship report to the faculty supervisor. The report shall present a systematic account of the activities undertaken, observations made and knowledge gained during the training period. Students may consult the industry supervisor during the preparation of the final report, subject to compliance with the confidentiality policies and norms of the host organisation. The completed report shall be duly certified and signed by the Industry Supervisor, Faculty guide and the Head of the Department.

The internship report shall be evaluated based on the following criteria:

- Originality and technical content
- Adequacy, clarity and relevance of the written presentation

- Organization, format, use of drawings, sketches, language and overall style
- Variety and relevance of learning experiences documented
- Demonstration of practical applications and linkage with theoretical concepts covered in the curriculum

**Evaluation done by the industry**

The performance of the student during the internship shall be evaluated by the industry supervisor or the person-in-charge using a prescribed evaluation format provided by the institution. The evaluation shall cover key parameters such as professional behaviour, technical competence, learning ability, initiative, quality of work, communication skills, teamwork, discipline, time management, and overall professional attitude.

The faculty guide shall share the evaluation format with the host organisation and coordinate the assessment process. The completed evaluation form shall be duly signed and sealed by the industry supervisor and submitted to the Department as part of the internship assessment records.

**Internship Viva Voce**

The viva voce examination shall be conducted by the Internship Review Committee. The committee shall assess the student's understanding of the internship work, technical competence, learning outcomes, and professional orientation.

# MINOR

| Course Code | Course Name | Category | L | T | P | J | Credit | Year of introduction |
|---|---|---|---|---|---|---|---|---|
| 23CSJ4Mx(A/C/E/G/I) | MINIPROJECT | VAC | 0 | 0 | 6 | 0 | 3 | 2023 |

## i) COURSE OVERVIEW

The course encourages students to **apply the concepts, methods, and tools** learned to solve **real-world or socially relevant problems.** The course also emphasizes **innovation, and critical thinking**, encouraging students to explore new ideas and technologies. The mini project topic selected will be assigned to a group of three/four students, under the guidance of a Supervisor. This is expected to provide a good initiation for the student(s) in R&D work. It helps students develop skills in teamwork, project planning, technical documentation, and communication.

### iii) COURSE OUTCOMES

After the completion of the course, the student will be able to:

| Course Outcomes | Description | Level |
|---|---|---|
| CO1 | Identify real life engineering problems that are socially relevant, technically feasible and economically viable. | Apply |
| CO2 | Design proper scientific methodology to successfully complete the project | Apply |
| CO3 | Develop solutions to socially relevant practical problems by applying suitable scientific tools | Apply |
| CO4 | Evaluate the performance of the developed solution using suitable data analysis, validation techniques, and engineering judgement. | Evaluate |
| CO5 | Prepare a technical report and present the project outcomes effectively using appropriate engineering and communication tools | Apply |
| CO6 | Build the culture of working effectively in a team, upholding professional and ethical responsibilities | Apply |

### ii) COURSE PLAN

In this course, each group consisting of three/four members is expected to identify a topic of interest in consultation with Faculty-in-charge of mini project, review the literature and gather information pertaining to the chosen topic, state the objectives and develop a methodology to achieve the objectives. Execute experimental procedure, design/fabrication or develop codes/programs or conduct case studies to achieve the objectives. Demonstrate the novelty of the project through the results and outcomes.

The progress of the mini project is evaluated based on a minimum of two reviews. The review committee may be constituted by the Head of the Department. A project report is submitted by each student at the end of the semester.

**ASSESSMENT PATTERN**

The final evaluation will be conducted as an internal evaluation based on the level of objectives achieved, the report and a viva-voce examination, conducted by a 3-member committee appointed by Head of the Department offering minor. The committee members shall be HoD or a senior faculty member, Mini project coordinator and project supervisor. The Committee will evaluate the level of completion and demonstration of functionality/specifications, presentation, oral examination, working knowledge and involvement. The progress of the mini project is evaluated through a minimum of TWO reviews. At the time of the 1st review, students are expected to propose a methodology to achieve the objectives after completing a thorough literature study of the existing systems under their chosen area. In the 2nd review students are expected to highlight the implementation details of the proposed solution.

**Mark Distribution:**

Total: 100 marks (Minimum required to pass : 50 Marks).

- Project progress evaluation by guide : 20 Marks.
- Two interim evaluations by the Evaluation Committee : 30 Marks (15 marks for each evaluation).
- Final evaluation by the Final Evaluation committee : 30 Marks
- Quality of the report evaluated by the evaluation committee: 20 Marks

**HONOURS**

**Honour Basket 1: SECURITY IN COMPUTING**

| Course Code | Course Name | Category | L | T | P | J | Credit | Year of Introduction |
|---|---|---|---|---|---|---|---|---|
| 23CSL4HA | CYBER FORENSICS | VAC | 3 | 0 | 0 | 0 | 3 | 2023 |

## i. COURSE OVERVIEW

Cyber Forensics course provides a comprehensive understanding of cybercrimes, their investigation, and prevention through digital forensics and ethical hacking. It covers the evolution of traditional computer crimes, the role of information and communication technologies, and the classification of cyber offenses. The course also introduces ethical hacking concepts such as foot printing, scanning, system and network hacking, and web application security. Emphasis is placed on real-world case studies, Indian cyber laws, and hands-on practices to develop the skills required to detect, investigate, and mitigate cyber threats effectively.

## ii. COURSE OUTCOMES

After the completion of the course, the student will be able to:

| Course Outcomes | Description | Level |
|---|---|---|
| CO1 | Explain the traditional computer crimes, their classification, and the evolving role of ICT in cybercrime, along with the fundamental concepts and techniques of cyber forensics and digital investigation. | Understand |
| CO2 | Explain cyber forensic principles, including evidence collection, forensic duplication, data acquisition, and use of forensic technologies and tools. | Understand |
| CO3 | Apply and validate digital evidence using appropriate forensic techniques such as data hiding detection, network forensics, and mobile device investigations, ensuring admissibility under cyber laws. | Apply |
| CO4 | Apply ethical hacking methodologies to identify vulnerabilities through foot printing, scanning, enumeration, and system exploitation in a controlled and ethical manner. | Apply |
| CO5 | Explain security measures to protect web applications, networks, and mobile platforms against attacks such as SQL injection, denial of service, and session hijacking. | Understand |

## iii. SYLLABUS

**Cyber Crime and Forensics** – Fundamentals of cybercrime, its classification, and the role of ICT in modern digital offenses. It also introduces cyber forensics, investigation steps, forensic techniques, and data acquisition methods.

**Evidence Collection and Forensics Tools** – Evidence collection and forensic tools, including processing digital crime scenes and handling various evidence sources. It also covers file systems, registry artifacts, forensic software/hardware tools.

**Analysis and Validation** – Validating and analyzing digital evidence, uncovering hidden data, and performing network, email, and mobile forensics. It also covers the admissibility of evidence, relevant

cyber laws in India, and real-world case studies.

**Ethical Hacking** – Ethical hacking fundamentals, including network scanning, enumeration, and system hacking techniques. It also covers malware threats, sniffing, email tracking etc. Web- and network-focused attacks such as social engineering, denial of service, session hijacking, and attacks on web servers and applications including SQL injection.

**iv (a) TEXT BOOKS**

1. Bill Nelson, Amelia Phillips, Christopher Steuart, Guide to Computer Forensics and Investigations‖, Cengage Learning, India Sixth Edition, 2019.
2. CEH Official Certified Ethical Hacking Review Guide, Wiley India Edition, Version 11, 2021.

**(b) REFERENCES**

1. Dejey, S. Murugan – Cyber Forensics, Oxford University Press, India, 2018
2. John R.Vacca, "Computer Forensics ", Cengage Learning, 2005
3. MarjieT.Britz, "Computer Forensics and Cyber Crime: An Introduction 3rd Edition, Prentice Hall, 2013.

**v. COURSE PLAN**

| Module | Contents | Hours |
|---|---|---|
| **I** | **Unit I:** Introduction To Cyber Crime And Forensics Introduction to Traditional Computer Crime, Traditional problems associated with Computer Crime. Role of ECD and ICT in Cybercrime – Classification of Cyber Crime. The Present and future of Cybercrime – Cyber Forensics -Steps in Forensic Investigation – Forensic Examination Process Types of CF techniques – Forensic duplication and investigation – Forensics Technology and Systems – Understanding Computer Investigation – Data Acquisition. | **10** |
| **II** | **Unit II:** Evidence Collection And Forensics Tools Processing Crime and Incident Scenes – Digital Evidence – Sources of Evidence -Working with File Systems. – Registry – Artifacts – Current Computer Forensics Tools: Software/ Hardware Tools – Forensic Suite – Acquisition and Seizure of Evidence from Computers and Mobile Devices Chain of Custody- Forensic Tools | **10** |
| **III** | **Unit III:** Analysis And Validation Validating Forensics Data – Data Hiding Techniques – Performing Remote Acquisition – Network Forensics – Email Investigations – Cell Phone and Mobile Devices Forensics – Analysis of Digital Evidence – Admissibility of Evidence – Cyber Laws in India – Case Studies | **10** |
| **IV** | **Unit IV:** Ethical Hacking Introduction to Ethical Hacking – Footprinting and Reconnaissance – Scanning Networks Enumeration – System Hacking – Malware Threats – Sniffing – Email Tracking | **8** |
| **V** | **Unit V:** Ethical Hacking In Web Social Engineering – Denial of Service – Session Hijacking – Hacking Web servers – Hacking Web Applications – SQL Injection – Hacking Wireless Networks – Hacking Mobile Platforms. | **7** |
| | **Total Hours** | **45** |

## vi. ASSESSMENT PATTERN

Continuous Assessment: End Semester Examination – 40: 60

| Continuous Assessment | | |
|---|---|---|
| Attendance | : | 5 marks |
| Assignments | : | 15 marks |
| Assessment through Tests | : | 20 marks |
| **Total Continuous Assessment** | : | **40 marks** |
| **End Semester Examination** | : | **60 marks** |
| **TOTAL** | : | **100 marks** |

## vii. CONTINUOUS ASSESSMENT TEST

- No. of Tests: 02
- Maximum Marks: 30
- Test Duration: 1 ½ hours
- Topics: 2 ½ modules

## viii. END SEMESTER EXAMINATION

- Maximum Marks: 60
- Exam Duration: 3 hours

## Honour Basket 2: MACHINE LEARNING

| Course Code | Course Name | Category | L | T | P | J | Credit | Year of Introduction |
|---|---|---|---|---|---|---|---|---|
| 23CSL4HC | REINFORCEMENT LEARNING | VAC | 3 | 0 | 0 | 0 | 3 | 2023 |

i. **PRE-REQUISITE :** Nil

ii. **COURSE OVERVIEW**

This course covers fundamental principles and techniques in reinforcement learning. Reinforcement learning is concerned with building programs that learn how to predict and act in a stochastic environment, based on past experience. Applications of reinforcement learning range from classical control problems, such as power plant optimization or dynamical system control, to game playing, inventory control, and many other fields. Topics include Markov decision process, dynamic programming, Monte Carlo, temporal difference, function approximation reinforcement learning algorithms, and applications of reinforcement learning. This course enables the leaners to apply reinforcement learning on real world applications and research problems.

iii. **COURSE OUTCOMES**

After the completion of the course the student will be able to

| Course Outcomes | Description | Level |
|---|---|---|
| CO1 | Solve computational problems using probability and random variables. | Apply |
| CO2 | Apply policy iteration and value iteration reinforcement learning algorithms. | Apply |
| CO3 | Apply policy evaluation and Monte Carlo control methods to learn optimal policies in reinforcement learning. | Apply |
| CO4 | Apply temporal-difference reinforcement learning algorithms. | Apply |
| CO5 | Apply on-policy and off-policy reinforcement learning algorithms with function approximation. | Apply |

## iv. SYLLABUS

Review Of Probability Concepts, Markov Decision Process, Finite Markov Decision Processes, Prediction And Control, Temporal-Difference (TD) Methods For Model Free Prediction And Control, Sarsa, Function Approximation Method, Eligibility Traces ,Policy Gradient Methods, The Policy Gradient Theorem, REINFORCE: Monte Carlo Policy Gradient, REINFORCE with Baseline, Actor– Critic Methods.

## v. (a) TEXT BOOKS

1. Richard S. Sutton and Andrew G. Barto, Reinforcement Learning: An Introduction, 2nd Edition

## (b) REFERENCES

1. Reinforcement Learning: State-of-the-Art, Marco Wiering and Martijn van Otterlo, Eds
2. Algorithms for Reinforcement Learning, Szepesvari (2010), Morgan & Claypool.
3. Artificial Intelligence: A Modern Approach, Stuart J. Russell and Peter Norvig
4. Mathematical Statistics and Data Analysis by John A. Rice,University of California, Berkeley, Third edition, published by Cengage.
5. Machine Learning: A Probabilistic Perspective, Kevin P. Murphy

## vi. COURSE PLAN

| Module | Contents | No. of Hours |
|---|---|---|
| I | **Review of Probability Concepts**<br>Axioms of probability, concepts of random variables, Probability mass function, Probability density function, Cumulative density functions, Expectation of random variables, Joint and multiple random variables,<br>Conditional and marginal distributions Correlation and independence | **8** |
| II | **Markov Decision Process**<br>Introduction to Reinforcement Learning(RL) terminology - Examples of RL, Elements of RL, Limitations and Scope of RL, Finite Markov Decision Processes, The Agent Environment Interface Goals and Rewards, Returns and Episodes, Policies and | **10** |

| | | |
|---|---|---|
| | Value Functions, Optimal Policies and Optimal Value Functions, Optimal Policies and Optimal Value Functions | |
| **III** | **Prediction And Control**<br>Policy Evaluation (Prediction), Policy Improvement, Policy Iteration, Value Iteration, Monte Carlo Prediction, Monte Carlo Estimation of Action Values, Monte Carlo Control, Monte Carlo Control without Exploring Starts, Off-policy Prediction via Importance Sampling, Incremental Implementation, Off-policy Monte Carlo Control | **10** |
| **IV** | **Temporal-Difference (Td) Methods TB-1**<br>TD Prediction, Advantages of TD Prediction Methods<br>Optimality of TD (0), Sarsa: On-policy TD Control, Q-learning: Off-policy TD Control, Expected Sarsa, n-step TD Prediction, n-step Sarsa, n-step Off-policy Learning Off-policy Learning Without Importance Sampling: The n-step Tree Backup Algorithm | **9** |
| **V** | **Function Approximation Method TB-1**<br>Value-function Approximation, The Prediction Objective, Stochastic- gradient Methods, Linear Methods, The Lambda-return , TD(Lambda), n- step Truncated Lambda-return Methods, Sarsa(Lambda), Policy Approximation and its Advantages, The Policy Gradient Theorem, REINFORCE: Monte Carlo Policy Gradient REINFORCE with Baseline,Actor–Critic Methods | **8** |
| | **Total hours** | **45** |

## vii. ASSESSMENT PATTERN

Continuous Assessment: End Semester Examination – 40: 60

| Continuous Assessment | | |
|---|---|---|
| Attendance | : | 5 marks |
| Assignments | : | 15 marks |
| Assessment through Tests | : | 20 marks |
| **Total Continuous Assessment** | : | **40 marks** |
| **End Semester Examination** | : | **60 marks** |
| **TOTAL** | : | **100 marks** |

## viii. CONTINUOUS ASSESSMENT TEST

- No. of Tests: 02
- Maximum Marks: 30
- Test Duration: 1 ½ hours
- Topics: 2 ½ modules

## ix. END SEMESTER EXAMINATION

- Maximum Marks: 60
- Exam Duration: 3 hours

**Honour Basket 3: FORMAL METHODS**

| Course Code | Course Name | Category | L | T | P | J | Credit | Year of Introduction |
|---|---|---|---|---|---|---|---|---|
| 23CSL3HE | LOGIC FOR COMPUTER SCIENCE | VAC | 3 | 0 | 0 | 0 | 3 | 2023 |

**i. PRE-REQUISITE:** Basic knowledge of **discrete mathematics** and **fundamental programming concepts.**

**ii. COURSE OVERVIEW**

This course introduces logic as a foundation for computer science, covering induction, Boolean algebra, propositional and first order logic. It emphasizes formal reasoning, deduction systems, and proof techniques such as resolution and unification. Students gain practical exposure to logic-based programming tasks including normal-form conversion and automated reasoning, forming a basis for applications in AI and formal verification.

**iii. COURSE OUTCOMES**

After the completion of the course, the student will be able to:

| Course Outcomes | Description | Level |
|---|---|---|
| CO1 | Explain the foundations of logic, including mathematical and structural induction and Boolean algebra. | Understand |
| CO2 | Construct and interpret propositional logic formulas and analyze their semantics, including satisfiability and validity. | Apply |
| CO3 | Apply deduction systems and proof techniques in propositional logic, including resolution and proof normalization. | Apply |
| CO4 | Formulate and analyze first order logic expressions using proof-theoretic techniques such as unification and Skolemization. | Apply |
| CO5 | Analyze first order theories using concepts from model theory, including completeness, compactness, and logic-based reasoning techniques. | Apply |

**iv. SYLLABUS**

This syllabus presents Mathematical and structural induction; Boolean algebra; propositional logic—syntax, semantics, satisfiability, validity, and deduction systems; proof normalization and resolution; first order logic—syntax, semantics, unification, Skolemization, and Horn clauses; introduction to model theory; completeness and compactness theorems; first order theories; logic programming exercises including normal forms, tautology checking, and binary decision diagrams.

**v(a)TEXTBOOKS**

1. The Essence of Logic. John Kelly. Prentice-Hall International, 1997. ISBN 81-203-1190-6.

**(b) REFERENCES**

1. Logic for Computer Science Steve Reeves and Michael Clarke. Addison-Wesley, 1990. ISBN: 0-201-41643-3
2. Logic for Computer Science. Jean H. Gallier. Harper and Row, New York, 1986.
3. First-Order Logic and Automated Theorem Proving. Melvin Fitting. Springer Verlag, Berlin, 1990.
4. A Mathematical Introduction to Logic. Herbert B. Enderton. Academic Press, New York, 1972.
5. Natural Deduction (A Proof-theoretical study). Dag Prawitz. Almqvist and Wiskell, 1965.

**vi. COURSE PLAN**

| Module | Contents | Hours |
|---|---|---|
| I | **Foundations of Logic and Induction**:- Review of the Principle of Mathematical Induction; Strong induction and applications in computer science; Structural Induction and its use in reasoning about recursive definitions; Review of Boolean Algebra; Boolean operations and identities; Boolean expressions and simplification; Applications of Boolean algebra in digital logic and computation. | **8** |
| II | **Propositional Logic – Syntax and Semantics**:- Syntax of propositional formulas; Formation rules and well-formed formulas; Semantics of propositional logic; Truth assignments and truth tables; Logical equivalence Satisfiability, validity, and inconsistency; Normal forms- Conjunctive Normal Form (CNF), Disjunctive Normal Form (DNF). | **9** |
| III | **Deduction Systems and Propositional Proofs:-** Deduction systems for propositional logic; Natural deduction and axiomatic systems; Soundness of deduction systems; Completeness of propositional logic; Proof strategies and proof normalization; Resolution principle; Tautology checking. | **9** |
| IV | **First Order Logic and Proof Theory:-** Introduction to First Order Logic (FOL); Syntax and semantics of FOL; Quantifiers and interpretations; Free and bound variables; Proof theory for FOL; Skolemization Unification; Conversion to Horn clauses. | **10** |
| V | **Model Theory and Advanced Topics:-** Introduction to Model Theory; Structures, interpretations, and models; First order theories; Completeness Theorem; Compactness Theorem; Binary Decision Diagrams (BDDs); Applications of logic in automated reasoning and verification. | **9** |
| | **Total Hours** | **45** |

## vi. ASSESSMENT PATTERN

Continuous Assessment: End Semester Examination – 40: 60

| Continuous Assessment | | |
|---|---|---|
| Attendance | : | 5 marks |
| Assignments | : | 15 marks |
| Assessment through Tests | : | 20 marks |
| **Total Continuous Assessment** | : | **40 marks** |
| **End Semester Examination** | : | **60 marks** |
| **TOTAL** | : | **100 marks** |

## vii. CONTINUOUS ASSESSMENT TEST

- No. of Tests: 02
- Maximum Marks: 30
- Test Duration: 1 ½ hours
- Topics: 2 ½ modules

## viii. END SEMESTER EXAMINATION

- Maximum Marks: 60
- Exam Duration: 3 hours

# SEMESTER VIII

# PROGRAM ELECTIVE IV

| Course Code | Course Name | Category | L | T | P | J | Credit | Year of Introduction |
|---|---|---|---|---|---|---|---|---|
| 23CSL44A | VIRTUALIZATION | PEC | 2 | 1 | 0 | 0 | 3 | 2023 |

**i. PRE-REQUISITE: Operating Systems(23CSL20D)**

**ii. COURSE OVERVIEW**

This course introduces the fundamental concepts and technologies of virtualization that enable efficient utilization of computing resources in modern IT infrastructures. It covers hardware, server, desktop, network, and storage virtualization, along with hypervisors and virtualization architectures. The course also familiarizes students with industry-standard virtualization tools and platforms through practical use cases, preparing them to understand and deploy virtualized environments in data centers and cloud systems.

**iii. COURSE OUTCOMES**

After the completion of the course, the student will be able to:

| Course Outcomes | Description | Level |
|---|---|---|
| CO1 | Explain the concept of virtualization and cloud computing, the need for virtualization, its benefits and limitations, and the types of hardware virtualization and hypervisors. | Understand |
| CO2 | Describe virtual machine basics, server and desktop virtualization approaches, their types, and considerations for selecting virtualization platforms. | Understand |
| CO3 | Explain the concepts, advantages, functions, and tools of network virtualization, including VLANs and WAN virtualization architectures. | Understand |
| CO4 | Describe storage virtualization techniques and underlying storage architectures used in virtualized environments. | Understand |
| CO5 | Use industry-standard virtualization tools to configure virtual machines and analyze virtualization solutions through case studies. | Apply |

**iv. SYLLABUS**

Introduction to virtualization and cloud computing, including hardware, server, desktop, network, and storage virtualization concepts. Study of hypervisors, virtualization architectures, SAN/NAS/RAID, and enterprise virtualization tools such as VMware, AWS, Hyper-V, VirtualBox, and Google platforms with case studies.

**v(a)TEXTBOOKS**

1. Anthony T.Velte , Toby J. Velte Robert Elsenpeter -Cloud computing a practical approach , TATA McGraw- Hill , New Delhi – 2010

2. James E. Smith, Ravi Nair, "Virtual Machines: Versatile Platforms for Systems and Processes", Elsevier/Morgan Kaufmann, 2005.

3. David Marshall, Wade A. Reynolds, Advanced Server Virtualization: VMware and Microsoft Platform in the Virtual Data Center, Auerbach

**(b) REFERENCES**

1. *Chris Wolf and Erick M. Halter ,Virtualization: From the Desktop to the Enterprise, APress, 2005.*

2. Rajkumar Buyya, James Broberg, Andrzej Goscinski-Cloud Computing (Principles and Paradigms), John Wiley & Sons, Inc. 2011

3. Dan C. Marinescu, *Cloud Computing: Theory and Practice*, Morgan Kaufmann (relevant virtualization foundations).

4. IBM Redbooks, *IBM PowerVM Virtualization*, IBM Corporation.

5. VMware Inc., VMware vSphere Documentation, Official VMware Publications.

**Corresponding MOOC course**: https://onlinecourses.nptel.ac.in/noc21_cs15/

**vi) COURSE PLAN**

| Module | Contents | Hours |
|---|---|---|
| **I** | **Introduction to Virtualization:** Virtualization and cloud computing – Need of virtualization – cost, administration, fast deployment, reduce infrastructure cost – limitations- Types of hardware virtualization: Full | **8** |

| | | |
|---|---|---|
| | virtualization – partial virtualization – Paravirtualization-Types of Hypervisors | |
| **II** | **Server and Desktop Virtualization:** Virtual machine basics- Types of virtual machines- Understanding Server Virtualization- types of server virtualization- Business Cases for Server Virtualization – Uses of Virtual Server Consolidation – Selecting Server Virtualization Platform-Desktop Virtualization-Types of Desktop Virtualization | **10** |
| **III** | **Network Virtualization:** Introduction to Network Virtualization-Advantages-Functions-Tools forNetwork Virtualization-VLAN-WAN Architecture-WAN Virtualization | **10** |
| **IV** | **Storage Virtualization:** Memory Virtualization-Types of Storage Virtualization-Block, File-Address space Remapping-Risks of Storage Virtualization-SAN-NAS-RAID | **9** |
| **V** | **Virtualization Tools:** VMWare-Amazon AWS-Microsoft HyperV- Oracle VM Virtual Box – IBM PowerVM- Google Virtualization- Case study. | **8** |
| | **Total Hours** | **45** |

## vii. ASSESSMENT PATTERN

Continuous Assessment: End Semester Examination – 40: 60

| Continuous Assessment | | |
|---|---|---|
| Attendance | : | 5 marks |
| Assignments | : | 15 marks |
| Assessment through Tests | : | 20 marks |
| **Total Continuous Assessment** | : | **40 marks** |
| **End Semester Examination** | : | **60 marks** |
| **TOTAL** | : | **100 marks** |

## viii. CONTINUOUS ASSESSMENT TEST

- No. of Tests: 02
- Maximum Marks: 30
- Test Duration: 1 ½ hours
- Topics: 2 ½ modules

**ix. END SEMESTER EXAMINATION**

- Maximum Marks: 60
- Exam Duration: 3 hours

| Course Code | Course Name | Category | L | T | P | J | Credit | Year of Introduction |
|---|---|---|---|---|---|---|---|---|
| 23CSL44B | APPROXIMATION ALGORITHMS | PEC | 2 | 1 | 0 | 0 | 3 | 2023 |

## i. COURSE OVERVIEW

The course provides a comprehensive understanding of algorithmic techniques for solving NP-hard optimization problems where exact solutions are computationally infeasible, focusing on the design and analysis of algorithms that produce near-optimal solutions with provable performance guarantees, covering fundamental concepts such as approximation ratios and complexity of optimization problems, greedy strategies, linear programming relaxation and rounding methods, primal–dual and randomized techniques, approximation schemes including PTAS and FPTAS, and the hardness of approximation, enabling students to analyse trade-offs between solution quality and computational efficiency and to apply suitable approximation techniques to real-world optimization problems.

## ii. COURSE OUTCOMES

After the completion of the course, the student will be able to:

| Course Outcomes | Description | Level |
|---|---|---|
| CO1 | Explain the fundamentals of NP-hard optimization problems, approximation approaches and performance guarantees. | Understand |
| CO2 | Apply greedy approximation methods and their approximation bounds for standard optimization problems. | Apply |
| CO3 | Apply linear programming relaxation, integrality gap analysis and rounding-based approximation techniques | Apply |
| CO4 | Examine primal–dual frameworks and randomized methods used in approximation algorithms. | Apply |
| CO5 | Apply approximation schemes and inapproximability results to identify the theoretical limits of efficient approximation algorithms. | Apply |

**iii. SYLLABUS**

Approximation algorithms for NP-hard optimization problems, performance guarantees and approximation ratios, greedy and basic approximation techniques, linear programming relaxation and rounding methods, primal–dual and randomized approximation algorithms, approximation schemes including PTAS and FPTAS, hardness of approximation and limits of efficient approximation algorithms.

**iv(a)TEXTBOOKS**


1. Vijay V. Vazirani, *Approximation Algorithms*, Springer-Verlag, 2001.
2. David P. Williamson and David B. Shmoys, *The Design of Approximation Algorithms*, Cambridge University Press, 2011.
3. Sanjeev Arora and Boaz Barak, *Computational Complexity: A Modern Approach*, Cambridge University Press, selected chapters on approximation algorithms.


**(b) REFERENCES**


1. Dorit S. Hochbaum (Ed.), *Approximation Algorithms for NP-Hard Problems*, PWS Publishing, 1997.
2. Michael R. Garey and David S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness*, W.H. Freeman, relevant sections.
3. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest and Clifford Stein, *Introduction to Algorithms*, MIT Press, chapters on approximation algorithms.


**v) COURSE PLAN**

| Module | Contents | Hours |
|---|---|---|
| I | **Introduction to Approximation Algorithms**<br>Optimization problems and their classification, minimization and maximization problems, NP-hard and NP-complete optimization problems, need and motivation for approximation algorithms, definition of approximation algorithms, performance guarantee and approximation ratio, absolute and relative approximation, lower bounds on optimal solutions. | **9** |
| II | **Greedy Approximation Algorithms**<br><br>Greedy paradigm for designing approximation algorithms, analysis of greedy strategies, approximation algorithms for Vertex Cover problem, Set Cover problem, scheduling problems, proof of approximation bounds, tightness of approximation ratios, limitations of greedy techniques. | **10** |
| III | **Linear Programming Based Approximation**<br><br>Linear programming formulation of optimization problems, integer programming and LP relaxation, integrality gap, rounding techniques, LP-based approximation algorithms, applications to Vertex Cover, Set Cover and Knapsack problems, analysis of approximation ratios. | **9** |

| | | |
|---|---|---|
| **IV** | **Primal–Dual and Randomized Techniques**<br><br>Primal–dual schema for approximation algorithms, design and analysis of primal–dual algorithms, applications to Set Cover and Facility Location problems, introduction to randomized algorithms, randomized rounding techniques, overview of semidefinite programming. | **8** |
| **V** | **Approximation Schemes and Hardness of Approximation**<br><br>Polynomial Time Approximation Schemes (PTAS), Fully Polynomial Time Approximation Schemes (FPTAS), approximation schemes for Knapsack problem, hardness of approximation, inapproximability results, limits of efficient approximation algorithms. | **9** |
| | Total Hours | **45** |

## vi. ASSESSMENT PATTERN

Continuous Assessment: End Semester Examination – 40: 60

| Continuous Assessment | | |
|---|---|---|
| Attendance | : | 5 marks |
| Assignments | : | 15 marks |
| Assessment through Tests | : | 20 marks |
| **Total Continuous Assessment** | : | **40 marks** |
| **End Semester Examination** | : | **60 marks** |
| **TOTAL** | : | **100 marks** |

## vii. CONTINUOUS ASSESSMENT TEST

- No. of Tests: 02
- Maximum Marks: 30
- Test Duration: 1 ½ hours
- Topics: 2 ½ modules

## viii. END SEMESTER EXAMINATION

- Maximum Marks: 60
- Exam Duration: 3 hours

| Course Code | Course Name | Category | L | T | P | J | Credit | Year of Introduction |
|---|---|---|---|---|---|---|---|---|
| 23CSL44C | NATURAL LANGUAGE PROCESSING | PEC | 2 | 1 | 0 | 0 | 3 | 2023 |

## i. COURSE OVERVIEW

This course introduces the fundamentals of Natural Language Processing, covering linguistic basics to advanced deep learning models. It equips students to build NLP applications and explore recent trends like large language models and ethical AI.

## ii. COURSE OUTCOMES

After the completion of the course, the student will be able to:

| Course Outcomes | Description | Level |
|---|---|---|
| CO1 | Explain the foundational concepts of Natural Language Processing and apply basic linguistic techniques such as tokenization, stemming, and POS tagging. | Understand |
| CO2 | Apply statistical and machine learning approaches to perform tasks like text classification, language modeling, and named entity recognition. | Apply |
| CO3 | Explain sentence structures and perform syntactic and semantic processing using parsing techniques and lexical resources. | Understand |
| CO4 | Use deep learning models for NLP tasks using embeddings, RNNs, and transformer-based architectures. | Apply |
| CO5 | Apply the current NLP technologies, including large language models, and discuss ethical considerations in their deployment. | Apply |

## iii. SYLLABUS

Fundamentals of NLP and Linguistic Concepts: NLP applications and challenges – Linguistic levels: morphology – syntax – semantics – pragmatics – Word sense disambiguation – Semantic role labeling – Coreference resolution – Discourse analysis – Lexical resources: WordNet – FrameNet. Text Processing and Preprocessing Techniques: Tokenization – Stemming – Lemmatization – Stop word removal – Normalization – Part-of-Speech (POS) tagging – Bag-of-Words – TF-IDF. Statistical and Traditional Machine Learning Approaches: N-gram models – Smoothing – Perplexity – Text classification using Naive Bayes – Logistic Regression – Support Vector Machines (SVM) –

Hidden Markov Models (HMMs) – Conditional Random Fields (CRFs). Parsing and Syntactic Structures: Constituency parsing – Dependency parsing – Context-Free Grammars (CFGs) – Probabilistic CFGs – Syntax-semantics interface. Deep Learning for NLP: Word2Vec – GloVe – fastText – Contextual embeddings – Feedforward networks – RNNs – LSTMs – GRUs – Sequence-to-sequence models – Attention mechanisms – Transformer architecture – BERT – GPT – T5 – Fine-tuning – Transfer learning – Prompt engineering. Evaluation, Applications, and Ethical Considerations: Accuracy – Precision – Recall – F1-score – BLEU – ROUGE – Bias – Fairness – Privacy – Safety – Responsible deployment of NLP systems and LLMs.

**iv) REFERENCES**


1. Speech and language processing (3rd ed., draft). Jurafsky, D., & Martin, J. H. (2023). Pearson Education.
2. Natural language processing with Python: Analyzing text with the natural language toolkit. Bird, S., Klein, E., & Loper, E. (2009). O'Reilly Media.
3. Deep learning for natural language processing. Goyal, P., Pandey, S., & Jain, K. (2018). Apress.
4. Natural Language Processing with Transformers. Tunstall, L., von Werra, L., & Wolf, T. (2022). O'Reilly Media.
5. Fairness and Machine Learning: Limitations and Opportunities. Barocas, S., Hardt, M., & Narayanan, A. (2023). MIT Press.
6. Conversational AI: Dialogue Systems, Conversational Agents, and Chatbots. McTear, M. (2020). Morgan & Claypool Publishers.


**v) COURSE PLAN**

| Module | Contents | Hours |
|---|---|---|
| **I** | Introduction to NLP: Definition, Importance, Applications, Challenges in NLP, Levels of NLP: Phonology, Morphology, Syntax, Semantics, Pragmatics, Discourse. Linguistic Background: Words, Sentences, Grammar, Syntax and Parsing, Morphological Analysis: Lemmatization vs. Stemming, POS (Part of Speech) Tagging. Text Processing Techniques: Tokenization, Sentence Segmentation, Normalization, Stop-word Removal.<br><br>Corpora and Annotation: Types of Corpora, Annotation Standards and Schemes. | **9** |

| | | |
|---|---|---|
| **II** | N-Grams and Language Models: Unigrams, Bigrams, Trigrams, Smoothing Techniques: Laplace, Good-Turing, Perplexity and Evaluation. Text Classification: Bag of Words and TF-IDF, Naïve Bayes Classifier, Logistic Regression and SVMs, Evaluation Metrics: Precision, Recall, F1 Score. Sequence Labelling Tasks: Named Entity Recognition (NER), | **9** |
| **III** | Syntactic Parsing: Dependency Parsing, Constituency Parsing, CYK Algorithm and Parse Trees. Semantic Analysis: Word Sense Disambiguation, Semantic Role Labelling, Lexical Resources: WordNet, VerbNet. Discourse and Pragmatics: Coreference Resolution, Discourse Coherence and Structure. Knowledge Representation: First-order Predicate Logic. | **9** |
| **IV** | Neural Networks for NLP: Word Embeddings: Word2Vec, GloVe, FastText, Recurrent Neural Networks (RNNs), LSTMs, GRUs. Sequence-to-Sequence Models: Encoder-Decoder Architectures, Attention Mechanism, Applications: Text Summarization, Machine Translation. Contextual Embeddings: ELMo, | **9** |
| **V** | Transformer Architecture, BERT and Variants (RoBERTa, DistilBERT). Text Generation: Language Generation Techniques, Text Completion and Dialogue Systems.<br>Large Language Models (LLMs): GPT Series (GPT-3, GPT-4), Instruction-tuned Models and Prompt Engineering, Few-shot, Zero-shot Learning. | **9** |
| | Total Hours | **45** |

## vi. ASSESSMENT PATTERN

Continuous Assessment: End Semester Examination – 40: 60

| Continuous Assessment | | |
|---|---|---|
| Attendance | : | 5 marks |
| Assignments | : | 15 marks |
| Assessment through Tests | : | 20 marks |
| **Total Continuous Assessment** | : | **40 marks** |
| **End Semester Examination** | : | **60 marks** |
| **TOTAL** | : | **100 marks** |

## vii. CONTINUOUS ASSESSMENT TEST

- No. of Tests: 02
- Maximum Marks: 30

- Test Duration: 1 ½ hours
- Topics: 2 ½ modules

**viii. END SEMESTER EXAMINATION**

- Maximum Marks: 60
- Exam Duration: 3 hours

| Course Code | Course Name | Category | L | T | P | J | Credit | Year of Introduction |
|---|---|---|---|---|---|---|---|---|
| 23CSL44D | DIGITAL FORENSICS | PEC | 2 | 1 | 0 | 0 | 3 | 2023 |

## i. COURSE OVERVIEW

This course introduces the principles and practices used to investigate cyber crimes and analyze digital evidence. It covers the nature of cyber crimes, types of digital evidence, and the forensic process involved in collecting, preserving, and analyzing electronic data. Students gain an understanding of the legal and ethical aspects governing digital investigations and the admissibility of evidence in courts. The syllabus also familiarizes learners with commonly used forensic tools and techniques for data recovery and analysis. Practical aspects such as handling computer, network, and mobile device evidence are emphasized. Overall, the course equips students with foundational knowledge and skills required for real-world digital forensic investigations.

## ii. COURSE OUTCOMES

After the completion of the course, the student will be able to:

| Course Outcomes | Description | Level |
|---|---|---|
| CO1 | Explain the principles and concepts of digital forensics. | Understand |
| CO2 | Explain various types of cyber crimes | Understand |
| CO3 | Analyze computer architectures, file systems, and operating systems relevant to digital forensics investigations. | Apply |
| CO4 | Explain the legal and ethical considerations associated with digital forensics, including the admissibility of digital evidence in court. | Understand |
| CO5 | Utilize popular forensic tools and software for digital investigations. | Apply |

## iii. SYLLABUS

Fundamentals of digital forensics, including its definition, scope, branches, and applications in law enforcement and cybersecurity, along with cybercrime concepts, types of cybercrimes, internet crimes, and handling of electronic evidence. It covers legal, ethical, and procedural aspects such as

e-discovery, admissibility of digital evidence, collection, preservation, documentation, forensic reporting, and expert testimony. Forensic tools including disk imaging, data recovery, slack space analysis, EnCase, FTK, vulnerability assessment tools, and anti-forensics with countermeasures. Computer and mobile forensics processes such as investigation workflows, multimedia evidence analysis, SIM and damaged data recovery, memory forensics, retrieval of deleted, renamed, compressed, and ghosted files, and forensic analysis of smartphones and tablets. Advanced topics include network, email, cloud, and IoT forensics, incident response and forensic readiness, cybercrime investigation procedures, and real-world case studies such as data breaches, financial fraud, ransomware attacks, and mobile device crime investigations.

**iv(a)TEXTBOOKS**


1. C. Altheide & H. Carvey, "Digital Forensics with Open Source Tools", Syngress
2. John Sammons "The Basics of Digital Forensics", Syngress-2012
3. Brain Carrier "File System Forensic Analysis", Addison-Wesley-2005
4. Harlan Carvey "Advanced Digital Forensic Analysis of the Windows Registry", Syngress-2014
5. Diane Barrett "Virtualization and Forensics - A Digital Forensic Investigator's Guide to Virtual Environments", Syngress-2010
6. B. Nelson, A. Phillips, and C. Steuart "Guide to Comter Forensics and Investigations", Cengage-2019


**(b) REFERENCES**


1. Jones, K. J., Bejtlich, R., & Rose, C. W. *Real Digital Forensics: Computer Security and Incident Response*. Addison-Wesley-2006
2. Vacca, J. R**.** *Computer Forensics: Computer Crime Scene Investigation*. Cengage Learning, **2012**.


**v) COURSE PLAN**

| Module | Contents | Hours |
|---|---|---|
| **I** | Introduction to digital forensics, definition and scope of digital forensics Different Branches of Digital Forensics Importance and applications of digital forensics in law enforcement and cybersecurity. Definition and types of cybercrimes Electronic evidence and handling, electronic media, collection, searching and storage of electronic media, Introduction to internet crimes. | **10** |
| **II** | Understanding of legal aspects and their impact on digital forensics, Electronics discovery Overview of legal and ethical issues in digital forensics. Types of digital evidence (e.g., documents, emails, logs). Collection, preservation, and documentation of digital evidence. Preparing forensic reports. Providing expert testimony in court. Admissibility of digital evidence in court. | **9** |
| **III** | Introduction to Forensic Tools Usage of Slack space Tools for Disk Imaging, Data Recovery, Vulnerability Assessment Tools, Encase and FTK tools Anti-Forensics and probable counters, Retrieving information. Retrieving deleted data: desktops, laptops and mobiles. Retrieving data | **9** |

| | | |
|---|---|---|
| | from slack space, renamed file, ghosting, compressed files. | |
| **IV** | Process of computer forensics and digital investigations, Processing of digital evidence, digital images, damaged SIM and data recovery, multimedia evidence. Techniques for analysing and extracting information from computer memory, Forensic analysis of smartphones and tablets. | **9** |
| **V** | Advanced Digital Forensics and Case Studies, Network Forensics - Email Forensics and Email Header Analysis, Log File Analysis and Traffic Analysis ,Cloud Forensics, IoT Forensics, Incident Response and Forensic Readiness, Cyber Crime Investigation Procedures, Digital Forensic Case Studies, Data Breach Investigation, Financial Fraud Case Study, Ransomware Attack Analysis, Mobile Device Crime Investigation. | **8** |
| | Total Hours | **45** |

## vii. ASSESSMENT PATTERN

Continuous Assessment: End Semester Examination – 40: 60

| Continuous Assessment | | |
|---|---|---|
| Attendance | : | 5 marks |
| Assignments | : | 15 marks |
| Assessment through Tests | : | 20 marks |
| **Total Continuous Assessment** | : | **40 marks** |
| **End Semester Examination** | : | **60 marks** |
| **TOTAL** | : | **100 marks** |

## viii. CONTINUOUS ASSESSMENT TEST

- No. of Tests: 02
- Maximum Marks: 30
- Test Duration: 1 ½ hours
- Topics: 2 ½ modules

## ix. END SEMESTER EXAMINATION

- Maximum Marks: 60
- Exam Duration: 3 hours

# PROGRAM ELECTIVE V

| Course Code | Course Name | Category | L | T | P | J | Credit | Year of Introduction |
|---|---|---|---|---|---|---|---|---|
| 23CSL45A | ALGORITHMIC GAME THEORY | PEC | 2 | 1 | 0 | 0 | 3 | 2023 |

## i. COURSE OVERVIEW

The objectives of this course are to:

1. To introduce the foundations of utility theory and strategic-form games.
2. To explain fundamental solution concepts and equilibrium notions in games.
3. To develop the ability to apply mixed strategies and Nash equilibrium concepts.
4. To understand algorithmic and computational aspects of equilibrium computation.
5. To apply game-theoretic principles to auctions, mechanism design, and matching markets.

## ii. COURSE OUTCOMES

After the completion of the course, the student will be able to:

| Course Outcomes | Description | Level |
|---|---|---|
| CO1 | Explain utility theory and represent strategic interactions using strategic-form games. | Understand |
| CO2 | Explain dominance, equilibrium, and security-based solution concepts in games. | Understand |
| CO3 | Apply mixed strategies and Nash equilibrium concepts to analyse strategic games. | Apply |
| CO4 | Apply algorithmic techniques to understand equilibrium computation in special classes of games. | Apply |
| CO5 | Apply game-theoretic models to auctions, mechanisms, and matching markets. | Apply |

## iii. SYLLABUS

Utility theory and von Neumann–Morgenstern axioms, Strategic (normal) form games and examples, Solution concepts: dominance, Nash equilibrium, and maximin, Zero-sum games and relationships among solution concepts, Mixed strategies and mixed extensions of games, Nash's Theorem, Complexity of Nash equilibrium computation, Two-player zero-sum games, minimax theorem, and LP duality, Improvement dynamics and potential games, Monderer–Shapley characterisation,

Congestion games and PLS-completeness, Graphical games and 0/1 polymatrix games, Hardness and tractable subclasses of games, Efficiency of equilibria (Price of Anarchy, Price of Stability), Auction theory: first-price and second-price auctions, incentive compatibility, Mechanism design: Groves, pivotal, Bailey–Cavallo mechanisms, Green–Laffont result, Stable matching, matching markets, and sponsored search.

**iv(a)TEXTBOOKS**

1. Nisan, N., Roughgarden, T., Tardos, É., & Vazirani, V. V. (Eds.)**,** *Algorithmic Game Theory*. Cambridge University Press, Cambridge, 2007.
2. Roughgarden, T. *Twenty Lectures on Algorithmic Game Theory*. Cambridge University Press, Cambridge, 2016.

**(b) REFERENCES**

1. Prakash, H., Narayanam, R., Garg, D., & Narahari, Y.**,** *Game Theoretic Problems in Network Economics and Mechanism Design Solutions*. Springer, London, 2014.
2. Krishna, V.**,** *Auction Theory*. 2nd Edition, Academic Press (Elsevier), 2010.
3. Vohra, R. V**.** *Mechanism Design: A Linear Programming Approach*. Cambridge University Press, Cambridge, 2011.

**v) COURSE PLAN**

| Module | Contents | Hours |
|---|---|---|
| **I** | **Utility Theory and Strategic Form Games :**Basics of utility theory, Von Neumann–Morgenstern axioms, Characterisation theorem, Strategic form (normal form) games, Representation of games and payoff matrices, Examples of strategic form games | **9** |
| **II** | **Solution Concepts in Strategic Games**: Dominance-based solution concepts (Strict dominance, Weak dominance), Iterated elimination of dominated strategies, Stability-based solution concepts (Nash equilibrium (pure strategies)), Security-based solution concepts (Maximin strategies), Relationship among dominance, equilibrium, and maximin, Special case: Two-player zero-sum games | **9** |
| **III** | **Mixed Strategies and Nash Equilibrium :** Mixed extension of strategic form games, Mixed strategies and expected utility, Solution concepts based on mixed strategies, Nash equilibrium in mixed strategies, Nash's Theorem (existence of equilibrium), Examples and applications | **9** |
| **IV** | **Computational Aspects and Special Game Classes :** Complexity of computation of Nash equilibrium, Two-player zero-sum games, Minimax theorem, Linear programming formulation, Correspondence between minimax theorem and strong duality in LP, Improvement dynamics, Potential games, Monderer–Shapley characterisation, Congestion games, Computation of pure strategy equilibria, PLS-completeness | **9** |
| **V** | **Graphical Games, Efficiency, Auctions, and Matching :** Graphical games, 0/1 polymatrix games, Hardness of computing equilibria, Subclasses of games with efficient equilibrium computation, Efficiency of equilibria( Price of Anarchy,Price of Stability) | **9** |

| | | |
|---|---|---|
| | Introduction to auction theory (First-price auction, Second-price auction, Incentive compatibility)<br><br>Introduction to mechanism design (Groves mechanism, Pivotal mechanism, Bailey–Cavallo mechanism, Green–Laffont result),Stable matching, Matching markets, Sponsored search | |
| | **Total Hours** | **45** |

## vii. ASSESSMENT PATTERN

Continuous Assessment: End Semester Examination – 40: 60

| Continuous Assessment | | |
|---|---|---|
| Attendance | : | 5 marks |
| Assignments | : | 15 marks |
| Assessment through Tests | : | 20 marks |
| **Total Continuous Assessment** | : | **40 marks** |
| **End Semester Examination** | : | **60 marks** |
| **TOTAL** | : | **100 marks** |

## viii. CONTINUOUS ASSESSMENT TEST

- No. of Tests: 02
- Maximum Marks: 30
- Test Duration: 1 ½ hours
- Topics: 2 ½ modules

## ix. END SEMESTER EXAMINATION

- Maximum Marks: 60
- Exam Duration: 3 hours

| Course Code | Course Name | Category | L | T | P | J | Credit | Year of Introduction |
|---|---|---|---|---|---|---|---|---|
| 23CSL45B | LARGE LANGUAGE MODELS | PEC | 2 | 1 | 0 | 0 | 3 | 2023 |

**i. PRE-REQUISITE:** 23CSL30C Artificial Intelligence

**ii. COURSE OVERVIEW**

The course introduces learners to Large Language Models by covering the basics of language models along with the evolution of language models. It then provides an in-depth understanding of Transformer architecture, the core building block of most Large Language Models (LLMs). The syllabus further focuses on exploring various LLM architectures and techniques like BERT, prompt engineering, and fine-tuning. Learners are equipped with the ability to evaluate LLM performance and identify potential biases Finally, the course discusses practical applications of LLMs, their challenges and limitations, and ethical considerations such as bias, responsible AI, and societal impact.

**iii. COURSE OUTCOMES**

After the completion of the course, the student will be able to:

| Course Outcomes | Description | Level |
|---|---|---|
| CO1 | Explain the fundamental concepts and an in-depth understanding of Transformer architecture, the core building block of Large Language Models (LLMs) | Understand |
| CO2 | Explain various LLM architectures and techniques. | Understand |
| CO3 | Apply prompt engineering techniques for effective LLM interaction and understand the concept of Retrieval Augmented Generation (RAG) and its role in LLMs. | Apply |
| CO4 | Summarize the metrics for evaluating LLM performance. | Understand |
| CO5 | Outline the applications and limitations of large language models in real-world scenarios. | Understand |

**iv. SYLLABUS**

Introduction to deep learning and Natural Language Processing, Text preprocessing, Bag of Words, TF-IDF, word embeddings, Attention mechanism, Transformer architecture, encoder–decoder models, Large Language Models, GPT, BERT, pretraining, fine-tuning, prompt engineering, Applications of LLMs.

**v(a)TEXTBOOKS**

1. Generative AI with LangChain: Build large language model (LLM) apps with Python, ChatGPT, and other LLMs, Ben Auffarth, Packt Publishing, 2023.

2. Modern Generative AI with ChatGPT and OpenAI Models, Valentina Alto**,** Packt Publishing, 2023.

3. Hands-On Large Language Models, Jay Alammar, Maarten Grootendorst**,** O'Reilly, 2023.

**(b) REFERENCES**

1. Introduction to Transformers for NLP: With the Hugging Face Library and Models to Solve Problems, Shashank Mohan Jain**,** Apress 2022.

2. Speech and Language Processing, Daniel Jurafsky & James H. Martin**,** Pearson/Prentice Hall, Second Edition, 2008.

3. Transformers for Natural Language Processing, Denis Rothman, Packt Publishing, First Edition, 2021.

4. Foundations of Statistical Natural Language Processing, Christopher D. Manning & Hinrich Schütze, MIT Press, 1999.

**vi) COURSE PLAN**

| Module | Contents | Hours |
|---|---|---|
| I | **Classical Language Modeling (CLM)** - n-grams, smoothing, class-based, brown clustering, Neural Language Modeling (NLM) - Word Embeddings, Word2Vec, FeedForward Neural.<br><br>**Understanding the core building block of most LLMs** - the Transformer model, Tokenization, Decoding the Transformer's components: encoders, decoders, attention mechanisms - types, Self-attention vs Flash Attention, feed-forward layer, Reinforcement Learning with AI Feedback (RLAIF), Reinforcement Learning from Human Feedback (RLHF) | **10** |

| | | |
|---|---|---|
| **II** | **Language Models** - Evolution of LLMs, Diving into different LLM architectures - GPT, BERT, T5, LLaMA, Pretraining vs fine-tuning, Mixture of Experts (MoE), various benchmarks to evaluate LLMs. | **9** |
| **III** | **Prompt Engineering** - Introduction to prompt, examples of prompt, prompt engineering, prompt techniques, zero shot, one shot, few-shot learning, a chain of thought prompting, ReAct Prompting, self-consistency, Tree of thought, LLM based Agents, Large Action Models(LAMs) | **8** |
| **IV** | **Evaluating LLMs - Measuring Performance and Biases -** Learning about common metrics for evaluating LLM performance (e.g., perplexity, BLEU score). Understanding the challenges of bias and fairness in LLMs. Exploring techniques for mitigating bias in LLM development and evaluation, considering prompt design and data selection for RAG models: RAGAS | **9** |
| **V** | **Training and Fine-Tuning LLMs:** Datasets for LLMs, Tokenizers (BPE, WordPiece), Transfer learning, Fine-tuning strategies, Parameter-efficient tuning (LoRA, adapters – intro), Evaluation of LLMs.<br>**Applications of LLMs:** Chatbots & virtual assistants, Code generation, Text summarization, Question answering systems, Search & recommendation, LLMs in healthcare, finance, education. | **9** |
| | Total Hours | **45** |

## vii. ASSESSMENT PATTERN

Continuous Assessment: End Semester Examination – 40: 60

| Continuous Assessment | | |
|---|---|---|
| Attendance | : | 5 marks |
| Assignments | : | 15 marks |
| Assessment through Tests | : | 20 marks |
| **Total Continuous Assessment** | : | **40 marks** |
| **End Semester Examination** | : | **60 marks** |
| **TOTAL** | : | **100 marks** |

## viii. CONTINUOUS ASSESSMENT TEST

- No. of Tests: 02
- Maximum Marks: 30
- Test Duration: 1 ½ hours
- Topics: 2 ½ modules

**ix. END SEMESTER EXAMINATION**

- Maximum Marks: 60
- Exam Duration: 3 hours

| Course Code | Course Name | Category | L | T | P | J | Credit | Year of Introduction |
|---|---|---|---|---|---|---|---|---|
| 23CSL45C | HARDWARE SECURITY | PEC | 2 | 1 | 0 | 0 | 3 | 2023 |

## i. COURSE OVERVIEW

The course enables learners to understand the design and evaluation of hardware security primitives and their roles in functional security and protection against supply chain issues. This course focus on all forms of side channel attacks, infrastructure-oriented attacks and information leakage attacks. It helps learners to study system level attacks and countermeasures.

## ii. COURSE OUTCOMES

After the completion of the course, the student will be able to:

| Course Outcomes | Description | Level |
|---|---|---|
| CO1 | Explain layered organization of computing systems, concepts of hardware security and trust, hardware attacks, vulnerabilities, basic countermeasures and mathematical foundation of cryptography. | Understand |
| CO2 | Identify hardware trojans and its counter measures. | Apply |
| CO3 | Make use of hardware security primitives and their applications for securing modern devices. | Apply |
| CO4 | Identify the various types of side channel attacks. | Apply |
| CO5 | Explain system level attacks and its counter measures. | Understand |

## iii. SYLLABUS

Introduction, Hardware Trojans, Hardware Security Primitives, Side Channel Attacks, Test Oriented Attacks, System level Attacks and Counter Measures.

## iv(a)TEXTBOOKS

1. Mark Tehranipoor and Swarup Bhunia – “Hardware Security – A Hands on Learning Approach” Morgan Kauffman Publications, 2019.

2. Mark Tehranipoor, Nitin Pundir, Nidish Vashistha, Farimah Farahmandi. Hardware Security Primitives, Springer, December 2022.

**(b) REFERENCES**

1. Wael Badawy. Hardware Security: Principles, Threats and Countermeasures, First Edition, 2025

**v. COURSE PLAN**

| Module | Contents | Hours |
|---|---|---|
| **I** | Overview of a computing system, layers of computing system, what is hardware security, hardware security vs trust, attacks, vulnerabilities and countermeasures, conflict between security and test/debug.<br>Mathematical foundations for cryptography: public and private key cryptography, elliptic curve cryptography; introduction to embedded systems security. | **11** |
| **II** | **Hardware Trojans:** Introduction, SoC Design flow, Hardware Trojans, Hardware Trojans in FPGA designs, Hardware Trojan taxonomy, Trust Benchmarks, Counter measures against hardware trojans, experiments | **8** |
| **III** | **Hardware Security Primitives:** Introduction, Preliminaries, Physical unclonable functions, True Random number generator, design for anti-counterfeit, existing challenges, Primitive designs with emerging nano devices, experiments | **8** |
| **IV** | **Side Channel Attacks:** Introduction, Background on side channel attacks, power analysis attacks, Electromagnetic (EM) side channel attacks, fault injection attacks, timing attacks, covert channels. Side channel attacks on control units, functional units, caches and interconnects; design techniques to prevent side-channel attacks, experiments | **9** |
| **V** | **System level Attacks and Counter Measures:** Introduction, Background on SoC design, SoC security requirement, security policy enforcement, Secure SoC design process, Threat modelling. | **9** |
| | Total Hours | **45** |

## vi. ASSESSMENT PATTERN

Continuous Assessment: End Semester Examination – 40: 60

| Continuous Assessment | | |
|---|---|---|
| Attendance | : | 5 marks |
| Assignments | : | 15 marks |
| Assessment through Tests | : | 20 marks |
| **Total Continuous Assessment** | : | **40 marks** |
| **End Semester Examination** | : | **60 marks** |
| **TOTAL** | : | **100 marks** |

## vii. CONTINUOUS ASSESSMENT TEST

- No. of Tests: 02
- Maximum Marks: 30
- Test Duration: 1 ½ hours
- Topics: 2 ½ modules

## viii. END SEMESTER EXAMINATION

- Maximum Marks: 60
- Exam Duration: 3 hours

| Course Code | Course Name | Category | L | T | P | J | Credit | Year of Introduction |
|---|---|---|---|---|---|---|---|---|
| 23CSL45D | SOCIAL MEDIA ANALYSIS | PEC | 2 | 1 | 0 | 0 | 3 | 2023 |

**i) COURSE OVERVIEW**

- Familiarize the learners with the concept of social media analytics and understand its significance.
- Enable the learners to develop skills required for analysing the effectiveness of social media.
- Familiarize the learners with different tools and visualization techniques for Social media analysis.

**ii. COURSE OUTCOMES**

After the completion of the course, the student will be able to:

| Course Outcomes | Description | Level |
|---|---|---|
| CO1 | Explain the concept of social media Analysis and its significance. | Understand |
| CO2 | Explain the different types of Social Media Networks | Understand |
| CO3 | Apply the different techniques for social media text analysis | Apply |
| CO4 | Apply the Search Engine tools to analyse the social media data. | Apply |
| CO5 | Apply the fundamental perspectives and hands-on skills needed to work with social media data. | Apply |

**iii. SYLLABUS**

Core Characteristics of Social Media, Types of Social Media, Social media landscape, Need for Social Media Analytics (SMA), SMA in small & large organizations. Basics of Social Network Structure - Nodes, Edges & Tie Describing the Networks Measures - Degree Distribution, Density, Connectivity, Centralization etc. Social Media Text Analysis - Types of Social Media Text, Purpose of Text Analysis, Steps in Text Analysis. Location Analysis - Sources of Location Data, Categories of Location Analysis, Location Analysis and Privacy Concerns. Social Sharing and filtering, Automated Recommendation systems, Traditional Vs social Recommendation Systems

**iv.(a)TEXTBOOKS**

1. Seven Layers of Social Media Analytics_ Mining Business Insights from Social Media Text, Actions, Networks, Hyperlinks, Apps, Search Engine, and Location Data, Gohar F. Khan,(ISBN-10: 1507823207).

2. Mining the Social Web_ Analyzing Data from Facebook, Twitter, LinkedIn, and Other Social Media Sites, Matthew A Russell, O'Reilly

3. Charu Aggarwal (ed.), Social Network Data Analytics, Springer, 2011

**(b) REFERENCES**

1. Social Media Analytics [2015], Techniques and Insights for Extracting Business Value Out of Social Media, Matthew Ganis, Avinash Kohirkar, IBM Press.

2. Social Media Analytics Strategy Using Data to Optimize Business Performance, Alex Gonçalves, APress Business Team.

**v. COURSE PLAN**

| Module | Contents | Hours |
|---|---|---|
| **I** | **Core Characteristics of Social Media**, Types of Social Media, Social media landscape, Need for Social Media Analytics (SMA), SMA in small & large organizations. Purpose of Social Media Analysis, Social Media vs. Traditional Business Analysis, Seven Layers of Social Media Analysis, Types of Social Media Analysis, Social Media Analysis Cycle, Challenges to Social Media Analysis, Social Media Analysis Tools. | **9** |
| **II** | **Social Network Structure, Measures & Visualization-** Basics of Social Network Structure - Nodes, Edges & Tie Describing the Networks Measures - Degree Distribution, Density, Connectivity, Centralization, Tie Strength & Trust Network Visualization - Graph Layout, Visualizing Network features, Scale Issues. Social Media Network Analysis - Common Network Terms, Common Social Media Network Types, Types of Networks, Common Network Terminologies, Network Analysis Tools. | **10** |
| **III** | **Social Media Text, Action Analysis** - Social Media Text Analysis - Types of Social Media Text, Purpose of Text Analysis, Steps in Text Analysis, Social Media Text Analysis Tools, Social Media Action Analysis, Common Social Media Actions, Action Analytics Tools. | **8** |

| | | |
|---|---|---|
| **IV** | **Social Media Location & Search Engine-** Location Analysis - Sources of Location Data, Categories of Location Analysis, Location Analysis and Privacy Concerns, Location Analytics Tools, Search Engine Analysis - Types of Search Engines, Search Engine Analysis, Search Engine Analysis Tools | **9** |
| **V** | **Social Information Filtering** - Social Sharing and filtering, Automated Recommendation systems, Traditional Vs Social Recommendation Systems. Understanding Social Media and Business Alignment, Social Media KPI, Formulating a Social Media Strategy, Managing Social Media Risks. **Social media in public sector** - Analysing public sector social media, analysing individual users,Case study. | **9** |
| | Total Hours | **45** |

## vi. ASSESSMENT PATTERN

Continuous Assessment: End Semester Examination – 40: 60

| Continuous Assessment | | |
|---|---|---|
| Attendance | : | 5 marks |
| Assignments | : | 15 marks |
| Assessment through Tests | : | 20 marks |
| **Total Continuous Assessment** | : | **40 marks** |
| **End Semester Examination** | : | **60 marks** |
| **TOTAL** | : | **100 marks** |

## vii. CONTINUOUS ASSESSMENT TEST

- No. of Tests: 02
- Maximum Marks: 30
- Test Duration: 1 ½ hours
- Topics: 2 ½ modules

## viii. END SEMESTER EXAMINATION

- Maximum Marks: 60
- Exam Duration: 3 hours

| Course Code | Course Name | Category | L | T | P | J | Credit | Year of Introduction |
|---|---|---|---|---|---|---|---|---|
| 23CSL45E | QUANTUM COMPUTING AND PROGRAMMING | PEC | 2 | 1 | 0 | 0 | 3 | 2023 |

## i. COURSE OVERVIEW

This course provides a comprehensive introduction to the quantum mechanical principles, mathematics, and practical implementations of quantum computing. It also covers the core concepts in quantum information processing to write quantum computer codes in qiskit framework. The practical applications of quantum computing are demonstrated through Grover's Search algorithm, Variational quantum eigensolver and quantum random number generators.

## ii. COURSE OUTCOMES

After the completion of the course, the student will be able to:

| Course Outcomes | Description | Level |
|---|---|---|
| CO1 | Explain the physical and mathematical principles underlying quantum computing such as entanglement, quantum logic gates | Understand |
| CO2 | Explain the concepts of matrix representations of gates, quantum Fourier transform, and its standard implementations. | Understand |
| CO3 | Apply quantum circuits with X, H, CNOT gates to evaluate and interpret quantum computational processes. | Apply |
| CO4 | Apply qiskit for Grover search and Variational Quantum Eigensolver (VQE) problems | Apply |
| CO5 | Explain the applications of quantum computing in cryptography and machine learning. | Understand |

**iii. SYLLABUS**

Basics of Quantum mechanics for information processing; basic terminologies; classical vs. quantum computing; Mathematical tools in quantum computing; quantum gates, representations and applications in searching and factoring problems; Qiskit programming environment and debugging; New trends in quantum computing research.

**iv(a)TEXTBOOKS**


1. Quantum Mechanics for Scientists and Engineers, D. Miller, Cambridge University Press; 2008 (ISBN 978-0-521-89783-9).
2. Quantum Computing - a gentle introduction, E. Rieffel, W. Polak, The MIT Press, Cambridge, Massachusetts, 2014 (ISBN-13: 978-0262526678).
3. Quantum Computing for Programmers, R. Hundt, Cambridge University Press, 2022 (ISBN 978-1-009-09817-5).
4. Introduction to Classical and Quantum computing, T Wong, Rooted Grove Publication, 2022; (ISBN-13 : 979-8985593105).
5. Quantum Programming in Depth: Solving Problems With Q# and Qiskit, M. Mykhailova, Manning Publications, 2025; (ISBN-13: 978-1-63343-690-9).
6. Learn Quantum Computing with Python and IBM Quantum, R. Loredo, Packt Publications, 2025 (ISBN: 978-1-80324-480-8).
7. Programming Quantum Computers: Essential Algorithms and Code Samples, E. R. Johnston, N. Harrigan, M. Gimeno-Segovia, Oreilly, 2021 (ISBN:9781492039631).


**(b) REFERENCES**


1. Quantum Computing - Progress and Prospects, E.Grumbling and M. Horowitz, Editors, The National Academies Press (ISBN 978-981-96-5645-5).
2. Essentials for Deeper Understanding of Quantum Computing, R. Maezono, Springer, 2025 (ISBN: 9789819656462).
3. Introduction to Quantum Computing, R. LaPierre, Springer, 2021 (ISBN 978-3-030-69317-6).


**v) COURSE PLAN**

| Module | Contents | Hours |
|---|---|---|

| | | |
|---|---|---|
| **I** | Quantum mechanics for quantum information processing; Classical and quantum computing; key components of quantum computer; Schrodinger equation and its solutions (eg. SHO); bra-ket notations; Postulates; operators (unitary and Hermitian), observable and probabilities; Principles of superposition, entanglement, decoherence, interference. Quantum bits, types of qubits; gates and qudits; Bloch's sphere. | **9** |
| **II** | Mathematics for Quantum Computing; Church-Turing Thesis and its variants; Hilbert space; Linearity; Matrix representation of gates; No-Cloning theorem; Bell's inequality; Hadamard operator; Deutsch–Jozsa algorithm; quantum Fourier Transform; Grover's search algorithm; Shor's factoring algorithm; Variational quantum eigensolver. classical and quantum random number generators; | **9** |
| **III** | Quantum information processing; Quantum circuit model and its diagrammatic representations (qubit register, measurements operations etc.); teleportation; examples (unitary transformation gate; I, X operators) transpilation; single- and multi- qubit gates (eg. Pauli-X,Y,Z and and CNOT); Hadamard and T qugate; analysis of measuring with X $\|0\rangle$, H $\|0\rangle$, HH $\|0\rangle$, and CNOT gates; Measuring multiple qubits; Noise and error mitigation techniques; | **9** |
| **IV** | Programming Quantum Computer with Qiskit environment; IBM Quantum Composer; Qiskit; *QuantumCircuit,Statevector,Operator* classes; Defining Quantum and Classical Registers. Implementing gates and circuits (single-qubit gates (H, T, Pauli-X, Y, Z) and multi-qubit gates (CNOT, SWAP, Toffoli); Visualizing circuits; Statevector to inspect quantum states. Visualizing results; Simulation-based debugging; | **10** |
| **V** | Advanced concepts in quantum computing; introduction to Q# and quantum development kit, Pennylane and Cirq framework and its applications in cryptography and machine learning. Quantum key distribution in secure communications;Hybrid approaches and scalability challenges; Fault-tolerant quantum computing. | **8** |
| | **Total Hours** | **45** |

## vi. ASSESSMENT PATTERN

Continuous Assessment: End Semester Examination – 40: 60

| Continuous Assessment | | |
|---|---|---|
| Attendance | : | 5 marks |
| Assignments | : | 15 marks |
| Assessment through Tests | : | 20 marks |
| **Total Continuous Assessment** | : | **40 marks** |
| **End Semester Examination** | : | **60 marks** |
| **TOTAL** | : | **100 marks** |

**viii. CONTINUOUS ASSESSMENT TEST**

- No. of Tests: 02
- Maximum Marks: 30
- Test Duration: 1 ½ hours
- Topics: 2 ½ modules

**ix. END SEMESTER EXAMINATION**

- Maximum Marks: 60
- Exam Duration: 3 hours

# PROGRAM ELECTIVE VI

| Course Code | Course Name | Category | L | T | P | J | Credit | Year of Introduction |
|---|---|---|---|---|---|---|---|---|
| 23CSL46A | PARALLEL ALGORITHMS | PEC | 2 | 1 | 0 | 0 | 3 | 2023 |

**i. PRE-REQUISITE: Data Structures(23CSL20A)**

**ii. COURSE OVERVIEW**

This course introduces the fundamental principles and techniques used in the design and analysis of parallel algorithms. It covers parallel programming models, interconnection network topologies, and performance evaluation frameworks essential for understanding parallel computation. The course explores algorithm design techniques for PRAM models, algorithms for various interconnection networks, and parallel complexity theory, including lower bounds and complexity classes. Emphasis is placed on analyzing efficiency, scalability, and feasibility of parallel solutions for computational problems, preparing students to reason about high-performance and parallel computing systems.

**iii. COURSE OUTCOMES**

After the completion of the course, the student will be able to:

| Course Outcomes | Description | Level |
|---|---|---|
| CO1 | Explain parallel programming models, interconnection network topologies, and performance measures used for evaluating parallel algorithms. | Understand |
| CO2 | Describe the fundamental design techniques used in PRAM models for developing parallel algorithms. | Understand |
| CO3 | Develop parallel solutions for fundamental problems such as sorting, searching, list ranking, and graph algorithms. | Apply |
| CO4 | Describe algorithmic techniques used on various interconnection network topologies for parallel computation. | Understand |
| CO5 | Use lower bound analysis and parallel complexity classes to evaluate the performance limits of parallel algorithms. | Apply |

**iv. SYLLABUS**

**Parallel Algorithm Models**-PRAM, SIMD, MIMD, interconnection networks, **Design Techniques**-Divide & conquer, pipelining, pointer jumping, prefix sums, **Parallel Algorithms**-Sorting (bitonic, merge), list ranking, graph algorithms, **Interconnection Networks Algorithms, Parallel Complexity-** Lower bounds for PRAM models, the complexity class NC, P-completeness.

**v(a)TEXTBOOKS**

1. Joseph F. JáJá, *An Introduction to Parallel Algorithms,* Addison Wesley,1992

2. Barry Wilkinson & Michael Allen, *Parallel Programming*, Pearson,1998

**(b) REFERENCES**

1. Seyed H Roosta, *Parallel Processing and Parallel Algorithms: Theory and Computation*, Springer, 2000.

2. Fayez Gebali, *Algorithms and Parallel Computing*, Wiley, 2011.

3. Michael J Quinn, Parallel Computing: Theory and Practice‖, second edition, McGraw Hill.

**vi Corresponding MOOC course**: nptel.ac.in/courses/106103188

**vi) COURSE PLAN**

| Module | Contents | Hours |
|---|---|---|
| **I** | **Parallel Programming Models:** Shared memory Model-PRAM-MIMD-SIMD, Network Model-line-ring-mesh-hypercube, Performance measurement of parallel algorithms, The work time presentation framework of parallel algorithms | **8** |
| **II** | **Algorithm Design Techniques for PRAM Models:** Balancing, divide and conquer, parallel prefix computation, pointer jumping, symmetry breaking, pipelining, accelerated cascading. | **10** |
| **III** | **Algorithms for PRAM Models:** List ranking, sorting and searching, tree algorithms, Euler Circuits, Graph algorithms. | **10** |

| | | |
|---|---|---|
| **IV** | **Interconnection Networks Algorithms:** Sorting on a 2D Mesh, Sorting on 3D-Mesh, Hypercube Algorithms, Butterfly Networks, Benes Network, CCC | **9** |
| **V** | **Parallel Complexity:** Lower bound for maxima, Lower bound for merging**,** Lower bounds for PRAM models, the complexity class NC, P-completeness. | **8** |
| | Total Hours | **45** |

## vii. ASSESSMENT PATTERN

Continuous Assessment: End Semester Examination – 40: 60

| Continuous Assessment | | |
|---|---|---|
| Attendance | : | 5 marks |
| Assignments | : | 15 marks |
| Assessment through Tests | : | 20 marks |
| **Total Continuous Assessment** | : | **40 marks** |
| **End Semester Examination** | : | **60 marks** |
| **TOTAL** | : | **100 marks** |

## viii. CONTINUOUS ASSESSMENT TEST

- No. of Tests: 02
- Maximum Marks: 30
- Test Duration: 1 ½ hours
- Topics: 2 ½ modules

## ix. END SEMESTER EXAMINATION

- Maximum Marks: 60
- Exam Duration: 3 hours

| Course Code | Course Name | Category | L | T | P | J | Credit | Year of Introduction |
|---|---|---|---|---|---|---|---|---|
| 23CSL46B | BIG DATA ANALYTICS | PEC | 2 | 1 | 0 | 0 | 3 | 2023 |

## i. COURSE OVERVIEW

This course introduces the fundamental concepts and technologies of Big Data analytics. It covers the characteristics and challenges of Big Data, distributed storage and processing using the Hadoop ecosystem, and large-scale data processing with the MapReduce framework. The course also explores NoSQL and NewSQL databases for managing Big Data, with hands-on exposure to MongoDB and Cassandra. Further, it addresses stream processing, real-time analytics, graph analytics, and Big Data analytics frameworks such as Pig, Hive, HBase, and ZooKeeper. The course equips learners with the skills required to design, store, process, and analyze large-scale data for real-world applications.

## ii. COURSE OUTCOMES

After the completion of the course, the student will be able to:

| Course Outcomes | Description | Level |
|---|---|---|
| CO1 | Explain fundamental concepts of Big Data and its technologies | Understand |
| CO2 | Apply data analytics solutions using Hadoop ecosystems | Apply |
| CO3 | Apply concepts of MapReduce framework for optimization | Apply |
| CO4 | Apply appropriate NoSQL database models to store and process large volumes of structured and unstructured data. | Apply |
| CO5 | Apply stream processing, graph analytics, and Big Data analytics frameworks to build real-time data analytics applications. | Apply |

**iii. SYLLABUS**

Big Data concepts and characteristics, challenges of conventional data processing, types of Big Data, intelligent data analysis, traditional business intelligence vs. Big Data approach, Hadoop ecosystem and HDFS architecture, distributed storage and scalability, MapReduce programming model, job execution and optimization, NoSQL databases and data models, SQL vs. NoSQL vs. NewSQL, MongoDB and Cassandra fundamentals with CRUD operations and querying, stream processing concepts, real-time analytics, graph analytics, Big Data analytics frameworks using Pig and Hive, HiveQL, HBase and ZooKeeper fundamentals, industry Big Data platforms and applications.

**iv(a)TEXTBOOKS**

1. Seema Acharya, Subhashini Chellappan, “Big Data and Analytics”, Wiley Publication, 2015.

2. Tom White, “Hadoop: The Definitive Guide”, O’Reilly Publications, 2009.

3. Kyle Banker, “Mongo DB in Action”, Manning Publications Company, 2012.

4. Russell Bradberry, Eric Blow, “Practical Cassandra A developers Approach “, Pearson Education, 2014.

5. Bart Baesens, Analytics in a Big Data World: The Essential Guide to Data Science and its Applications, Wiley, 2014.

**(b) REFERENCES**

1. Hurwitz JS, Nugent A, Halper F, Kaufman M. Big data for dummies. John Wiley & Sons; 2013.

2. Bill Franks, "Taming the Big Data Tidal Wave: Finding Opportunities in Huge Data Streams with Advanced Analytics", Wiley, 2012.

3. Anand Rajaraman and Jeffrey D. Ullman, "Mining of Massive Datasets", Cambridge University Press, 2011.

4. Michael Minelli, Michele Chambers, and Ambiga Dhiraj, "Big Data, Big Analytics: Emerging Business Intelligence and Analytic Trends for Today’s Businesses", Wiley India, 2013.

5. Boris Lublinsky, Kevin T. Smith, and Alexey Yakubovich, "Professional Hadoop Solutions", Wiley, 2013.

6. Vignesh Prajapati, "Big Data Analytics with R and Hadoop", Packt Publishing, 2013.

7. Holden Karau, Andy Konwinski, Patrick Wendell, and Matei Zaharia, "Learning Spark: Lightning-Fast Big Data Analysis", O'Reilly Media, 2015.

8. Erik Brynjolfsson et al., The Second Machine Age: Work, Progress, and Prosperity in a Time of Brilliant Technologies, W. W. Norton & Company, 2014.

9. Leskovec, Rajaraman, Ullman, Mining of Massive Datasets, Cambridge University Press, 2014.

**v) COURSE PLAN**

| Module | Contents | Hours |
|---|---|---|
| **I** | **Introduction to Big Data-** Introduction to Big Data, Big Data characteristics, challenges of conventional data processing systems, types of Big Data (structured, semi-structured, unstructured), intelligent data analysis, traditional business intelligence vs. Big Data business approach, case studies of Big Data solutions. | **7** |
| **II** | **Hadoop Ecosystem and Distributed Storage Platforms -**History and Evolution of Apache Hadoop, Hadoop Ecosystem Overview, HDFS: Architecture and Design, Cluster Architecture and Compute Node Organization, Java Interfaces to HDFS (Basics), Scaling Out vs Scaling Up, Introduction to Apache Spark -comparison with Hadoop , Hadoop Streaming and Cluster Environment. | **9** |
| **III** | **MapReduce Framework and Data Processing -**MapReduce programming model, developing a MapReduce application, how MapReduce works, anatomy of a MapReduce job run, shuffle and sort phase, task execution, failures and fault tolerance, job scheduling, MapReduce types and input/output formats, MapReduce features, optimization concepts in MapReduce, Introduction to Spark RDD Model. | **9** |
| **IV** | **NoSQL Databases for Big Data Management -** Introduction to NoSQL, types of NoSQL databases, advantages of NoSQL databases, NewSQL databases, comparison of SQL vs. NoSQL vs. NewSQL. MongoDB: introduction, features, data types, MongoDB data model, MongoDB Query Language, CRUD operations, arrays and embedded documents, functions (count, sort, limit, skip, aggregation framework), MapReduce in MongoDB, cursors, indexing techniques, MongoDB import and export utilities. Cassandra: introduction, features, data types, CQL and CQLSH, keyspaces, | **10** |

| | | |
|---|---|---|
| | CRUD operations, collections, counters, TTL, alter commands, data import and export, querying system tables. | |
| **V** | **Stream Processing and Analytics Frameworks –** Stream Data Models and Architectures, Window-Based Processing Models, Sampling and Filtering Streams, Counting Distinct Elements, Real-Time Analytics Platforms (RTAP), Introduction to Apache Spark Streaming, Introduction to Apache Kafka, Real-Time Sentiment Analysis, Stock Market Prediction (Streaming Analytics Use Case), Graph Analytics Concepts, Applications using Apache Hive and Apache Pig, Fundamentals of HBase and Apache ZooKeeper, Enterprise Platforms: IBM InfoSphere BigInsights, IBM Streams | **10** |
| | Total Hours | **45** |

## vi. ASSESSMENT PATTERN

Continuous Assessment: End Semester Examination – 40: 60

| Continuous Assessment | | |
|---|---|---|
| Attendance | : | 5 marks |
| Assignments | : | 15 marks |
| Assessment through Tests | : | 20 marks |
| **Total Continuous Assessment** | : | **40 marks** |
| **End Semester Examination** | : | **60 marks** |
| **TOTAL** | : | **100 marks** |

## vii. CONTINUOUS ASSESSMENT TEST

- No. of Tests: 02
- Maximum Marks: 30
- Test Duration: 1 ½ hours
- Topics: 2 ½ modules

## viii. END SEMESTER EXAMINATION

- Maximum Marks: 60
- Exam Duration: 3 hours

| Course Code | Course Name | Category | L | T | P | J | Credit | Year of Introduction |
|---|---|---|---|---|---|---|---|---|
| 23CSL46C | BLOCKCHAIN AND CRYPTO CURRENCIES | PEC | 2 | 1 | 0 | 0 | 3 | 2023 |

## i. COURSE OVERVIEW

This course introduces the fundamental concepts of blockchain technology and cryptocurrencies, covering distributed ledger systems, cryptographic foundations, Bitcoin architecture, and Ethereum platform. It provides an understanding of smart contracts and Solidity programming, including the process of developing and deploying simple decentralized applications. The course also explores blockchain applications, security challenges, and emerging trends such as Layer 2 solutions and Web3, with emphasis on conceptual clarity and practical exposure suitable for undergraduate students.

## ii. COURSE OUTCOMES

After the completion of the course, the student will be able to:

| Course Outcomes | Description | Level |
|---|---|---|
| CO1 | Explain the fundamentals of distributed systems and the need for blockchain technology. | Understand |
| CO2 | Describe the cryptographic principles underlying blockchain technology. | Understand |
| CO3 | Explain the working of Bitcoin including transaction processing and consensus mechanisms. | Understand |
| CO4 | Develop and deploy simple smart contracts using Solidity on the Ethereum platform. | Apply |
| CO5 | Explain blockchain applications and security challenges in decentralized systems. | Understand |

## iii. SYLLABUS

Introduction to distributed ledger technology, blockchain structure, consensus mechanisms, and types of blockchains. Cryptographic primitives including hash functions, digital signatures, and Merkle trees. Working of Bitcoin and Ethereum, fundamentals of smart contracts and Solidity,

decentralized applications, blockchain applications across industries, and basic security challenges including 51% attacks and double spending.

**iv(a)TEXTBOOKS**

1. Arvind Narayanan, Joseph Bonneau, Edward Felten, Andrew Miller, and Steven Goldfeder, Bitcoin and Cryptocurrency Technologies: A Comprehensive Introduction, Princeton University Press, 2016.

2. Imran Bashir, Mastering Blockchain: A Deep Dive into Distributed Ledgers, Consensus Protocols, Smart Contracts, DApps, Cryptocurrencies, Ethereum, and More, 3rd Edition, Packt Publishing, 2020.

3. Chris Dannen, Introducing Ethereum and Solidity: Foundations of Cryptocurrency and Blockchain Programming for Beginners, Apress, 2017.

4. Andreas M. Antonopoulos, Mastering Bitcoin: Programming the Open Blockchain, 3rd Edition, O'Reilly Media, 2024.

**(b) REFERENCES**

1. Antony Lewis, The Basics of Bitcoins and Blockchains: An Introduction to Cryptocurrencies and the Technology that Powers Them, Mango Publishing, 2018.
2. Shaoen Wu, Song Guo, Junhao Hu, and Yi-Zhe Cheng (Eds.), Blockchain for Cybersecurity and Privacy: Architectures, Challenges, and Applications, CRC Press, 2020.
3. Henning Diedrich, Ethereum: Blockchains, Digital Assets, Smart Contracts, Decentralized Autonomous Organizations, Wildfire Publishing, 2016.
4. Elad Elrom, The Blockchain Developer: A Practical Guide for Designing, Implementing, Publishing, Testing, and Securing Distributed Blockchain-based Projects, Apress, 2019

**v) COURSE PLAN**

| Module | Contents | Hours |
|---|---|---|
| **I** | **Introduction to Blockchain -** Introduction to distributed systems and the need for distributed record keeping. Centralized versus decentralized systems, peer-to-peer networks, characteristics of blockchain technology, types of blockchains (public, private, and consortium), basic concepts of consensus mechanisms, overview of cryptocurrencies, and structure of a blockchain including blocks, hash, previous hash, and nonce. | **9** |
| **II** | **Cryptographic Foundations** - Fundamentals of cryptography used in blockchain including hash functions and their properties, digital signatures, public key cryptography, basics of elliptic curve cryptography (concept level), Merkle trees and Merkle proofs, Proof of Work concept, and basic transaction verification process in blockchain systems. | **9** |

| | | |
|---|---|---|
| **III** | **Bitcoin Technology -** Introduction to Bitcoin and its architecture, components of the Bitcoin network, mining process, Proof of Work mechanism, block creation and validation, Bitcoin transactions, UTXO model, wallets and keys, advantages and limitations of Bitcoin, and overview of alternative consensus mechanisms such as Proof of Stake (concept level) | **9** |
| **IV** | **Ethereum and Smart Contract Development -** Introduction to Ethereum architecture, Ethereum Virtual Machine (EVM), and account-based model. Concepts of smart contracts and basics of Solidity programming including data types, functions, modifiers, events, mappings, and structs. Writing, compiling, testing, and deploying smart contracts using Remix IDE. Gas mechanism and transaction fees. Interaction with deployed contracts and overview of decentralized applications (DApps). Comparison between Bitcoin scripting and Ethereum smart contracts. | **10** |
| **V** | **Blockchain Applications and Security -** Applications of blockchain in healthcare, banking, supply chain management, digital identity, and voting systems. Basic security issues in blockchain including 51% attack, double spending, and Sybil attack. Introduction to Layer 2 scaling solutions, basic concept of blockchain interoperability, regulatory considerations, and overview of Web3 concepts. | **10** |
| | Total Hours | **45** |

**vii. ASSESSMENT PATTERN**

Continuous Assessment: End Semester Examination – 40: 60

| Continuous Assessment | | |
|---|---|---|
| Attendance | : | 5 marks |
| Assignments | : | 15 marks |
| Assessment through Tests | : | 20 marks |
| **Total Continuous Assessment** | : | **40 marks** |
| **End Semester Examination** | : | **60 marks** |
| **TOTAL** | : | **100 marks** |

**viii. CONTINUOUS ASSESSMENT TEST**

- No. of Tests: 02
- Maximum Marks: 30
- Test Duration: 1 ½ hours
- Topics: 2 ½ modules

**ix. END SEMESTER EXAMINATION**

- Maximum Marks: 60
- Exam Duration: 3 hours

| Course Code | Course Name | Category | L | T | P | J | Credit | Year of Introduction |
|---|---|---|---|---|---|---|---|---|
| 23CSL46D | VIBE CODING | PEC | 2 | 1 | 0 | 0 | 3 | 2023 |

## i. COURSE OVERVIEW

This course focuses on intuitive, creative, and AI-assisted programming practices that emphasize problem-solving, rapid prototyping, code readability, collaboration, and developer experience. The course blends foundational coding concepts with modern tools, including AI pair programming, low-code platforms, and human-centric software design.

## ii. COURSE OUTCOMES

After the completion of the course, the student will be able to:

| Course Outcomes | Description | Level |
|---|---|---|
| CO1 | Explain the concept, philosophy, and significance of Vibe Coding. | Understand |
| CO2 | Interpret clean coding principles and human-centric programming practices. | Understand |
| CO3 | Apply AI-assisted and low-code tools for basic program development. | Apply |
| CO4 | Apply creative and rapid application development techniques to solve simple problems. | Apply |
| CO5 | Apply collaborative coding practices and basic testing techniques in project development. | Apply |

## iii. SYLLABUS

**Introduction to Vibe Coding**: Traditional coding vs Vibe Coding**,** Developer mindset and creative problem-solving**,** Programming paradigms overview, Code readability, writability, and maintainability**,** Ethical and responsible coding practices**.**

**Human-Centric Programming & Developer Experience (DX):** Clean code principles and coding standards**,** Code aesthetics, Self-documenting code and comments, Debugging strategies and intuitive error handling, Version control fundamentals using Git, Collaborative workflows using GitHub**,** Inclusive and accessible software design.

**AI-Assisted and Low-Code Programming:** Fundamentals of prompt engineering, Pair programming with AI, Low-code / no-code platforms overview, Integrating low-code tools with traditional development, Evaluating AI-generated code for correctness and security.

**Creative Coding and Rapid Application Development:** Concepts and experimentation, Building small applications, Rapid prototyping workflows, UI/UX basics for programmers**,** Event-driven and interactive programming**,** Performance vs creativity trade-offs**,** Case studies of innovative software solutions.

**From Prototype to Product:** MVP thinking and iterative development, Feature prioritization and scope management, Managing technical debt in creative projects, Importance of testing in modern development, Secure coding fundamentals, Performance Optimization & Maintainability.

**Capstone Project.**

**iv (a) TEXTBOOKS**

1. Martin, Robert C. Clean Code: A Handbook of Agile Software Craftsmanship. *Pearson Education*, 2008.
2. Noring, C., et al. AI-Assisted Programming for Web and Machine Learning. *Packt Publishing*, 2024.
3. Taulli, Tom. AI-Assisted Programming: Better Planning, Coding, Testing, and Deployment. 1st ed., *O'Reilly Media*, 2024.

**(b) REFERENCES**

1. Jeremy Morgan, Coding with AI. Manning, 2025.
2. Vootukuri, Naga Santhosh Reddy. Vibe Coding with GitHub Copilot: Enhancing Productivity by Leveraging GitHub Copilot Inside Visual Studio. Apress, 2025.
3. Kernighan, Brian W., and Rob Pike. The Practice of Programming. Addison-Wesley, 1999.

**v. COURSE PLAN**

| Module | Contents | Hours |
|---|---|---|
| **I** | **Introduction to Vibe Coding**: concept, philosophy, and evolution, Traditional coding vs Vibe Coding, Creativity, intuition, empathy, and flow in programming, Developer mindset and creative problem-solving**,** Programming paradigms overview (procedural, object-oriented, functional and Event-driven), Selecting paradigms based on problem context, Hybrid and multi-paradigm programming, Code readability, writability, and maintainability, Industry-standard coding styles and conventions, Ethical and responsible coding practices, Sustainable and socially responsible software. | **9** |
| **II** | **Human-Centric Programming & Developer Experience (DX):** Understanding user needs and developer needs, Defining problems from a human perspective, Clean code principles and coding standards**,** Code | **9** |

| | | |
|---|---|---|
| | aesthetics: naming, structure, and documentation**,** Self-documenting code and comments, Documentation using Markdown and README standards, Debugging strategies and intuitive error handling, Version control fundamentals using Git, Collaborative workflows using GitHub, Branching strategies, pull requests, and code reviews, Open-source contribution models and licensing basics, Inclusive and accessible software design. | |
| **III** | **AI-Assisted and Low-Code Programming:** Introduction to AI-assisted coding tools, Strengths and limitations of AI coding tools, Fundamentals of prompt engineering, Context, constraints, roles, and examples in prompts, Iterative prompting and refinement techniques, Debugging prompts and improving outputs, Pair programming with AI, Human–AI collaboration models (use cases and limitations), Low-code / no-code platforms overview, Building simple applications and workflows using Bubble and Glide, Designing web interfaces and frontend experiences using Webflow, Creating automated workflows and integrations using Zapier, Integrating low-code tools with traditional development, Evaluating AI-generated code for correctness and security. | **9** |
| **IV** | **Creative Coding and Rapid Application Development:** Creative coding concepts and experimentation, Creative problem-solving strategies, Building small applications (scripts, tools, web apps), Rapid prototyping workflows, Introduction to frontend–backend interaction, UI/UX basics for programmers, Visual hierarchy, layouts, and interaction design, Wireframing and interface ideation using Figma, Familiarization of Visual Studio Code, Event-driven and interactive programming**,** Performance vs creativity trade-offs**,** Case studies of innovative software solutions. | **9** |
| **V** | **From Prototype to Product:** MVP thinking and iterative development, Feature prioritization and scope management, Managing technical debt in creative projects, Agile development basics and team workflows, Vibe Coding in startups vs enterprise environments, Importance of testing in modern development, Unit testing concepts and basic automation, API testing and backend validation using Postman, Secure coding fundamentals, Common vulnerabilities and safe coding practices, Privacy-aware and responsible application design, Ethical considerations in AI-assisted systems, Performance Optimization & Maintainability, Code maintainability and long-term scalability, Environment consistency and deployment basics using Docker.<br><br>**Capstone Project**: Team-based micro-project using Vibe Coding principles. | **9** |
| | Total Hours | **45** |

## vi. ASSESSMENT PATTERN

Continuous Assessment: End Semester Examination – 40: 60

| Continuous Assessment | | |
|---|---|---|
| Attendance | : | 5 marks |
| Assignments | : | 15 marks |
| Assessment through Tests | : | 20 marks |
| **Total Continuous Assessment** | : | **40 marks** |
| **End Semester Examination** | : | **60 marks** |
| **TOTAL** | : | **100 marks** |

## vii. CONTINUOUS ASSESSMENT TEST

- No. of Tests: 02
- Maximum Marks: 30
- Test Duration: 1 ½ hours
- Topics: 2 ½ modules

## viii. END SEMESTER EXAMINATION

- Maximum Marks: 60
- Exam Duration: 3 hours

**HONOURS**

| Course Code | Course Name | Category | L | T | P | J | Credit | Year of Introduction |
|---|---|---|---|---|---|---|---|---|
| 23CSJ4H(B, D,F) | MINI PROJECT | VAC | 0 | 0 | 6 | 0 | 3 | 2023 |

## i) COURSE OVERVIEW:

The main objective of this course is to apply fundamental concepts learned in the respective Honour streams of Engineering through practical implementation. The course provides students with exposure to the development of application-oriented software, hardware solutions, or software simulations in their chosen field. Working in small teams under faculty supervision, students identify a relevant problem, perform literature review, plan and execute the project, and present the outcomes through a working model or prototype. The mini project enhances problem-solving ability, teamwork, project planning, and technical communication skills, and serves as a foundation for future research or major project work.

## ii) COURSE OUTCOMES:

After the completion of the course the student will be able to

| Course Outcomes | Description | Level |
|---|---|---|
| CO1 | Identify engineering problems that are socially relevant, technically feasible, and economically viable. | Apply |
| CO2 | Make use of relevant literature and existing engineering principles to explore and analyze potential solutions. | Apply |
| CO3 | Develop a suitable design or methodology using modern tools while adhering to professional ethics. | Apply |
| CO4 | Evaluate the performance or feasibility of the proposed solution using theoretical or experimental validation. | Evaluate |
| CO5 | Apply effective communication techniques to prepare technical reports, presentations, and demonstrations. | Apply |
| CO6 | Plan and execute project activities and coordinate effectively within a team to meet objectives and deadlines. | Apply |

**iii) COURSE PLAN / GUIDELINES:**

In this course, each group consisting of a maximum of 4 members is expected to design and develop a moderately complex software/hardware system relevant to their domain with practical applications. The basic concept of product design may be taken into consideration. Students should identify a topic of interest in consultation with the Project guide assigned and Project coordinator in charge of the course. They should review the literature from latest journals, relevant conferences and gather information pertaining to the chosen topic. Later the group should state the objectives and develop a methodology to achieve the objectives and then carry out the design/fabrication or develop codes/programs to achieve the objectives. Finally the team should demonstrate the novelty of the project through the working results and outputs. The progress of the mini project is evaluated based on a minimum of two reviews. The review committee may be constituted by the Head of the Department/Project Coordinator in charge. A project report is required at the end of the semester which will be evaluated and approved by the department. Innovative design concepts, reliability considerations, aesthetics/ergonomic aspects taken care of in the project shall be given due considerations. A zeroth review may be conducted to finalize the topic and plan in the beginning. Two continuous reviews will assess progress and implementation.

**iv) EVALUATION SCHEME**

The Mini Project shall be evaluated through a comprehensive assessment framework designed to monitor continuous progress, technical competence, and quality of documentation. The evaluation carries a total of 100 marks, with a minimum pass requirement of 50 marks, and is conducted through multiple stages involving the project guide and an evaluation committee. Continuous assessment includes project progress evaluation by the guide, two interim evaluations by the Evaluation Committee to assess problem formulation, design, and implementation progress, and a final evaluation to judge the level of completion, functionality, demonstration, technical understanding, and viva-voce performance. In addition, the quality of the project report is assessed for technical depth, clarity, organization, and adherence to prescribed standards. The Evaluation Committee comprises the Head of the Department or a senior faculty member, the Mini Project Coordinator, and the Project Supervisor, ensuring a fair and holistic evaluation of the mini project.

**v) EVALUATION PATTERN AND MARK DISTRIBUTION**

| | | |
|---|---|---|
| Project progress evaluation by Guide | : | 20 marks |
| Interim Evaluation – I | : | 15 marks |
| Interim Evaluation – II | : | 15 marks |
| Final Evaluation by Evaluation Committee: | | 30 marks |
| Quality of Project Report | : | 20 marks |
| **Total** | : | **100 marks** |