# Academic Program - UG

## Syllabus 2023-24

# V & VI Semester B.E.

# Computer Science and Engineering

SHRI DHARMASTHALA MANJUNATHESHWARA COLLEGE OF ENGINEERING & TECHNOLOGY,
DHARWAD – 580 002
**(An Autonomous Institution approved by AICTE & Affiliated to VTU, Belagavi)**

**Ph**: **0836-2447465Fax: 0836-2464638 Web:** www.sdmcet.ac.in

## SDM College of Engineering & Technology, Dharwad

It is certified that the scheme and syllabus for V & VI Semesters of UG program in Computer Science and Engineering is recommended by Board of Studies of Computer Science and Engineering Department and approved by the Academic Council, SDM College of Engineering &Technology, Dharwad. This scheme and syllabus will be in force from the academic year 2023-24 till further revision.

Principal

Chairman BoS & HoD

## Department of Computer Science and Engineering

## College Vision and Mission

### Vision

To develop competent professions with human values

### Mission

- To have contextually relevant Curricula.
- To promote effective Teaching Learning Practices supported by Modern Educational Tools and Techniques.
- To enhance Research Culture.
- To involve the Industrial Expertise for connecting Classroom contents to real-life situations.
- To inculcate Ethics and soft-skills leading to overall personality development.

**QUALITY POLICY:**

In its quest to be a role model institution, committed to meet or exceed the utmost interest of all the stake holders.

**Core Values:**

- Competency
- Commitment
- Equity
- Team work and
- Trust

## DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

### VISION

To develop competent professionals in the field of Computer Science and Engineering with human values.

### MISSION

1. To have contextually relevant curricula in line with industry trends and body of knowledge stated by IEEE/ACM.
2. To promote OBE based effective Teaching Learning Practices supported by modern educational tools and techniques.
3. To enhance research.
4. To involve the industrial expertise for connecting classroom contents to real-life situations.
5. To inculcate ethics and soft-skills leading to overall personality development.

### Program educational Objectives (PEO)

**I.** To prepare students for successful careers in Industry, Research and Institutions of higher learning

**II.** To encourage students to work in teams to address industrial and socially relevant problems / projects.

**III.** To provide students with a sound mathematical, scientific and engineering fundamentals necessary to formulate, analyse and solve engineering problems.

**IV.** To promote student awareness and commitment to lifelong learning and professional ethics during the course of professional practice.

## PROGRAMME OUTCOMES (POs) and Programme Specific Outcomes (PSOs)

**Program Outcomes (POs):**

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 modeling 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.

**Program Specific outcomes (PSOs):**

13. **System Inception and Elaboration:** Conceptualize the software and/or hardware systems, system components and process/procedures through requirement analysis, modeling /design of the system using various architectural / design patterns, standard notations, procedures and algorithms.
14. **System Construction:** Implement the systems, procedures and processes using the state-of-the-art technologies, standards, tools and programming paradigms.
15. **System Testing and Deployment:** Verify and validate the systems, procedures and processes using various testing and verification techniques and tools.
16. **System Quality and Maintenance:** Manage the quality through various product development strategies under revision, transition and operation through maintainability, flexibility, testability, portability, reusability, interoperability, correctness, reliability, efficiency, integrity and usability to adapt the system to the changing structure and behavior of the systems /environments

## SDM COLLEGE OF ENGINEERING AND TECHNOLOGY, DHARWAD
## Department of Computer Science and Engineering
## V Semester
## Scheme of Teaching and Examinations 2023 – 24

| Course Code | Course Category | Course Title | Teaching | | Examination | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | | | | CIE | Theory (SEE) | | Practical (SEE) | |
| | | | L-T-P (Hrs/Week) | Credits | Max. Marks | Max. Marks | Duration in Hrs. | Max. Marks | Duration In Hrs. |
| 21UHUC520 | HU | Management, Entrepreneurship and IPR | 3-0-0 | 3 | 50 | 100 | 3 | - | - |
| 21UCSC500 | PC | Database Management Systems | 3-0-0 | 3 | 50 | 100 | 3 | - | - |
| 21UCSC501 | PC | Object Oriented Modeling and Design Patterns | 3-0-0 | 3 | 50 | 100 | 3 | - | - |
| 21UCSC502 | PC | Web Technology | 3-0-0 | 3 | 50 | 100 | 3 | - | - |
| 21UCSE5XX | PE | Program Elective – 1 | 3-0-0 | 3 | 50 | 100 | 3 | - | - |
| 21UCSL503 | PC | Database Management Systems Lab | 0-0-2 | 1 | 50 | - | - | 50 | 3 |
| 21UCSL504 | PC | Web Technology Lab | 0-0-2 | 1 | 50 | - | - | 50 | 3 |
| 21UAEE520 | PC | Data Analysis and Visualization Tools | 2-0-0 | 2 | 50 | 50 | 2 | - | - |
| 21UCSL505 | PC | Minor Project – 1 | 0-0-2 | 1 | 50 | - | - | - | - |
| 21UCSL506 | PC | Internship – I | Minimum 2 Weeks | 1 | 50 | - | - | - | - |
| | | **Total** | **17-0-6** | **21** | **500** | **550** | | **100** | |

**Program Elective – 1:**

| Course Code | Course Category | Course Title | Teaching | | Examination | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | | L-T-P (Hrs/Week) | Credits | CIE | Theory (SEE) | | Practical (SEE) | |
| | | | | | Max. Marks | Max. Marks | Duration in Hrs. | Max. Marks | Duration In Hrs. |
| 21UCSE507 | PE | Internet of Things | 3-0-0 | 3 | 50 | 100 | 3 | - | - |
| 21UCSE508 | PE | Unix Systems Programming | 3-0-0 | 3 | 50 | 100 | 3 | - | - |
| 21UCSE509 | PE | Mobile Applications Development | 3-0-0 | 3 | 50 | 100 | 3 | - | - |
| 21UCSE510 | PE | Operations Research | 3-0-0 | 3 | 50 | 100 | 3 | - | - |
| 21UCSE511 | PE | Computer Graphics | 3-0-0 | 3 | 50 | 100 | 3 | - | - |
| 21UCSE512 | PE | Advanced Graph Theory | 3-0-0 | 3 | 50 | 100 | 3 | - | - |
| 21UCSE513 | PE | System Simulation and Modeling | 3-0-0 | 3 | 50 | 100 | 3 | - | - |

## SDM COLLEGE OF ENGINEERING AND TECHNOLOGY, DHARWAD
## Department of Computer Science and Engineering
## VI Semester
## Scheme of Teaching and Examinations 2023 – 24

| Course Code | Course Category | Course Title | Teaching | | Examination | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | | | | CIE | Theory (SEE) | | Practical (SEE) | |
| | | | L-T-P (Hrs/Week) | Credits | Max. Marks | Max. Marks | Duration in Hrs. | Max. Marks | Duration In Hrs. |
| 21UCSC600 | PC | Computer Networks | 3-0-0 | 3 | 50 | 100 | 3 | - | - |
| 21UCSC601 | PC | Advanced Computer Architecture | 3-0-0 | 3 | 50 | 100 | 3 | - | - |
| 21UCSC602 | PC | Automata Theory and Compiler Design | 3-0-0 | 3 | 50 | 100 | 3 | - | - |
| 21UCSE6XX | PE | Program Elective – 2 | 3-0-0 | 3 | 50 | 100 | 3 | - | - |
| 21UCSE6XX | PE | Program Elective – 3 | 3-0-0 | 3 | 50 | 100 | 3 | - | - |
| 21UCSO6XX | OE | Open Elective – 1 | 3-0-0 | 3 | 50 | 100 | 3 | - | - |
| 21UCSL603 | PC | Computer Networks Lab | 0-0-2 | 1 | 50 | - | - | 50 | 3 |
| 21UCSL604 | PC | Automata Theory and Compiler Design Lab | 0-0-2 | 1 | 50 | - | - | 50 | 3 |
| 21UCSL605 | PC | Minor Project – 2 | 0-0-3 | 1 | 50 | - | - | 50 | 3 |
| 21UHUL606 | HU | Soft Skills and Aptitude | 0-0-2 | 1 | 50 | - | - | - | - |
| | | **Total** | **18-0-9** | **22** | **500** | **600** | | **150** | |

**Program Elective – 2:**

| Course Code | Course Category | Course Title | Teaching | | Examination | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | | L-T-P (Hrs/Week) | Credits | CIE | Theory (SEE) | | Practical (SEE) | |
| | | | | | Max. Marks | Max. Marks | Duration in Hrs. | Max. Marks | Duration In Hrs. |
| 21UCSE607 | PE | Advanced Data Structures and Algorithms | 3-0-0 | 3 | 50 | 100 | 3 | - | - |
| 21UCSE608 | PE | Data Science | 3-0-0 | 3 | 50 | 100 | 3 | - | - |
| 21UCSE609 | PE | Natural Language Processing | 3-0-0 | 3 | 50 | 100 | 3 | - | - |

**Program Elective – 3:**

| Course Code | Course Category | Course Title | Teaching | | Examination | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | | L-T-P (Hrs/Week) | Credits | CIE | Theory (SEE) | | Practical (SEE) | |
| | | | | | Max. Marks | Max. Marks | Duration in Hrs. | Max. Marks | Duration In Hrs. |
| 21UCSE610 | PE | Soft and Evolutionary Computing | 3-0-0 | 3 | 50 | 100 | 3 | - | - |
| 21UCSE611 | PE | Software Testing | 3-0-0 | 3 | 50 | 100 | 3 | - | - |
| 21UCSE612 | PE | Pattern Recognition | 3-0-0 | 3 | 50 | 100 | 3 | - | - |

**Open Elective – 1: (All Branches)**

| Course Code | Course Category | Course Title | Teaching | | Examination | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | | L-T-P (Hrs/Week) | Credits | CIE | Theory (SEE) | | Practical (SEE) | |
| | | | | | Max. Marks | Max. Marks | Duration in Hrs. | Max. Marks | Duration In Hrs. |
| 21UCSO613 | OE | Cloud Computing | 3-0-0 | 3 | 50 | 100 | 3 | - | - |
| 21UCSO614 | OE | Ontology and Semantic Web | 3-0-0 | 3 | 50 | 100 | 3 | - | - |
| 21UCSO615 | OE | Introduction to Software Engineering | 3-0-0 | 3 | 50 | 100 | 3 | - | - |
| 21UCSO616 | OE | Introduction to Data Structures | 3-0-0 | 3 | 50 | 100 | 3 | - | - |

## V Semester

**21UHUC520 Management, Enterprenuership, and IPR (3-0-0) 3**

**Contact Hours: 39**

**Course Learning Objectives (CLOs):** This course focuses on the following learning perspectives:

- The evolution of managerial aspects like planning, organizing, decision making, and staffing.
- The scope of entrepreneurship in small, medium, and large scale industries.
- The issues related to intellectual property rights.

.

**Course Outcomes (COs):**

| **Description of the Course Outcome:** At the end of the course the student will be able to: | | **Mapping to POs(1-12)/ PSOs (13-16)** | | |
|---|---|---|---|---|
| | | **Substantial Level (3)** | **Moderate Level (2)** | **Slight Level (1)** |
| **CO-1** | **Explain** the characteristics of management in an organization. | 11 | - | 9 |
| **CO-2** | **Describe** the role of staffing and motivation in management. | - | 1 | - |
| **CO-3** | **Explain** the role of entrepreneur in building firms. | - | 6 | - |
| **CO-4** | **Describe** the importance of institutional support for an enterprise. | 1 | - | 9 |
| **CO-5** | **Explain** the contemporary issues related to intellectual propert rights. | - | 8 | - |

| **POs/PSOs** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** | **13** | **14** | **15** | **16** |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Mapping Level** | 2.0 | - | - | - | - | 2.0 | - | 2.0 | 1.0 | - | 2.0 | - | - | - | - | - |

**Pre-requisites:** Knowledge of humanities course.

**Contents:**

### Unit-I

**Engineering and Management:** Historical Development of Engineering and Management, Management as synthesis.

**Planning, Forecasting and Decision Making:** Nature of Planning, foundation of planning, some planning concepts, forecasting, nature of decision making, management science, tools for decision making. **7 Hrs**

## Unit-II

**Organizing and staffing:** Nature of organizing, traditional organizational theory, technology and modern organization structures, staffing technical organization, authority and power; delegation, meeting and committees.
**Motivation:** Motivation, leadership, motivating and leading technical professionals.
**Controlling:** Process of control, financial and non-financial controls. **8 Hrs**

## Unit-III

**Foundations of Entrepreneurship:** Meaning of entrepreneur, functions of entrepreneur, types of entrepreneur, concept of entrepreneurship, role of entrepreneurs in economic development & barriers of entrepreneurship.
**Small Scale Industry:** Definition, characteristics of SSI, role of SSI in economic development, advantages of SSI, steps to start an SSI, impact of liberalization, privatization, and globalization on SSI, definition of ancillary and tiny industry. **8 Hrs**

## Unit-IV

**Government and Institutional Support:** Nature of support from government, objectives and functions of SSI, SIDBI, DIC, single window agency, KIADB, KSSIDC, KSFC.
**Preparation of Project:** Meaning of project identification, project report, contents and formulation, identification of business opportunities, feasibility studies, types and purpose. **8 Hrs**

## Unit-V

**Intellectual Property Rights:** Meaning and forms of intellectual property rights, competing rationale for protection, international conventions and security.
**Copyright:** Meaning of copyright, content of copy right, ownership and rights, period of copyright, assignment and relinquishment of copyright, license, infringement of copy right, fair use, offenses and penalties.
**Patents:** Concept of patent, patentable inventions, procedure for obtaining patent, rights and obligations of patent holders, infringements and remedies, offenses and penalties.
**Industrial Designs:** Definition of design, procedure for registration, rights conferred by registration, infringements, Trademark and related issues. **8 Hrs**

**Reference Books:**


**1)** N.V.R. Naidu and T. Krishna Rao, “Management and Enterpreneurship”, International Publishing House Pvt Ltd, 2008
**2)** Morse and Bobcock, “Managing Engineering and Technology”, 5/E, PHI. India 2013.
**3)** Thomas.W.Zimmerer, “Essentials of Enterpreneurship”, 2/E, PHI. India 2012.
**4)** N.K.Acharya, “Intellectual Property Rights”, 4/E, Asia Law House, 2012.

| 21UCSC500 | Database Management Systems | (3-0-0) 3 |
|---|---|---|

**Contact Hours: 39**

**Course Learning Objectives (CLOs):** This course focuses on the following learning perspectives:

- Data models and relational theories
- Database design, programming using SQL/PL-SQL, database architecture and transaction concepts.

.

**Course Outcomes (COs):**

| **Description of the Course Outcome:** At the end of the course the student will be able to: | | **Mapping to POs(1-12)/ PSOs (13-16)** | | |
|---|---|---|---|---|
| | | **Substantial Level (3)** | **Moderate Level (2)** | **Slight Level (1)** |
| **CO-1** | **Compare** the traditional file system and Data Base approach | - | 2,13 | - |
| **CO-2** | **Identify** entities, attributes, their relationships and **prepare** ER model for the given application scenario. | 2,3,13 | - | 15 |
| **CO-3** | **Write** the queries using relational algebra for the given data manipulation requirement of an RDBMS. | 2,3,13,14 | - | 15 |
| **CO-4** | **Write** SQL queries using all the standard clauses, correlated queries, aggregate and date related functions for the given application scenario. | 2,3,13,14 | - | 15 |
| **CO-5** | **Write** triggers, stored procedures and functions for the given application scenario. | 2,3,13,14 | - | 15 |
| **CO-6** | **Design** database in appropriate normal form for a given application scenario. | 2,3,13,14 | - | 15 |
| **CO-7** | **Explain** the strategies to deal with the issues related to transaction management and to ensure ACID properties. | | 1,13 | |

| POs/PSOs | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Mapping Level | 2.0 | 2.8 | 3.0 | - | - | - | - | - | - | - | - | - | 2.7 | 3.0 | 1.0 | - |

**Pre-requisites:** Knowledge of

- Programming languages
- Set Theory
- File Systems
- Abstract application development process.

**Contents:**

**Unit-I**

**Introduction:** Introduction; An example; Characteristics of Database approach; Actors on the screen; Workers behind the scene; Advantages of using DBMS approach; A brief history of database applications; when not to use a DBMS. Data models, schemas and instances; Three-schema architecture and data independence; Database languages and interfaces; The database system environment; Centralized and client-server architectures;

**Entity-Relationship Model:** Using High-Level Conceptual Data Models for Database Design; An Example Database Application; Entity Types, Entity Sets, Attributes and Keys; Relationship types, Relationship Sets, Roles and Structural Constraints; Weak Entity Types; Refining the ER Design; ER Diagrams, Naming Conventions and Design Issues; Relationship types of degree higher than two. **8 Hrs**

**Unit-II**

**Relational Model and Relational Algebra:** Relational Model Concepts; Relational Model Constraints and Relational Database Schemas; Update Operations, Transactions and dealing with constraint violations; Unary Relational Operations: SELECT and PROJECT; Relational Algebra Operations from Set Theory; Binary Relational Operations: JOIN and DIVISION; Examples of Queries in Relational Algebra; Relational Database Design ER-to-Relational Mapping. **8 Hrs**

**Unit-III**

**SQL:** Data Definition and Data Types; DDL statements like creation and specification od table, DCL statements for Schema change, alter, delete etc; DML statements like Insert, Delete and Update statements in SQL etc and more complex statements for Basic queries Nested sub queries, Correlated sub queries. PL/SQL; Specifying constraints as Assertion and Trigger; Views (Virtual Tables) in SQL; Additional features of SQL; Database programming issues and techniques; Embedded SQL, Dynamic SQL; Database stored procedures and SQL. **8 Hrs**

**Unit-IV**

**Database Design – 1:** Informal Design Guidelines for Relation Schemas; Functional Dependencies; Inference rule, Equivalence of sets, Minimal set cover. Normal Forms Based on Primary Keys; General Definitions of Second and Third Normal Forms; Boyce-Codd Normal Form. **8 Hrs**

**Unit-V**

**Database Design – 2:** Properties of Relational Decompositions; Dependency preservation, Lossless (non additive) join properties, Problem with null values and dangling tuples.

**Transaction Management:** The ACID Properties; Transactions and Schedules; Concurrent Execution of Transactions; Lock- Based Concurrency Control; Performance of locking; Transaction support in SQL; Introduction to crash recovery; 2PL, Serializability and Recoverability; Lock Management; Introduction to ARIES; The log; Other recovery-related structures; The write-ahead log protocol; Check pointing.

**7 Hrs**

**Reference Books:**


**1)** Elmasri & Navathe, "Fundamentals of Database Systems", 6/E, Addison-Wesley, 2012.

**2)** Raghu Ramakrishnan & Johannes Gehrke, "Database Management Systems", 3/E, McGraw-Hill, 2003.

**3)** Silberschatz, Korth and Sudharshan, "Data base System Concepts", 6/E, Mc-Graw Hill, 2010.

**4)** C.J. Date, A. Kannan & S. Swamynatham, "An Introduction to Database Systems", 8/E, Pearson Education, 2006.


## 21UCSC501 Object Oriented Modeling and Design Patterns (3-0-0) 3

**Contact Hours: 39**

**Course Learning Objectives (CLOs):** This is a 3 credit, 39 contact hours course at undergraduate level focusing on the process of object oriented system modeling, design, patterns and tools used in the industry to enable them to construct software system using various standards and techniques.

**Course Outcomes (COs):**

| **Description of the Course Outcome:** At the end of the course the student will be able to: | | **Mapping to POs(1-12)/ PSOs (13-16)** | | |
|---|---|---|---|---|
| | | **Substantial Level (3)** | **Moderate Level (2)** | **Slight Level (1)** |
| **CO-1** | **Apply** fundamental Object Oriented concepts in solving problems. | 13,14 | 1,2,3,15 | - |
| **CO-2** | **Analyze** problem scenario and **identify** classes/ Objects, their properties and associations. | 13 | 1,2,3,5,15 | - |
| **CO-3** | **Analyze** problem scenario and **model** the system using UML diagrams. | 13 | 1,2,3,5,15 | - |
| **CO-4** | **Evaluate** the quality of Object Oriented system in-terms of | - | 16 | 15 |

| | | | | |
|---|---|---|---|---|
| | Cohesion, coupling, sufficiency, completeness and primitiveness. | | | |
| **CO-5** | **Implement** Object Oriented model in any Object Oriented language. | 13,14 | 1,2,3 | 15 |
| **CO-6** | **Identify** and **apply** the appropriate patterns in solving problems. | - | 13,16 | 1 |
| **CO-7** | **Propose** the appropriate strategies to incorporate standard quality parameters in the design of a system. | - | 13,16 | 1 |

| POs/PSOs | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Mapping Level | 1.7 | 2.0 | 2.0 | - | 2.0 | - | - | - | - | - | - | - | 2.7 | 3.0 | 1.6 | 2.0 |

**Pre-requisites:** Knowledge of

- Object Oriented Programming Language (any)
- Software Engineering

**Contents:**

**Unit-I**

**Review:** Object Oriented Concepts and principles.

**Introduction, modeling concepts, class modeling:** Object Orientation, developments themes; Evidence for usefulness of developments; modeling history. Modeling as Design Technique: Modeling; abstraction; The three models. Class Modeling: Object and class concepts; Link and associations concepts; Generalization and inheritance; A sample class model; Navigation of class models; Practical tips.

**7 Hrs**

**Unit-II**

**Advanced class modeling, state modeling:** Advanced object and class concepts; Association ends; N-ary associations; Aggregation; Abstract classes; Multiple inheritance; Metadata; Reification; Constraints; Derived data; Packages; Practical tips. State Modeling: Events, States, Transitions and Conditions; State diagrams; State diagram behavior; Practical tips.

**Advanced state modeling, interaction modeling:** Advance state modeling: nested state diagrams; nested states; signal generalization; concurrency; a sample state model; relation of class and state models; practical tips. Interaction modeling: use case models; sequence models; activity models. Use case relationships; procedural sequence models; special constructs for activity models.

**Evaluation**: OO system quality in terms of Cohesion, coupling, sufficiency, completeness and primitiveness.

**Implementation** : OO design in appropriate language **8 Hrs**

**Unit-III**

**Patterns–Part 1:** Introduction; layers, pipes and filters, blackboard.Distributed systems: broker; interactive systems: mvc, presentation-abstraction-control. **8 Hrs**

**Unit-IV**

**Patterns—Part2:** Adaptable systems: microkernel; reflection. Structural decomposition**:** whole - part; organiz**a**tion of work: master - slave; access control: proxy. Others: Command Processor, View Handler, Forward Receiver, Client-Dispatcher-Server and publish Subscriber. **8 Hrs**

**Unit-V**

**Quality:** Functionality and architecture; architecture and quality attributes; system quality attributes; Quality attribute scenarios in practice; Other system quality attributes; Business qualities; Architecture qualities. Achieving Quality: Introducing tactics; Availability tactics; Modifiability tactics; Performance tactics; Security tactics; Testability tactics; Usability tactics; Relationship of tactics to architectural patterns; Styles. **8 Hrs**

**Reference Books:**


**1)** Michael Blaha & James Rumbaugh, "Object-Oriented Modeling and Design with UML", 2/E, Pearson Education, 2007.
**2)** Frank Buschmann, RegineMeunier, Hans Rohnert, Peter Sommerlad & Michael Stal, "Pattern-Oriented Software Architecture, A System of Patterns", Volume 1, John Wiley and Sons, 2006.
**3)** Len Bass, Paul Clements & Rick Kazman, "Software Architecture in Practice", 2/E, Pearson Education, 2003.
**4)** Grady Booch et al, "Object-Oriented Analysis and Design with Applications", 3/E, Pearson Education, 2007.
**5)** Ali Bahrami, "Object oriented systems development", McGrawHill, 1999.
**6)** Mary Shaw and David Garlan, "Software Architecture Perspectives on an Emerging Discipline", Prentice-Hall of India, 2007.


**21UCSC502 Web Technology (3-0-0) 3**

**Contact Hours: 39**

**Course Learning Objectives (CLOs):** This course emphasizes on practice-based learning. It enables the students to understand the web application architecture and use the state-of-the-art technologies to provide web-based solutions.

**Course Outcomes (COs):**

| **Description of the Course Outcome:** At the end of the course the student will be able to: | | **Mapping to POs(1-12) / PSOs (13-16)** | | |
|---|---|---|---|---|
| | | **Substantial Level (3)** | **Moderate Level (2)** | **Slight Level (1)** |
| **CO-1** | **Explain** the fundamentals of web technology and **develop** static web pages using XHTML | 13 | 14 | 1,5,9 |

| | | | | |
|---|---|---|---|---|
| **CO-2** | **Explain** the significance of CSS and its associated frameworks in designing web pages and **develop** web pages using CSS and its associated frameworks | 13 | 14 | 1,5,9 |
| **CO-3** | **Design** and **develop** an interactiveweb application using jQuery, XHTML and CSS | 13 | 14 | 1,5,9 |
| **CO-4** | **Design** and **develop** dynamic webapplications using server-sideprogramming and **demonstrate**databaseconnectivity | 13 | 14 | 1,5,9 |
| **CO-5** | **Explain** the use of AJAX technology to interact with the server asynchronously and **demonstrate** the use of AJAX in web applications | 13 | 14 | 1,5 |
| **CO-6** | **Explain** the role of web services in representing the data in the standard formats as per the requirements | - | 5 | 1 |
| **CO-7** | **Explain** the future of World WideWeb and its associated technologies | - | 5 | 1,12 |

| POs/PSOs | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Mapping Level | 1.0 | - | - | - | 1.5 | - | - | - | 1.0 | - | - | 1.0 | 3.0 | 2.0 | - | - |

**Prerequisites:** Knowledge of:

- Any programming language
- Basics of database management systems

**Contents:**

**Unit-I**

**Fundamentals of Web:** Introduction to the Internet, The World Wide Web, Web Browsers, Web Servers, Uniform Resource Locators, Multipurpose Internet Mail Extensions, The Hypertext Transfer Protocol, The Web Programmer's Toolbox
**Introduction to XHTML:**Origins and Evolution of HTML and XHTML, Basic Syntax, Standard XHTML Document Structure, Basic Text Markup, Images, Hypertext Links, Lists, Tables, Forms, Syntactic Differences between HTML and XHTML **8 Hrs**

**Unit-II**

**Cascading Style Sheets:**Introduction, Levels of Style Sheets, Style Specification Formats, Selector Forms, Property Value Forms, Font Properties, List Properties, Color, Alignment of Text, The Box Model, Background Images, The  and  Tags, Conflict Resolution

**Tailwind CSS:** Tailwind basics,typography, the Box, page layout, responsive design **7 Hrs**

**Unit-III**

**jQuery:** Getting Started- what jQuery does, why jQuery works, jQuery powered web-page, Plain JavaScript vs. jQuery; Selecting Elements- The Document Object Model, The $() function, CSS selectors, custom selectors, DOM traversal methods, accessing DOM elements; Handling events- performing tasks on page load, simple events, compound events, journey of an event, the event object, removing an event handler; Manipulating the DOM- manipulating attributes, DOM tree manipulation, copying elements, content getter and setter methods, DOM manipulation methods **8 Hrs**

**Unit-IV**

**Introduction to PHP:** Origins and uses of PHP, Overview of PHP, General Syntactic Characteristics, Primitives, Operations, and Expressions, Output, Control Statements, Arrays, Functions, Pattern Matching, Form Handling, Cookies, Session Tracking

**Database access through the Web:**Relational Databases, Architectures for Database Access, The MySQL Database System, Database Access with PHP and MySQL **9 Hrs**

**Unit-IV**

**Introduction to AJAX:** Overview of Ajax, The Basics of Ajax, Return Document Forms

**Introduction to Web 2.0:** Overview of WWW 2.0, JSON, Web Services - SOAP &WSDL, RESTful

**Future of Web:** Overview of Semantic Web, Applications of Semantic Web, VirtualReality, Web OS **7 Hrs**

**CTA activity:**

As part of CTA activity, the student need to mandatorily work on simple use-case based project in a team (maximum 2 members per team) and demonstrate the working in front of evaluators.

**Reference Books:**

1) Robert W. Sebesta, “Programming the World Wide Web”, 7/E Pearson Education, 2014
2) Noel Rappin, “Modern CSS with Tailwind: flexible styling without the fuss”, 1/E, The Pragmatic Bookshelf, 2021

3) Jonathan Chaffer and Karl Swedberg, “Learning jQuery”, 3/E, Pack Publishing, 2011
4) Karin K. Breitman, Marco Antonio Casanova and Walter Truszkowski, “Semantic Web: Concepts, Technologies and Applications”, Springer International Edition, 2007
5) Nicholas C Zakas et al, “Professional AJAX”, Wrox Publication, 2007

## 21UCSE507 Internet of Things (3-0-0) 3

**Contact Hours: 39**

**Course Learning Objectives (CLOs):** This course provides the basic understanding of IoT technology, communication protocols, sensor networks and its applications. It focuses on setting up IoT ecosystem to implement use cases by applying the key concepts of IoT.

**Course Outcomes (COs):**

| **Description of the Course Outcome:** At the end of the course the student will be able to: | | **Mapping to POs(1-12)/ PSOs (13-16)** | | |
|---|---|---|---|---|
| | | **Substantial Level (3)** | **Moderate Level (2)** | **Slight Level (1)** |
| **CO-1** | **Explain** the working of IoT and its enabling technologies. | - | 5 | 1, 12 |
| **CO-2** | **Explain** the use cases of IoT and the use of sensors & actuators in IoT ecosystem. | - | 5 | 1, 12 |
| **CO-3** | **Compare** and **contrast** IoT & M2M and **explain** the generic design methodology for IoT system. | - | 5, 13 | 1, 12 |
| **CO-4** | **Develop** IoT applications using Arduino by making use of sensors and modules. | 13 | 14 | 1, 5, 9, 15 |
| **CO-5** | **Develop** IoT applications using Raspberry Pi microcontroller by making use of sensors and modules. | 13 | 14 | 1, 5, 9, 15 |

| **POs/PSOs** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** | **13** | **14** | **15** | **16** |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Mapping Level** | 1.0 | - | - | - | 1.6 | - | - | - | 1.0 | - | - | 1.0 | 2.7 | 2.0 | 1.0 | - |

**Pre-requisites:** Knowledge of Programming and Basics of Computer Networking

**Contents:**

**Unit-I**

**Introduction to Internet of Things (IoT):** Introduction, Physical and Logical Design of IoT, IoT Enabling Technologies, IoT levels and Deployment templates **7 Hrs**

**Unit-II**

**Domain Specific IoTs:** Introduction, Home Automation, Cities, Environment, Energy, Retail, Logistics, Agriculture, Industry, Health &Lifestyle

**IoT Sensors and Actuators:** Introduction, IoT sensors, RFID, Video Tracking, IoT Actuators **8 Hrs**

**Unit-III**

**IoT and M2M:** Introduction, M2M, Difference between IoT and M2M, SDN and NFV for IoT

**IoT Platforms Design Methodology:** Introduction, IoT Design Methodology, Case Study on IoT System for Weather Monitoring **7 Hrs**

**Unit-IV**

**Arduino Programming:**Features of Arduino, Arduino UNO, Arduino IDE, Arduino sketch structure, Arduino function libraries, Blinking LED example; Operators, control statements, loops, arrays, string, interrupts, Traffic Control System example

**Integration of Sensors, Actuators and modules with Arduino:** Sensor Types, Sensor Interface with Arduino, Interfacing DHT, LM35, LDR, Accelerometer and HC-SR04 sensors with Arduino; Actuators, Types of Motor Actuators, Servo motor, Servo library on Arduino; HC05 Bluetooth module, Features, Interfacing HC05 with Arduino, Controlling LED using Bluetooth; ESP8266 wifi module, Features, Send LM35 sensor data to cloud using wifi module **9 Hrs**

**Unit-V**

**Raspberry Pi:** Introduction, Specifications, Basic Architecture, Pin configuration, Blinking LED example; Capture image using Raspberry Pi

**Implementation of IoT with Raspberry Pi:** Temperature dependent auto cooling system; interfacing DHT, LM35, LDR, Accelerometer and HC-SR04 sensors with Raspberry Pi; Send LM35 sensor data to cloud using wifi module. **8 Hrs**

**Conduction of Practical Sessions:**

Practical sessions shall include experiments on the following:

1) Use of Arduino board and coding to blink built-in as well as external LEDs
2) Interfacing temperature, humidity, soil moisture, light intensity, accelerometer, ultrasonic and obstacle detection sensors with Arduino and displaying the results
3) Connecting modules like Bluetooth and Wi-Fi to Arduino and sending the sensed data to cloud for storage and analytics.
4) Use of Raspberry Pi as a surveillance system

**Reference Books:**


1) Arshdeep Bahga, Vijay Madisetti ,"Internet of Things – A Hands-on Approach", Universities Press, 2015
2) Ammar Rayes, Samer Salam, "Internet of Things From Hype to Reality – The Road to Digitalization", 2/E, Springer Nature Switzerland AG 2017, 2019


**Additional References:**


3) NPTEL course on "Introduction to internet of things" by Prof. Sudip Misra


**21UCSE508 UNIX Systems Programming (3-0-0) 3**

**Contact Hours: 39**

**Course Learning Objectives (CLOs):** This course facilitates the students to get familiarity with system calls, UNIX kernel structure and use of standards like ANSI and POSIX in programming.

**Course Outcomes (COs):**

| **Description of the Course Outcome:** At the end of the course the student will be able to: | | **Mapping to POs(1-12)/ PSOs (13-16)** | | |
|---|---|---|---|---|
| | | **Substantial Level (3)** | **Moderate Level (2)** | **Slight Level (1)** |
| **CO-1** | **Explain** the ANSI and POSIX standards used in UNIX system programming and **write** programs that conform to these standards. | 14 | 13 | 1,5 |
| **CO-2** | **Explain** the UNIX file types and **demonstrate** the use of UNIX file APIs in programming. | - | 14,13 | 1,5 |
| **CO-3** | **Explain** various process management activities of UNIX and **write** programs that make use of processes and their environment to solve the problems related to process management. | 14 | 13 | 1,5 |
| **CO-4** | **Explain** the use of signals in UNIX system and **write** programs that make use of signals to solve problems related to signal handling. | 14 | 13 | 1,5 |
| **CO-5** | **Explain** the daemons feature in UNIX system and **write** programs to create and use demons to deploy services in UNIX OS. | 14 | 13 | 1,5 |

| | | | | |
|---|---|---|---|---|
| **CO-6** | **Explain** inter process communication mechanisms of UNIX and **write** programs to demonstrate IPCs for client-server interactions. | 13,14 | 1,2,3 | 15 |

| POs/PSOs | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Mapping Level | 1.0 | - | - | - | 1.0 | - | - | - | - | - | - | - | 2.0 | 2.8 | - | - |

**Pre-requisites:** Knowledge of

- UNIX operating system and its commands
- Operating Systems Fundamentals
- Basics of Networking

**Contents:**

**Unit-I**

**ANSI and POSIX Standards:** UNIX and ANSI Standards – ANSI C standard, POSIX standards, POSIX environment, POSIX feature test macros, limits checking at compile time and run time; UNIX and POSIX APIs – POSIX APIs, UNIX and POSIX development environment, API common characteristics

**UNIX Files:** File types, UNIX and POSIX file systems, UNIX and POSIX file attributes, inodes in UNIX system V, UNIX kernel support for files, directory files, hard and symbolic links. **7 Hrs**

**Unit-II**

**UNIX File APIs:** General file APIs, open, read, write, close, fcntl, lseek, link, unlink, stat, fstat, lstat, access, chmod, fchmod, chown, fchown, lchown, utime, file and record locking, directory file APIs, device file APIs, FIFO file APIs, symbolic link file APIs. **7 Hrs**

**Unit-III**

**Environment of a UNIX Process:** Introduction, main function, process termination, command line arguments, environment list, memory layout of a C program, alloca function, environment variables, setjmp and lonjmp functions, getrlimit and setrlimit functions

**Process Control:** Introduction, process identifiers, fork function, vfork function, exit functions, wait and waitpid functions, race conditions, exec functions, changing user IDs and group IDs, system function. **9 Hrs**

**Unit-IV**

**Process Relationships:** Introduction, terminal logins, network logins, process groups, sessions, controlling terminal, job control

**Signals and Daemon Processes:** Signals – UNIX kernel support for signals, signal, signal mask, sigaction, sigsetjmp and siglongjmp APIs, kill, alarm, interval timers;

Daemon Processes – introduction, daemon characteristics, coding rules, error logging, client-server model. **9 Hrs**

**Unit-V**

**Interprocess Communication:** Introduction, pipes, message queues, UNIX APIs for message queues, client-server example for message queue, sockets, socket APIs, client-server example for socket. **7 Hrs**

**Reference Books:**


**1)** Terrence Chan, "UNIX System programming using C++",Prentice Hall India, 2015

**2)** W. Richard Stevens, "Advanced Programming in the UNIX environment", Pearson Education/ PHI, 2005

**3)** Kay A Robbins & Steven Robbins, "Unix Systems Programming: Communication, Concurrency, and Threads", Prentice Hall Publications, 2003.


## 21UCSE509 Mobile Applications Development (3-0-0) 3

**Contact Hours: 39**

**Course Learning Objectives (CLOs):** This course focuses on the following learning perspectives:

- Familarize with mobile apps development aspects.
- Design and develop mobile apps, using Android as development platform, with key focus on user experience design, native data handling and background tasks and modifications.
- Appreciation of naunces such as native hardware play, location awareness, graphics, and multimedia.
- Perform testing, signing, packaging and distribution of mobile apps.

**Course Outcomes (COs):**

| **Description of the Course Outcome:** At the end of the course the student will be able to: | | **Mapping to POs(1-12)/ PSOs (13-16)** | | |
|---|---|---|---|---|
| | | **Substantial Level (3)** | **Moderate Level (2)** | **Slight Level (1)** |
| **CO-1** | **Explain** the android platform architecture. | 5,13 | 7 | 3 |
| **CO-2** | **Acquire** the knowledge of UI components and life cycle of activity with respect to android platform. | 5 | 4 | 13 |
| **CO-3** | **Design** and **develop** apps using native data handling on-devices like file I/O, SQLite, and enterprise data access. | 14 | 3,13 | 7 |
| **CO-4** | **Design** and **develop** spruce apps to demonstrate the use of various components involved in android platform. | 13,14 | 3,13 | 7 |

| | | | | |
|---|---|---|---|---|
| **CO-5** | **Explain** and **apply** use of testing tools to test mobile apps. | 15 | 16 | 5 |
| **CO-6** | **Explain** the concept of versioning and distributing apps to mobile market place. | 14 | 5 | 16 |

| POs/PSOs | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Mapping Level | - | - | 1.6 | 2.0 | 2.3 | - | 1.3 | - | - | - | - | - | 2.2 | 3.0 | 3.0 | 1.5 |

**Pre-requisites:** Knowledge of Java (J2SE) and basic RDBMS.

**Contents:**

**Unit-I**

**Getting started with Mobility:** Mobility landscape, Mobile platforms, Mobile apps development, Overview of Android platform, setting up the mobile app development environment along with an emulator, a case study on Mobile app development. **7 Hrs**

**Unit-II**

**Building blocks of mobile apps:** App user interface designing – mobile UI resources (Layout, UI elements, Draw-able, Menu), Activity – states and life cycle, interaction amongst activities, App functionality beyond user interface – Threads, Async task, Services – states and life cycle, Notifications, Broadcast recievers, Telephony and SMS APIs, Native data handling – on device file I/O, shared preferences, mobile databases such as SQLite, and enterprise data access (via Internet/Intranet). **9 Hrs**

**Unit-III**

**Sprucing up mobile apps:** Graphics and animation – custom views, canvas, animation APIs, multimedia – audio/video playback and record, location awareness, and native hardware access (sensors such as accelorometer and gyroscope) **9 Hrs**

**Unit-IV**

**Testing mobile apps:** Debugging mobile apps, White box testing, Black box testing, and test automation of mobile apps, Junit for Android, Robotium, MonkeyTalk **7 Hrs**

**Unit-V**

**Taking mobile apps to Market:** Versioning, signing and packaging mobile apps, distributing apps on mobile market place. **7 Hrs**

**Practical / Project work:** Students should implement (and learn to use the tools to accomplish this task) the following during Practical hours:

1. Understand the app idea and design user interface/wireframes of mobile app.
2. Set up the mobile app development environment.
3. Develop and debug mobile app components – User Interface, Services, Notifications, Broadcast recievers, data components.
4. Usingemulator to deploy and run mobile apps.
5. Testing mobile app – unit testing, black box testing and test automation.

**Reference Books:**

**1)** Donn Felker, Joshua Dobbs, and Barry Burd, "Android Application Development for Dummies", Wiley Publishing Inc, 2011

**2)** Reto Meier, "Professional Android 2 Application Development", Wrox Publisher, 3/E, 2012.

## 21UCSE510 Operations Research (3-0-0) 3

**Contact Hours: 39**

**Course Learning Objectives (CLOs):** This course focuses on following learning perspectives:

- Apply the fundamental techniques (graphical method, simplex method, dual simplex method) of operations research to solve real world problems.
- Estimate the transportation and assignment costs.
- Design, code, test, and debug programs to solve problems in the domain of operations research.

**Course Outcomes (COs):**

| **Description of the Course Outcome:** At the end of the course the student will be able to: | | **Mapping to POs(1-12)/ PSOs (13-16)** | | |
|---|---|---|---|---|
| | | **Substantial Level (3)** | **Moderate Level (2)** | **Slight Level (1)** |
| **CO-1** | **Formulate** real-world problems in the form of mathematical model and **solve** using graphical method. | - | 2 | 1 |
| **CO-2** | **Formulate** real-world problems in the form of mathematical model and **solve** using simplex and artificial variables techniques. | - | 1,2,3 | - |
| **CO-3** | **Solve** LP problems using dual simplex method and **perform** sensitivity analysis. | - | 1,2,3 | 16 |
| **CO-4** | **Build** Models to **Solve** Transportation and Assignment problems. | - | 1,2 | 5,16 |
| **CO-5** | **Solve** a given problem using game theory techniques. | - | 1,2 | 5 |

| **POs/PSOs** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** | **13** | **14** | **15** | **16** |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Mapping Level** | 1.8 | 2.0 | 2.0 | - | 1.0 | - | - | - | - | - | - | - | - | - | - | 1.0 |

**Pre-requisites:** Knowledge of

- Algebra
- Probability

**Contents:**

**Unit-I**

**Introduction:** Definitions and various phases of operations research, Role of computers in operations research, Formulating a problem as a mathematical model in the form of linear programming (LP) problem – Minization and Maximization Types, Solving LP problems using Graphical method. **7 Hrs**

**Unit-II**

**Simplex Method:** Canonical and Standard forms of LP problem, The simplex method – examples, Artificial variables Techniques (Big M and Two Phase method) - examples. **7 Hrs**

**Unit-III**

**Advanced Topics in Linear Programming:** Duality, Dual Simplex Method, Economic interpretation of duality, Sensitivity analysis, Dynamic Programming. **9 Hrs**

**Unit-IV**

**Transportation Model:** Introduction, Formulation, Methods for Initial Basic Feasible solution, Improving the basic feasible solution using Stepping Stone and MODI methods.

**Assignment Model:** Introduction, Formulation, Comparison with transportation model, Hungarian method. **9 Hrs**

**Unit-V**

**Game Theory:** Introduction, Formulation, Strategies – Pure and Mixed, Methods for solving Game theory problems – Saddle point, Rule of Dominance, Arithmetic method, Graphical method. **7 Hrs**

**Reference Books:**


**1)** Er. Premakumar Gupta and Dr. D.S.Hira, “Operations Research”, S Chand Publications, 2014.

**2)** Frederick S. Hillier and Gerald J. Lieberman, “Introduction to Operations Research”, 8/E, Tata McGraw Hill, 2005.

**3)** Wayne L. Winston, “Operations Research Applications and Algorithms”, 4/E, Thomson Course Technology, 2003.

**4)** Hamdy A T, “Operations Research: An Introduction”, 9/E, Pearson Publishers, 2014


## 21UCSE511 Computer Graphics (3-0-0) 3

**Contact Hours: 39**

**Course Learning Objectives (CLOs):** This course introduces fundamental principles of computer graphics, its architecture and how transformations of objects are carried out. It facilitates students to identify good design principles to solve challenges involved in simulating real world objects/conditions. It also provides the students to learn and apply the aspects of interaction with computer and exposes them to open-source tools like OpenGL.

**Course Outcomes (COs):**

| Description of the Course Outcome: At the end of the course the student will be able to: | | Mapping to POs(1-12) / PSOs(13-16) | | |
|---|---|---|---|---|
| | | Substantial Level (3) | Moderate Level (2) | Slight Level (1) |
| **CO-1** | **Explain** the principles of Computer Graphics Architecture used in industry relevant tool like OpenGL. | - | 13,14 | 1,2,12 |
| **CO-2** | **Explain** the design objectives of APIs used in OpenGL. | - | 14,12 | 1,5 |
| **CO-3** | **Apply** input interaction techniquesused in graphics environment. | 14 | 2 | - |
| **CO-4** | **Apply** affine transformations to solve problems relating to object transformations. | 13 | 14 | 1 |
| **CO-5** | **Discriminate** the views of objects in parallel and perspective projections under various lighting conditions. | - | 13 | 14 |
| **CO-6** | **Formulate** mathematical strategies for scan conversion algorithms to realize basic primitives, and **represent** curve and surfaces. | 13,14 | - | 1,12 |

| POs/PSOs | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Mapping Level | 1.0 | 1.5 | - | - | 1.0 | - | - | - | - | - | - | 1.3 | 2.5 | 2.1 | - | - |

**Pre-requisites:** Knowledge of

- Basic Engineering Graphics
- Linear Algebra (Scalars, Vectors, Matrices)
- Algorithms and C programming.

**Contents:**

**Unit-I**

**Introduction:** Applications of computer graphics; A graphics system; Images: Physical and synthetic; Imaging systems; The synthetic camera model; The programmer's interface; Graphics architectures; Programmable pipelines; Performance characteristics. Graphics Programming: The Sierpinski gasket; Programming two dimensional applications

**Input and Interaction:** Interaction; Input devices; Clients and servers; Display lists; Display lists and modeling; Programming event-driven input; Menus; Picking; A simple CAD program; Building interactive models; Animating interactive programs; Design of interactive programs; Logic operations. **9 Hrs**

**Unit-II**

**Geometric Objects and Transformations:** Scalars, points, and vectors; Coordinate systems and frames; Modeling a colored cube; Affine transformations; Rotation, translation and scaling; Transformations in homogeneous coordinates; Concatenation of transformations; Interfaces to three-dimensional applications. **9 Hrs**

**Unit-III**

**Viewing and Lighting:** Classical and computer viewing; Viewing with a computer; Positioning of the camera; Simple projections; Projections in OpenGL, Hidden surface removal; Parallel-projection matrices; Perspective-projection matrices.
**OpenGL:** Introduction to OpenGL; Programming two-dimensional Application; The OpenGL API; Primitives and Attributes; Color; Control Functions. **7 Hrs**

**Unit-IV**

**Basic Raster Graphics Algorithms for drawing 2D primitives:** Scan converting lines, circles, Filling Rectangles, Polygons; Clipping in a raster world; Clipping lines, polygons; Anti-aliasing **7 Hrs**

**Unit-V**

**Lighting and Shading:** Light and matter; Light sources; The Phong lighting model.
**Representing Curves and Surfaces:** Parametric Cubic Curves – Hermite Curves, Bézier Curves. **7 Hrs**

**Conduction of Practical Sessions:** Practical Sessions to be held with the focus of learning Open Source Tools like OpenGL and its API features. For the successful completion of the course, students are expected to undertake project to explore advanced features of Open Source Tools like OpenGL.

**Reference Books:**


1) Edward Angel, “Interactive Computer Graphics A Top-Down Approach with OpenGL”, 5/E, Addison-Wesley, 2008
2) James D Foley, Andries Van Dam, Steven K Feiner & John F Hughes, “Computer Graphics Principles and Practice”, 2/E, Addison-Wesley, 1997.
3) Edward Angel & Dave Shreiner, “Interactive Computer Graphics A Top-Down Approach with Shader-Based OpenGL”, 6/E, Addison-Wesley, 2012
4) F.S. Hill, Jr., “Computer Graphics Using OpenGL”, 2/E, Pearson Education, 2005
5) Donald Hearn and Pauline Baker, “Computer Graphics- OpenGL Version”, 2/E, Pearson Education, 2003

**21UCSE512 Advanced Graph Theory (3-0-0) 3**

**Contact Hours: 39**

**Course Learning Objectives (CLOs):** This course focuses on following learning perspectives:

- Isomorphism, Connected and Disconnected Graphs.
- Spanning trees and Cutsets.
- Planarity of Graphs.
- Chromatic Number and Polynomial.
- Directed Graphs.

**Course Outcomes (COs):**

| **Description of the Course Outcome:** At the end of the course the student will be able to: | | **Mapping to POs(1-12)/ PSOs (13-16)** | | |
|---|---|---|---|---|
| | | **Substantial Level (3)** | **Moderate Level (2)** | **Slight Level (1)** |
| **CO-1** | **Determine** whether the two graphs are isomorphic or not. | - | 1,2 | - |
| **CO-2** | **Prove** the properties of trees, cutsets and **determine** the spanning trees of a graph. | - | 2,13 | 1 |
| **CO-3** | **Determine** the planarity and dual of a graph. | - | 2,13 | 1 |
| **CO-4** | **Determine** the chromatic polynomial and chromatic number of a graph. | - | 2,13 | 1 |
| **CO-5** | **Explain** the principles of directed graphs and **represent** the digraphs in different forms. | - | 2,13 | 1 |

| **POs/PSOs** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** | **13** | **14** | **15** | **16** |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Mapping Level** | 1.0 | 2.0 | - | - | - | - | - | - | - | - | - | - | 2.0 | - | - | - |

**Pre-requisites:** Knowledge of Discrete Mathematical Structures.

**Contents:**

**Unit-I**

**Introduction to Graphs:** Definition, Finite and Infinite Graphs, Incidence and Degree, Isolated vertex, Pendant Vertex, and Null graph.

**Paths and Circuits:** Isomorphism, Subgraphs, Walks, Paths, and Circuits, Connected and Disconnected Graphs, Operations on Graphs, Euler graphs, Hamiltonian Paths and Circuits **8 Hrs**

**Unit-II**

**Trees and Fundamental Circuits:** Trees, Fundamental properties of trees, Pendant vertices in a tree, Distance and Centres in a tree, Rooted and Binary trees, Spanning Trees, Finding all spanning trees in a graph, Spanning trees in a weighted graph, The matrix – tree theorem.

**Cutsets:** Introduction, Properties, All cutsets in a graph, Fundamental circuits and cutsets**,** The Chinese Postman problem. **8 Hrs**

**Unit-III**

**Planar and Dual Graphs:** Introduction to Planar graphs, Kuratowski's two graphs, Different representations of a Planar graph, Detection of Planarity, Geometric Dual. **8 Hrs**

**Unit-IV**

**Coloring:** Chromatic number, Chromatic Partitioning, Chromatic Polynomial, Matchings, Coverings, The Four Color Problem, Brooks Theorem. **7 Hrs**

**Unit-V**

**Directed Graphs:** Definition, Types of Digraphs, Binary Relations, Directed Paths and Connectedness, Euler Digraphs, Trees with Directed Edges, Fundamental Circuits in Digraphs, Matrices A,B, and C of Digraphs, Adjacency matrix of Digraph, Random graphs. **8 Hrs**

**Reference Books:**


**1)** Narasingh Deo,"Graph Theory with Applications to Engineering and Computer Science", Dover Publications Inc 2016.

**2)** R.Balakrishnan & K.Ranganathan, "A Text Book of Graph Theory", 1/E, Springer Publications, 2000

**3)** Ralph P. Grimaldi "Discrete and Combinatorial Mathematics", 5/E, Pearson Education. 2006.

**4)** Kenneth H. Rosen "Discrete Mathematics and its Applications", 7/E, McGraw Hill, 2012.


## 21UCSE513 System Simulation and Modeling (3-0-0) 3

**Contact Hours: 39**

**Course Learning Objectives (CLOs):** This course focuses on following learning perspectives:

- To introduce students to basic simulation methods and tools for modelling and simulation of continuous, discrete and combined systems.
- The ability to analyze a system and to make use of the information to simulate various systems to improve the performance.
- Analytical methods (Markov Models and Queuing Networks) and simulation techniques (Monte Carlo Techniques and Event Driven Simulation) applied in performance modelling of communication systems and networks.

**Course Outcomes (COs):**

| Description of the Course Outcome: At the end of the course the student will be able to: | | Mapping to POs(1-12) / PSOs (13-16) | | |
|---|---|---|---|---|
| | | Substantial Level (3) | Moderate Level (2) | Slight Level (1) |
| **CO-1** | **Conceptualize** the discrete system simulation with the aid of real time examples. | - | 5,13 | 1,2,3,15 |
| **CO-2** | **Write** a program to simulate a given scenario. | - | 5,13,14 | 1,2,3,15 |
| **CO-3** | **Apply** different statistical models available in simulation and their usage in specific applications. | - | 5,13,14 | 1,2,3,15 |
| **CO-4** | **Design** the queuing systems and **evaluate** the performance. | - | 5,13 | 1,2,3,15 |
| **CO-5** | **Identify** the distribution of data to adhere to fitness test and also analyze the corresponding simulation. | - | 5,13 | 1,2,3,15 |

| POs/PSOs | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Mapping Level | 1.0 | 1.0 | 1.0 | - | 2.0 | - | - | - | - | - | - | - | 2.0 | 2.0 | 1.0 | - |

**Pre-requisites:** Knowledge of Probability and Statistics

**Contents:**

**Unit-I**

**Introduction:** When simulation is the appropriate tool and when it is not appropriate; Advantages and disadvantages of Simulation; Areas of application; Systems and system environment; Components of a system; Discrete and continuous systems; Model of a system; Types of Models; Discrete-Event System Simulation; Steps in a Simulation Study. Simulation examples: Simulation of queuing systems; Simulation of inventory systems; other examples of simulation. General Principles. **7 Hrs**

**Unit-II**

**Concepts in Discrete-Event Simulation:** The Event-Scheduling, Time-Advance Algorithm, World Views, Manual simulation Using Event scheduling; List processing. **7 Hrs**

**Unit-III**

**Statistical Models In Simulation:** Review of technology and concepts; Useful statistical models; discrete distributions; Continuous distributions; Poisson process; Empirical distributions. **8 Hrs**

**Unit-IV**

**Queuing Models:** Characteristics of queuing systems; Queuing notation; Long-run measures of performance of queuing systems; Steady state behavior of M/G/1 queue; Networks of queues. **8 Hrs**

**Unit-V**

**Input Modeling:** Data Collection; Identifying the distribution with data; Parameter estimation; Goodness of Fit Tests; Fitting a non-stationary Poisson process; selecting input models without data; Multivariate and Time-Series input models.

**Estimation of absolute performance:** Types of simulations with respect to output analysis; stochastic nature of output data; Measures of performance and their estimation; Output analysis for terminating simulations; Output analysis for steady-state simulations. **9 Hrs**

**Reference Books:**


**1)** Banks, John S. Carson II, Barry L. Nelson- David M. Nicol, "Discrete-Event System Simulation", 5/E, Pearson Education, 2013.

**2)** J. A. Sokolowski, C.M. Banks, "Principles of Modeling and Simulation: A multidisciplinary Approach", John Wiley & Sons Publications, 2011.

**3)** Sheldon M. Ross, "Simulation", 4/E, Elsevier, 2006.

**4)** D.S.Hira, "System Simulation", 2/E, S.Chand Publications, 2008.


**21UCSL503 Database Management Systems Lab (0-0-2)1**

**Contact Hours: 24**

**Course Learning Objectives (CLOs):** This laboratory course focuses on hands on experience on creation of data models, database design, programming using SQL/PL-SQL and development of an application using any high level language.

**Course outcomes (COs):**

| **Description of the Course Outcome:** At the end of the course the student will be able to: | | **Mapping to POs(1-12)/ PSOs (13-16)** | | |
|---|---|---|---|---|
| | | **Substantial Level (3)** | **Moderate Level (2)** | **Slight Level (1)** |
| **CO-1** | **Identify** entities, attributes, their relationships and **prepare** ER model for the given problem. | 2, 3, 13 | - | 15 |
| **CO-2** | **Design** database in appropriate normal form for the given problem. | 2,3,13 | - | 15 |
| **CO-3** | **Write** SQL queries using all the standard clauses, correlated queries, aggregate and date related functions for the given application scenario. | 2,3,13,14 | - | 15 |

| | | | | |
|---|---|---|---|---|
| **CO-4** | **Write** the programs using advanced features of data base programming that includes PL/SQL, Cursors, Triggers, Stored procedures and Functions for given application scenario. | 2,3,13,14 | - | 15 |

| POs/PSOs | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Mapping Level | - | 3.0 | 3.0 | - | - | - | - | - | - | - | - | - | 3.0 | 3.0 | 1.0 | - |

**Prerequisites:** Registration for / Completion of: DBMS theory course

**Contents**:

**Part A:** Multiple standalone preparatory exercises to meet the course outcomes.

**Part B:** Project work (Standalone application or web enabled application) based on knowledge gained from theory course and part-A hands on experience.

**21UCSL504 Web Technology Lab (0-0-2)1**

**Contact Hours: 24**

**Course Learning Objectives (CLOs):** This course emphasizes on practice-based learning. It enables the students to understand the web application architecture and use the state-of-the-art technologies to provide web-based solutions.

**Course Outcomes (COs):**

| **Description of the Course Outcome:** At the end of the course the student will be able to: | | **Mapping to POs(1-12)/ PSOs (13-16)** | | |
|---|---|---|---|---|
| | | **Substantial Level (3)** | **Moderate Level (2)** | **Slight Level (1)** |
| **CO-1** | **Explain** the fundamentals of web technology and **develop** static web pages using XHTML | 13 | 14 | 1,5,9 |
| **CO-2** | **Explain** the significance of CSS and its associated frameworks in designing web pages and **develop** web pages using CSS and its associated frameworks | 13 | 14 | 1,5,9 |
| **CO-3** | **Design** and **develop** an interactiveweb application using jQuery, XHTML and CSS | 13 | 14 | 1,5,9 |

| | | | | |
|---|---|---|---|---|
| **CO-4** | **Design** and **develop** dynamic webapplications using server-sideprogramming and **demonstrate** databaseconnectivity | 13 | 14 | 1,5,9 |
| **CO-5** | **Explain** the use of AJAX technology to interact with the server asynchronously and **demonstrate** the use of AJAX in web applications | 13 | 14 | 1,5 |
| **CO-6** | **Explain** the role of web services in representing the data in the standard formats as per the requirements | - | 5 | 1 |
| **CO-7** | **Explain** the future of World WideWeb and its associated technologies | - | 5 | 1,12 |

| POs/PSOs | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Mapping Level | 1.0 | - | - | - | 1.5 | - | - | - | 1.0 | - | - | 1.0 | 3.0 | 2.0 | - | - |

**Prerequisites:** Registration/ completion of Web Technologies and Applications course

**Suggested Platforms:**

- Any text editor (like notepad, HTML editors, IDEs like Visual Studio code) etc
- Any simple framework for web development (like WAMP, XAMPP, etc)

**Contents:**

Minimum one exercise to cover each course outcome specified above. Minimum 5 experiments to be completed by each student independently covering all course outcomes defined for this course. Course teacher has to publish list of experiments along with individual outcome for every experiment, at the beginning of the semester. Examiner may set any problem based on the published term work during examination.

**Reference Books:**

**1)** Robert W. Sebesta, “Programming the World Wide Web”, 7/E PearsonEducation, 2014

**2)** Noel Rappin, “Modern CSS with Tailwind: flexible styling without the fuss”, 1/E, The Pragmatic Bookshelf, 2021

**3)** Jonathan Chaffer and Karl Swedberg, "Learning jQuery", 3/E, Pack Publishing, 2011

**4)** Karin K. Breitman, Marco Antonio Casanova and Walter Truszkowski, "Semantic Web: Concepts, Technologies and Applications", Springer International Edition, 2007

**5)** Nicholas C Zakas et al, "Professional AJAX", Wrox Publication, 2007

## 21UAEE520 Data Analysis and Visualization Tools (2-0-0) 2

**Contact Hours: 26**

**Course Learning Objectives (CLOs):** This course focuses on to interpret data plots and understand visualization concepts, to explore the relationship between two continuous variables using scatter plots and line plots, to translate and present data correlations in a simple way.

**Course Outcomes (COs):**

| **Description of the Course Outcome:** At the end of the course the student will be able to: | | **Mapping to POs(1-12) / PSOs (13-16)** | | |
|---|---|---|---|---|
| | | **Substantial Level (3)** | **Moderate Level (2)** | **Slight Level (1)** |
| **CO-1** | **Appreciate** the key concepts of data visualization. | 4 | 3 | 6 |
| **CO-2** | **Design** effective data visualization for visual mapping. | 4 | 3 | 6 |
| **CO-3** | **Demonstrate** skills on creating visual representation of data. | 4 | 3 | 6 |
| **CO-4** | **Explain** visualization classification and its techniques. | 4 | 3 | 6 |
| **CO-5** | **Demonstrate** skills in creating different types of representation. | 4 | 3 | 6 |

| **POs/PSOs** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** | **13** | **14** | **15** | **16** |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Mapping Level** | - | - | 2.0 | 3.0 | - | 1.0 | - | - | - | - | - | - | - | - | - | - |

**Pre-requisites:** Knowledge of programming

**Contents:**

**Unit-I**

An overview of data and collection, Preprocessing and Representation, Data Visualization and Tools – Introduction. **6 Hrs**

**Unit-II**

Introduction to visualization and visual perception, visual representation of data, Gestalt principles, information overloads, Creating visual representations, visualization reference model, visual mapping, visual analytics, Design of visualization applications. **5 Hrs**

**Unit-III**

Classification of visualization systems, Interaction and visualization techniques misleading, Visualization of one, two and multi-dimensional data, text and text documents **5 Hrs**

**Unit-IV**

Visualization of groups, trees, graphs, clusters, networks, software, Metaphorical visualization **5 Hrs**

**Unit-V**

Visualization of volumetric data, vector fields, processes and simulations, Visualization of maps, geographic information, GIS systems, collaborative visualizations, evaluating visualizations **5 Hrs**

**Reference Books:**


**1)** Ward and Grinstein Keim, "Interactive Data Visualization: Foundations, Techniques, and Applications". A K Peters, Ltd, 1/E, 2010

**2)** Kieran Healy, "Data Visualization: A Practical Introduction", 1/E, 2018

**3)** Andy Krik, "Data Visualisation : A Handbook for Data Driven Design", 1/E, 2016


**21UCSL505 Minor Project - 1 (0-0-2)1**

**Contact Hours: 26**

**Course Learning Objectives (CLOs):** This course enables the student to identify the community expectations in terms of possible engineering solutions and prepare project proposal.

**Course Outcomes (COs):**

| **Description of the Course Outcome:** At the end of the course the student will be able to: | | **Mapping to POs(1-12) / PSOs (13-16)** | | |
|---|---|---|---|---|
| | | **Substantial Level (3)** | **Moderate Level (2)** | **Slight Level (1)** |
| **CO-1** | **Identify** the societal problems. | - | 2,6,7,9,12,13 | - |
| **CO-2** | **Analyze** real environment and **Formulate** the problem statement. | - | 2,9,12,13 | - |

| | | | | |
|---|---|---|---|---|
| **CO-3** | **Conduct** exhaustive literature survey | - | 2, 9,12,13 | - |
| **CO-4** | **Propose** sustainable engineering solutions / prototypes. | - | 3,5,7,12,13 | - |
| **CO-5** | **Prepare** the report and communicate effectively through presentation. | - | 8,9,10,12 | - |
| **CO-6** | **Manage** the project in terms of various resources in a particular discipline or in a multi-disciplinary domain. | - | 11 | - |

| POs | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Mapping Level** | - | 2.0 | 2.0 | - | 2.0 | 2.0 | 2.0 | 2.0 | 2.0 | 2.0 | 2.0 | 2.0 | 2.0 | - | - | - |

Guidelines for conduction:

1. Team consists of 4-5 students.
2. In consultation with guide, the team should carry out their project work.
3. Final evaluation is based on following evidence will be looked into and accordingly rubrics will be developed, such as problem statement, design, prototype/part of implementation, use of standard tools and techniques for testing and validation procedure.
4. This requires designated committee to monitor the process of conduction

**21UCSL506 Internship - I (Minimum 2 Weeks) 1**

**Contact Hours: 13**

**Course Learning Objectives (CLOs):** Internship provides an opportunity to get industry exposure to real time scenarios that include professional skill development programs and adhere to the professional standards.

**Course Outcomes (COs):**

| **Description of the Course Outcome:** At the end of the course the student will be able to: | | **Mapping to POs(1-12) / PSOs (13-16)** | | |
|---|---|---|---|---|
| | | **Substantial Level (3)** | **Moderate Level (2)** | **Slight Level (1)** |
| **CO-1** | **Apply** the Technical knowledge in real industrial scenarios. | 1, 8, 2,12 | - | - |
| **CO-2** | **Explain** the social, economic and | 3,5, 8, 13,12 | 11, 16 | 6, 7 |

| | | | | |
|---|---|---|---|---|
| | administrative considerations that influence the working environment of industrial organizations | | | |
| **CO-3** | **Inculcate** Engineer's responsibilities and Professional ethics. | 4,5, 8, 14,12 | 16 | - |
| **CO-4** | **Build** technical reports and **present** the work carried out. | 5, 8, 15,12 | - | - |

| POs/PSOs | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Mapping Level | 3.0 | 3.0 | 3.0 | 3.0 | 3.0 | 1.0 | 1.0 | 3.0 | - | - | 2.0 | 3.0 | 3.0 | 3.0 | 3.0 | 2.0 |

**Internship Guidelines:**

- Students are required to take up the internship in **offline** mode only.
- Students are **not permitted** to undergo internship in training institutes and non-technical organizations.
- Student need to obtain **NOC** (No Objection Certificate) from the head of the department and send it to the industry supervisor, who will approve and send the final acknowledgement letter to the students via Email/Post.

**Evaluation Guidelines:**

- **Evaluation by Industry:** Industry supervisor will evaluate overall performance of intern for 50 marksand will be considered as **CIE** marks.

- **SEE Evaluation through Seminar Presentation /VIVA VOCE at the Institute**
  The student will give a seminar based on his training report. The evaluation will be based on the following criteria:
  - Quality of content presented.
  - Proper planning for presentation.
  - Effectiveness of presentation
  - Depth of knowledge and skills.
  - Attendance record.

Seminar presentation will enable sharing knowledge & experience amongst students & teachers and develop communication skills and confidence in students.

**VI Semester**

| 21UCSC600 | Computer Networks | (3-0-0) 3 |
|---|---|---|

**Contact Hours: 39**

**Course Learning Objectives (CLOs):** This course focuses on the following learning perspectives:

- Evolution of network and Internet
- Protocols, applications pertaining to network and internet communication.
- Layered architecture, various network services and switching networks.
- Protocol design, implementation and performance issues.
- Various network management issues and possible remedies.
- Virtual networks for security issues in Internet Protocol (IP).

**Course Outcomes (COs):**

| **Description of the Course Outcome:** At the end of the course the student will be able to: | | **Mapping to POs(1-12)/ PSOs (13-16)** | | |
|---|---|---|---|---|
| | | **Substantial Level (3)** | **Moderate Level (2)** | **Slight Level (1)** |
| **CO-1** | **Explain** the evolution and core operating principles of computer network in terms of architecture, functions, switching techniques and protocols. | - | 1 | 2 |
| **CO-2** | **Analyze** the communication channels for errors and explain the fundamental concepts of analog and digital communication and principles of switching techniques and their applications. | - | 1 | 2, 13 |
| **CO-3** | **Explain** the working principles of peer to peer and logical link control protocols and MAC protocols in building networked space. | - | 1 | 2,13 |
| **CO-4** | **Analyze and implement** the protocols and algorithms of packet-switching networks. | 1 | 2 | 3, 13 |
| **CO-5** | **Explain** the working of TCP/IP layered model of communication and analyze the traffic management and security. | 1 | 1,2 | 3,13 |

| POs/PSOs | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Mapping Level | 2.2 | 1.4 | 1.0 | - | - | - | - | - | - | - | - | - | 1.0 | - | - | - |

**Pre-requisites:** Knowledge of Operating Systems

**Contents:**

**Unit-I**

**Communication Networks, Services, applications and layered architectures:** Evolution of Network Architecture and Services: Telegraph Networks and Message Switching, Telephone Networks and Circuit Switching, and the Internet, Computer Networks and Packet Switching, examples of Protocols, Services, and Layering: HTTP, DNS, and SMTP, TCP and UDP Transport Layer Services; The OSI Reference Model: The seven layers, Unified View of Layers, Protocols, and Services

**Overview of TCP/IP Architecture:** TCP/IP Architecture, TCP/IP Protocol: How the layers work together, Protocol Overview; Application Layer Protocols and TCP/IP Utilities **8 Hrs**

**Unit-II**

**Digital Transmission Fundamentals:** Digital Representation of Information: Block-Oriented Information, Stream Information; Why Digital Communications? Comparison of Analog and Digital Transmission, Basic properties of Digital Transmission Systems; Digital Representation of Analog Signals: Bandwidth of Analog Signals, Line Coding, Error Detection and Correction: Error Detection, Two-Dimensional Parity Checks, Internet Checksum, Polynomial Codes.

**Switching Networks:** Multiplexing: Frequency Division Multiplexing, Time Division Multiplexing, Wavelength - Division Multiplexing **8 Hrs**

**Unit-III**

**Peer-to-Peer Protocols and Data Link Layer:** Peer-to-Peer Protocols: Peer –to-Peer Protocols and Service Models; ARQ Protocols and Reliable Data Transfer Service: Stop-and-Wait ARQ, Go-Back-N ARQ, Selective Repeat ARQ; Other Peer-to-Peer Protocols; Data Link Controls: Framing; Point to Point Protocol; HDLC Data link Control.

**Medium Access Control Protocols and Local Area Networks:** The Medium Access Control Protocols: Multiple Access Communications; Random Access: ALOHA, Slotted ALOHA, CSMA, CSMA-CD; Scheduling Approaches to Medium Access Control: Reservation Systems, Polling, Token-Passing Rings **8 Hrs**

**Unit-IV**

**Packet-Switching Networks:** Network services and internal network operations, Packet network topology, Datagrams and virtual circuits, Routing in packet networks, Shortest-path routing, ATM networks, Applications of ATM. Traffic management using Fair queueing, RED, Leaky Bucket algorithm.

**TCP / IP:** TCP / IP architecture, The Internet protocol, IP addressing, subnet addressing, CIDR, IPv6, User datagram protocol, Transmission control protocol. Multicast routing **8 Hrs**

**Unit-V**

**Internet Routing Protocols:** RIP, OSPF, BGP, DHCP, NAT, Mobile IP**.** Virtual Private Networks. Overview of Security and cryptographic algorithms. **7 Hrs**

**Reference Books:**


1) Alberto Leon – Garcia & Indra Widjaja, "Communication networks – Fundamental Concepts and Key Architecture", 2/E, Tata McGraw Hill, 2005.
2) Nader F.Mir, "Computer and Communication Networks", 2/E, Pearson Education, 2009
3) Behrouz A Forouzon, "Data Communications and Networking", 10/E, Tata McGraw Hill, 2006
4) William Stallings, "Data and Computer Communication", 8/E, Pearson Education, 2013.


## 21UCSC601 Advanced Computer Architecture (3-0-0) 3

**Contact Hours: 39**

**Course Learning Objectives (CLOs):** This course focuses on the different computer architecture designs in the present scenario by considering performance parameters. Further, the concept of parallel processing and the relationship between parallelism and performance of different parallel architectures and software tools are emphasized. The concepts in memory hierarchy design and storage systems are also discussed.

**Course Outcomes (COs):**

| **Description of the Course Outcome:** At the end of the course the student will be able to: | | **Mapping to POs(1-12) / PSOs (13-16)** | | |
|---|---|---|---|---|
| | | **Substantial Level (3)** | **Moderate Level (2)** | **Slight Level (1)** |
| **CO-1** | **Analyze** the various performance characteristics of computer architectures with respect to theory of parallelism. | 4 | 1 | - |
| **CO-2** | **Explain** the working principles of processors, Memory hierarchy, Memory Consistency models. | - | 3 | 2 |
| **CO-3** | **Analyze** various cache memory organizations, **identify** the characteristics of shared memory organization and **illustrate** sequential and weak consistency models. | 5 | 1 | 13,16 |
| **CO-4** | **Identify** the generations of multi-computer architectures and **Analyze** the concept of message passing mechanisms. | 3 | 2 | 13,16 |

| | | | | |
|---|---|---|---|---|
| **CO-5** | **Detect** the instruction level parallelism and **explain** the role of compiler in exploitation of ILP. | 2 | 3 | 1 |

| POs/PSOs | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Mapping Level | 1.6 | 2.0 | 2.5 | 3.0 | 3.0 | - | - | - | - | - | - | - | 1.0 | - | - | 1.0 |

**Pre-requisites:** Knowledge of

- Computer Organization
- High level Programming
- Assembly Language Programming

**Contents:**

**Unit-I**

**Theory of Parallelism:** Parallel Computer Models,The State of Computing, Multiprocessors and Multicomputer, Multi-vector and SIMD Computers, Program and Network Properties,Conditions of Parallelism, Program Partitioning and Scheduling, Program Flow Mechanisms, System Interconnect Architectures, Principles of Scalable Performance, Performance Metrics and Measures, Parallel Processing Applications, Speedup Performance Laws, Scalability Analysis and Approaches. **8 Hrs**

**Unit-II**

**Hardware Technologies:** Processors and Memory Hierarchy: Advanced Processor Technology, Superscalar and Vector Processors, Memory Hierarchy Technology **8 Hrs**

**Unit-III**

**Bus, Cache, and Shared Memory**: Bus Systems, Cache Memory Organizations, Shared Memory Organizations, Sequential and Weak Consistency Models.
**Pipelining and Superscalar Techniques:** Linear Pipeline Processors, Nonlinear Pipeline Processors, Instruction Pipeline Design, Arithmetic Pipeline Design **8 Hrs**

**Unit-IV**

**Parallel and Scalable Architectures:** Multiprocessors and Multi computers - Multiprocessor System Interconnects, Cache Coherence and Synchronization Mechanisms, Three Generations of Multi computers, Message-Passing Mechanisms. **8 Hrs**

**Unit-V**

**Software for parallel programming:** Parallel Program Development and Environments: Parallel Programming Environments, Synchronization and Multiprocessing Modes. Instruction and System Level Parallelism, Instruction Level Parallelism: Computer Architecture, Basic Design Issues, Problem Definition, Model of a Typical Processor, Compiler-detected Instruction Level Parallelism, Operand Forwarding, Reorder Buffer, Register Renaming, Tomasulo's Algorithm, Branch Prediction, Limitations in Exploiting Instruction Level Parallelism, Thread Level Parallelism. **7 Hrs**

**Reference Books:**

**1)** Kai Hwang and Naresh Jotwani, "Advanced Computer Architecture (SIE): Parallelism, Scalability, Programmability", 3rd Edition, McGraw Hill Education, 2015
**2)** John L. Hennessy and David A. Patterson "Computer Architecture: A quantitative approach" 5th Edition, Morgan Kaufmann, Elseveir 2013.
**3)** Richard Y.Kain, "Advanced Computer Architecture: A System's Design Approach", Pearson Publications, 2015
**4)** John D. Carpinelli, "Computer Systems Organization and Architecture", Pearson Publications, 2001

## 21UCSC602 Automata Theory and Compiler Design (3-0-0) 3

**Contact Hours: 39**

**Course Learning Objectives (CLOs):**

This is a 3 credit course at undergraduate level enabling the students to understand structure of a compiler, representation of patterns and syntax using lexical rules and grammars respectively, employ finite state machines to solve problems in computing. Understand working of parsers, translation schemes, code optimization and code generation.

**Course Outcomes (COs):**

| **Description of the Course Outcome:** At the end of the course the student will be able to: | | **Mapping to POs(1-12) / PSOs (13-16)** | | |
|---|---|---|---|---|
| | | **Substantial Level (3)** | **Moderate Level (2)** | **Slight Level (1)** |
| **CO-1** | **Construct** a finite automaton for a given pattern and **explain** its working principles. | - | 1,2,3,13 | - |
| **CO-2** | **Write** regular expressions for given patterns and **explain** different techniques and principles used. | - | 1,2,3,13 | - |
| **CO-3** | **Explain** the basic structure and working principles of phases of compiler. | - | 13 | 1 |
| **CO-4** | **Write** grammar for a given language specification and **Design** a parser based on the appropriate parsing technique and **validate** the design. | 13 | 1,2,3,14 | 15 |
| **CO-5** | **Generate** an optimized intermediate code. | - | 1,2,3, 13 | 15 |
| **CO-6** | **Explain** the working principles of various run time environments like stack allocation, heap management and garbage collection technique used in compiler. | - | 1,2,3, 13 | 15 |
| **CO-7** | **Generate** optimized code for the given intermediate code. | - | 1,2,3, 13 | 15 |

| POs | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Mapping Level | 1.9 | 2.0 | 2.0 | - | - | - | - | - | - | - | - | - | 2.1 | 2.0 | 1.0 | - |

**Prerequisites:** Knowledge of: Programming Language (any).

**Contents:**

**Unit-I**

**Introduction to Finite Automata:** The central concepts of Automata theory – Alphabet, Strings & Languages. Finite Automata: Deterministic Finite Automata (DFA), Non-Deterministic Finite Automata (NFA), Equivalence of NFA and DFA, FA with Epsilon ( ϵ) transitions, Equivalence and Minimization of Automata

**Regular Expressions and languages:** Regular Expressions, Applications of Finite Automata and Regular Expressions.

**8 Hrs**

**Unit-II**

**Context-Free Grammars (CFG) and Languages (CFL):** Context-Free Grammars, Parse Trees, Applications of Context-Free Grammars, Ambiguity in Grammars and Languages.

**Pushdown Automata (PDA):** Definition of Pushdown Automata, The languages of a PDA,Normal forms for Context Free Grammar **8 Hrs**

**Unit-III**

**Introduction to Compilers:** Different Phases of Compilers, Comparison of Compilers and Interpreters. Top-down Parsing: Predictive parser. Bottom-up Parsing: LALR parser **8 Hrs**

**Unit-IV**

**Intermediate Code Generation and Optimizations:** Syntax-directed translation; Syntax-directed translation schemes, Variants of syntax trees; Three-address code; Types and declarations; Translation of expressions; Type checking; Control flow; Various techniques of machine independent optimization.

**8 Hrs**

**Unit-V**

**Run-Time Environments:** Storage Organization; Stack allocation of space; Access to non-local data on the stack; Heap management; Introduction to garbage collection.

**Code Generation:** Issues in the design of Code Generator; The Target language; Addresses in the target code; Basic blocks and Flow graphs; Optimization of basic blocks; A Simple Code Generator. **7 Hrs**

**Reference Books:**

1. John E. Hopcroft, Rajeev Motwani, Jeffrey D. Ullman Introduction to Automata Theory, Languages and Computation, Pearson Education, 3/E, 2013.

2. Peter Linz, An Introduction to Formal Languages and Automata, Narosa Publishing House, 5/E, 2011.
3. Alfred V Aho, Monica S. Lam, Ravi Sethi, Jeffrey D Ullman,"Compilers-Principles, Techniques and Tools", 2/E, Addison-Wesley, 2007.
4. D.M.Dhamdhere, "System Programming and Operating Systems", 2/E (Revised), Tata McGraw - Hill, 2009 reprint.

## 21UCSE607 Advanced Data Structures and Algorithms (3-0-0) 3

**Contact Hours: 39**

**Course Learning Objectives (CLOs):** This course focuses on following learning perspectives:

- Asymptotic and Amortized Analyses
- Linear sorting algorithms
- Advanced data structures such as Heaps, B-trees, Red-Black trees etc.
- String matching algorithms

**Course Outcomes (COs):**

| **Description of the Course Outcome:** At the end of the course the student will be able to: | | **Mapping to POs(1-12) / PSOs (13-16)** | | |
|---|---|---|---|---|
| | | **Substantial Level (3)** | **Moderate Level (2)** | **Slight Level (1)** |
| **CO-1** | **Analyze** the performance of the given algorithm using asymptotic notations and amortized techniques. | - | 1,2 | - |
| **CO-2** | **Explain** the working and assumptions of linear sorting methods and **apply** them to solve a given problem. | - | 2,13 | 1 |
| **CO-3** | **Compare** the working of string matching algorithms and **use** them appropriately in developing applications. | - | 2,13 | 1 |
| **CO-4** | **Build** and **perform** the operations on heap structures. | - | 2,13 | 1 |
| **CO-5** | **Build** and **perform** the operations on search structures. | - | 2,13 | 1 |
| **CO-6** | **Use** the hash tables for the implementation of dictionary operations. | - | 2,13 | 1 |
| **CO-7** | **Choose** the appropriate data structure and **use** relevant algorithms to solve problems in different domains including project management. | - | 2,13 | 1,11 |

| POs/PSOs | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Mapping Level | 1.1 | 2.0 | - | - | - | - | - | - | - | - | 1.0 | - | 2.0 | - | - | - |

**Pre-requisites:** Knowledge of

- Programming language (any)
- Data Structures
- Algorithms

**Contents:**

**Unit-I**

**Introduction to Complexity Analysis:** Review of Asymptotic notations and their properties, Amortized analysis – Aggregate, Accounting and Potential methods.
**Linear Sorts:** Counting Sort, Bucket Sort, and Radix Sorting with Analysis for all algorithms. **8 Hrs**

**Unit-II**

**String Matching:** Naive algorithm; Rabin-Karp algorithm; String matching with Finite automata, KMP algorithm, Boyer-Moore algorithm. **8 Hrs**

**Unit-III**

**Heap Structures:** Binomial heaps, Fibonacci heaps.
**Search Structures:** 2-3 trees, 2-3-4 trees, B-trees, $B^+$ trees, Red-black trees. **8 Hrs**

**Unit-IV**

**Hashing:** Direct Address Tables, Hash Tables, Collision Resolution by Chaining – Analysis, Hash Functions – Properties, Division and Multiplication methods, Universal Hashing, Open Addressing – Linear and Quadratic Probing, Double hashing. **8 Hrs**

**Unit-V**

**Applications:** Huffman coding, Garbage collection and compaction, Min-Cut Max-Flow algorithm, Activity networks. **7 Hrs**

**Reference Books:**


1. Thomas H.Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein,"Introduction to Algorithms", 3/E, Prentice Hall India, 2009.
2. E. Horowitz, S.Sahni and Dinesh Mehta, "Fundamentals of Data structures in C++", Galgotia, 2006.
3. Anany Levitin, "Introduction to the Design and analysis of algorithms", 3/E, Pearson Education, 2011

**21UCSE608 Data Science (3-0-0) 3**

**Contact Hours: 39**

**Course Learning Objectives (CLOs):** This course is a 3 credit undergraduate course focusing on the data and types, data preprocessing. The details of R language are discussed to enable the students to perform data analytics. It also discusses the supervised and unsupervised learning.

**Course Outcomes (COs):**

| **Description of the Course Outcome:** At the end of the course the student will be able to: | | **Mapping to POs(1-12)/ PSOs (13-16)** | | |
|---|---|---|---|---|
| | | **Substantial Level (3)** | **Moderate Level (2)** | **Slight Level (1)** |
| **CO-1** | **Explain** the flow process for processing data science problems and the data types. | - | 1, 2 | 13 |
| **CO-2** | **Apply** the preprocessing methods to the raw data to make it available for analysis. | 1, 2 | - | - |
| **CO-3** | **Use** the R language to perform the data visualization and the analysis using fundamental statistical technques. | 3 | - | 12 |
| **CO-4** | Explain the machine learning technques. | - | 1, 2 | 13 |
| **CO-5** | Explain the unsupervised learning technques. | - | 1, 2 | 13 |

| **POs/PSOs** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** | **13** | **14** | **15** | **16** |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Mapping Level** | 2.3 | 2.3 | 3.0 | - | - | - | - | - | - | - | - | 1.0 | 1.0 | - | - | - |

**Pre-requisites:** Knowledge of Statistics

**Contents:**

**Unit-I**

**Introduction:** Evolution, Roles, Stages in Data Science Project, Applications, Data Security and Privacy Issues

**Data:** Data types - Structured and Unstructured, Challenges with Unstructured data, Social media data, Multi modal data, Data Storage and Presentation **7 Hrs**

**Unit-II**

**Data Preprocessing:** Cleaning, Integration, Transformation, Reduction, Discretization.

**Techniques:** Correlation, Regression. Exploratory Analysis. **7 Hrs**

**Unit-III**

**R Language**: Basics, Control structures, Functions, Impoting Data.
**Graphics and Data Visualization:** Installing gglot2, Loading the data, Plotting the Data.
**Statistics and Machine Learning:** Basic Statistics, Regression, Clustering **9 Hrs**

**Unit-IV**

**Machine Learning:** Introduction, Regression, Classification, Gradient Descent. **8 Hrs**

**Unit-V**

**Unsupervised Learning:** Introduction, Agglomerative Clustering, Reinforcement Learning **8 Hrs**

**Reference Books:**

**1)** Chirag Shah, "A Hands on Introduction to Data Science", Cambridge University Press, 2020
**2)** Laura Igual and Santi Segui, "Introduction to Data Science", Springer International Publications, 2017
**3)** Richord Cotton "Learning R", O'Reilly Publications, 2013.

**21UCSE609 Natural Language Processing (3-0-0) 3**

**Contact Hours: 39**

**Course Learning Objectives (CLOs):**

This course focuses on the concepts, techniques and applications of natural language processing and text mining.

**Course Outcomes (COs):**

| **Description of the Course Outcome:** At the end of the course the student will be able to: | | **Mapping to POs(1 to 4)** | | |
|---|---|---|---|---|
| | | **Substantial Level (3)** | **Moderate Level (2)** | **Slight Level (1)** |
| **CO-1** | **Explain** the Concepts of mathematics and linguistic foundations of natural language processing | - | 3 | 6 |
| **CO-2** | **Perform** the various processing on the words. | - | 1 | 6 |
| **CO-3** | **Apply** the formal relationships between words. | - | 1 | 6 |

| | | | | |
|---|---|---|---|---|
| **CO-4** | **Represent** the meaning of the sentence and perform various semantic analysis. | - | 1 | 5,6 |
| **CO-5** | **Explain** the architecture and Operations of text mining. | - | 1 | 6 |
| **CO-6** | **Explain** and **apply** the text categorization and clustering techniques | - | 1 | 5,6 |

| POs/PSOs | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Mapping Level | 2.0 | - | 2.0 | - | 2.0 | 2.0 | - | - | - | - | - | - | - | - | - | - |

**Prerequisites:** Knowledge of

- Mathematical foundation, Linguistic essentials
- Finite Automata, Regular Expression and Grammar
- Machine LearningTechniques.

**Contents:**

**Unit-I**

**Introduction**: Basics of Natural Language Processing, Mathematical Foundations, Linguistic Essentials, Corpus Based Work

**Words:** Collocations, Empirical Laws, Basics of Text processing, Morphology and Finite State Transducers, Probabilistic Models of Pronunciation and Spelling, N – grams**,** Maximum Entropy models, Random Fields **8 Hrs**

**Unit-II**

**Syntax:** Word Classes and Part-of-Speech Tagging, Context Free Grammar for Languages, Parsing with Context Free Grammar, Markov Models, Lexicalized and Probabilistic Parsing **8 Hrs**

**Unit-III**

**Semantics:** Distributional Semantics, Lexical Semantics, Topic Models, Word Sense Disambiguation, Methodological Preliminaries, Supervised & Unsupervised Disambiguation, Dictionary-Based Disambiguation, Latent Dirichlet Allocation for text classification, Latent Semantic Indexing, Probabilistic Latent Semantic Indexing **8 Hrs**

**Unit-IV**

**Introduction to Text Mining:** Overview of text mining, General Architecture, Core Operations, Preprocessing techniques, Document classification, Information extraction, Evaluation of performance, sentiment analysis **8 Hrs**

**Unit-V**

**Text Categorization:** Machine Learning Approach to Text Categorization. Classification of Linked and Web Data, Text Clustering: Supervised and Unsupervised Clustering. Text Summarization Techniques. **8 Hrs**

**Reference Books:**


1. Dan Jurafsky, James H Martin, “Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics and Speech Recognition”, Pearson Education India,2nd edition (2013).
2. Christopher Manning “Foundations of Statistical Natural Language Processing”, MIT Press, July 1999.
3. Kao, Anne, and Steve R. Poteet, eds. “Natural language processing and text mining”, Springer Science & Business Media, 2007.
4. Ronen Feldman and James Sanger “The Text Mining Handbook: Advanced Approaches in Analyzing Unstructured Data”, Cambridge University Press, 2007.
5. Sholom Weiss, Nitin Indurkhya, Tong Zhang, Fred Damerau “Text Mining Predictive Methods for Analyzing Unstructured Information”, Springer, paperback 2010.


## 21UCSE610 Soft and Evolutionary Computing (3-0-0) 3

**Contact Hours: 39**

**Course Learning Objectives (CLOs):** This course focuses on the fundamental concepts and implementation of artificial neural network, Supervised and Unsupervised Learning Network, Associative Memory Networks, and Fuzzy Logic.

**Course Outcomes (COs):**

| **Description of the Course Outcome:** At the end of the course the student will be able to: | | **Mapping to POs(1-12) / PSOs (13-16)** | | |
|---|---|---|---|---|
| | | **Substantial Level (3)** | **Moderate Level (2)** | **Slight Level (1)** |
| **CO-1** | **Explain** the evolution and basic models of artificial neural networks | - | - | 1,2,13 |
| **CO-2** | **Implement** single-layer and multilayer feed-forward networks. | - | 3 | 14 |
| **CO-3** | **Explain** the training algorithms used for associative memory networks. | - | - | 1,2,13 |
| **CO-4** | **Explain** the architecture, training algorithm, flowchart depicting training process, and testing algorithm of different unsupervised networks. | - | - | 1,2,13 |
| **CO-5** | **Explain** the various operations and properties of classical and fuzzy sets. | - | - | 1,2,13 |

| **POs/PSOs** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** | **13** | **14** | **15** | **16** |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Mapping Level** | 1.0 | 1.0 | 2.0 | - | - | - | - | - | - | - | - | - | 1.0 | 1.0 | - | - |

**Pre-requisites:** Knowledge / Registration of Computer Networks.

**Contents:**

**Unit-I**

**Introduction:** Neural Networks, Application Scope, Fuzzy Logic, Genetic Algorithm, Hybrid Systems, Soft Computing.
**Artificial Neural Networks:** Fundamental Concepts, Evolution, Basic Models, Important Terminologies, McCulloch-Pitts Neuron, Linear Separability. **8 Hrs**

**Unit-II**

**Supervised Learning Network:** Introduction, Perception Networks, Adaptive Linear Neuron, Multiple Adaptive Linear Neuron, Back-Propagation Networks, Radial Basis Function Network. **8 Hrs**

**Unit-III**

**Associative Memory Networks:** Introduction, Training Algorithms for Pattern Association, Associative Memory Networks – Basic, Hetero associative, Bidirectional, Hopfield, Iterative. **8 Hrs**

**Unit-IV**

**Unsupervised Learning Networks:** Introduction, Fixed Weight Competitive Nets, Kohonen Self-Organizing Feature Maps, Learning Vector Quantization, Conter propagation networks. **8 Hrs**

**Unit-V**

**Fuzzy Logic:** Introduction, Classical Sets, Fuzzy Sets
**Classical Relations and Fuzzy Relations:** Introduction, Cartesian Product of Relation, Classical Relation, Fuzzy Relations. **7 Hrs**

**Reference Books:**


1. S.N.Shivanandam and S.N.Deepa, “Principles of Soft Computing”, 2/E, Wiley India, 2011.
2. N.P.Padya and S.P.Simon, “Soft Computing with MATLAB Programming”, Oxford University Press, 2015.
3. Saroj Kaushik and Sunita Tiwari, “Soft Computing – Fundamentals, Techniques, and Applications”, 1/E, McGraw Hill, 2018

## 21UCSE611 Software Testing (3-0-0) 3

**Contact Hours: 39**

**Course Learning Objectives (CLOs):** This course discusses the notations and terminologies used in software testing with Principles of programming, Software Development Process from the perspective of quality, maintenance, testing, programming style. Further, it focuses on System quality through: requirements, design, coding, verification and validation procedures and applications, case studies, use of tools / programming techniques for design and implementation.

**Course Outcomes (COs):**

| **Description of the Course Outcome:** At the end of the course the student will be able to: | | **Mapping to POs(1-12) / PSOs (13-16)** | | |
|---|---|---|---|---|
| | | **Substantial Level (3)** | **Moderate Level (2)** | **Slight Level (1)** |
| **CO-1** | **Map** inputs to the expected outputs of the system by understanding the system behavior represented in the form of: requirements and design specifications / Models. | 1, 13 | - | - |
| **CO-2** | **Design** test cases based on requirement specifications. | 2, 14 | 1 | - |
| **CO-3** | **Design** test cases based on traditional design specifications. | - | 3, 13 | - |
| **CO-4** | **Design** test cases based on Object specifications/Models (UML). | 13 | 3 | - |
| **CO-5** | **Write** script in any programming Language/Tools to implement test cases designed. | 14 | 15 | - |
| **CO-6** | **Design** test cases based on various testing strategies to check the correctness of computer program. | - | 15, 13 | - |

| **POs/PSOs** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** | **13** | **14** | **15** | **16** |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Mapping Level** | 2.5 | 3.0 | 2.0 | - | - | - | - | - | - | - | - | - | 2.5 | 3.0 | 2.0 | - |

**Pre-requisites:** Knowledge of

- Programming language
- Software Engineering
- Object Oriented System Analysis and Design

**Contents:**

### Unit-I

**Programming Practice:** A Testing Perspective: Programming Principles & Guidelines, Coding Processes, Refactoring, Verifications & Metrics, Style- Names, Expression and Statements, Consistency and Idioms, Function Macros, Debugging, Performance, Portability.

**Basics of Software Testing:** Human Errors and Testing, Software Quality, Requirements, Behavior and Correctness, Correctness versus Reliability, Testing and Debugging, Test Metrics, Software and Hardware Testing, Testing and Verification, Defect Management, Execution History, Test generation Strategies, Static Testing. Model-Based Testing and Model Checking, Control-Flow Graph, Types of Testing, The Saturation Effect. **9 Hrs**

**Unit-II**

**Test Generation from Requirements:** Introduction; The Test-Selection Problem; Equivalence Partitioning; Boundary Value Analysis, Category-Partition Method, Cause-Effect Graphing, Test Generation from Predicates. **7 Hrs**

**Unit-III**

**Dependence, Data Flow Models and Data Flow Testing:** Definition-Use pairs; Data flow analysis; Classic analysis; From execution to conservative flow analysis; Data flow analysis with arrays and pointers; Inter-procedural analysis; Overview of data flow testing; Definition- Use associations; Data flow testing criteria; Data flow coverage with complex structures; The infeasibility problem. **7 Hrs**

**Unit-IV**

**Structural Testing:** Overview; Statement testing; Branch testing; Condition testing, Path testing; Procedure call testing; Comparing structural testing criteria; The infeasibility problem.

**Test Case Selection and Adequacy Test Execution:** Overview; Test specification and cases; Adequacy criteria; Comparing criteria; Overview of test execution; From test case specification to test cases; Scaffolding; Generic versus specific scaffolding; Test oracles; Self-checks as oracles; Capture and replay **9 Hrs**

**Unit-V**

**Testing Object Oriented Software:** Issues in Testing OO Software, Intra Class Testing, Testing with State Machine Models, Inter-Class Testing, Structural Testing of Class **7 Hrs**

**Reference Books:**


1. Brain W. Kernighan & Rob Pike, “The Practice of Programming”, Pearson education, 2008.
2. Pankaj Jalote & Narosa, “An Integrated Approach to Software Engineering”, 3/E, Publishing House
3. Edward Kit, “Software Testing in the Real World”, Pearson Education, 2006
4. Aditya P Mathur, “Foundations of Software Testing”, Pearson Education, 2008.
5. Mauro Pezze, Michal Young, John Wiley & Sons, “Software Testing and Analysis: Process, Principles and Techniques”, 2008.

| 21UCSE612 | Pattern Recognition | (3-0-0) 3 |
|---|---|---|

**Contact Hours: 39**

**Course Learning Objectives (CLOs):** This course focuses on the following learning objectives:

- Fundamentals of pattern recognition system
- Feature extraction and pattern classification algorithms.
- Unsupervised classification or clustering techniques
- Applications of pattern classification algorithm for a pattern recognition problem

**Course Outcomes (COs):**

| **Description of the Course Outcome:** At the end of the course the student will be able to: | | **Mapping to POs(1-12)/ PSOs (13-16)** | | |
|---|---|---|---|---|
| | | **Substantial Level (3)** | **Moderate Level (2)** | **Slight Level (1)** |
| **CO-1** | **Explain** the basic structure and inherent issues of the pattern recognition problems. | - | 1, 13 | 2, 3, 4 |
| **CO-2** | **Apply** feature types and classification techniques like Bayesian classifier and its derivatives in solving problems. | - | 1, 13 | 2, 3, 4 |
| **CO-3** | **Compute** the probability density using parametric, non-parametric and linear discriminant functions. | - | 1, 13 | 2, 3, 4 |
| **CO-4** | **Distinguish** supervised learning methods from the unsupervised ones and **apply** learning methods to the classifier design. | - | 1, 13 | 2, 3, 4 |
| **CO-5** | **Use** non metric methods to classify the models that can be described by logical rules. | - | 1, 13 | 2, 3, 4 |
| **CO-6** | **Apply** a suitable clustering method to solve a given problem. | - | 1, 13 | 2,3, 4 |

| POs/PSOs | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Mapping Level | 2.0 | 1.0 | 1.0 | 1.0 | 2.0 | - | - | - | - | - | - | - | 2.0 | - | - | - |

**Pre-requisites:** Knowledge of Statistics, Linear Algebra and Programming

**Contents:**

**Unit-I**

**Introduction:** What is pattern recognition? Pattern Recognition System; The Design Cycle; Learning and Adaptation. Clustering vs. Classification; Applications;
**Features:** Feature vectors - Feature spaces - Problem of feature identification Feature selection and feature extraction. **8 Hrs**

## Unit-II

**Bayesian Decision Theory:** Introduction, Bayesian Decision Theory; Continuous Features, Minimum error rate classifiers, discriminant functions, and decision surfaces; The normal density; Discriminant functions for the normal density.

**Maximum-Likelihood and Bayesian Parameter Estimation:** Introduction; maximum-likelihood estimation; Bayesian Estimation; Bayesian parameter estimation: Gaussian Case, general theory; Hidden Markov Models. **8 Hrs**

## Unit-III

**Non-Parametric Techniques:** Introduction; Density Estimation; Parzen windows; K Nearest- Neighbor Estimation; The Nearest- Neighbor Rule; Metrics and Nearest-Neighbor Classification.

**Linear Discriminant Functions:** Introduction; Linear Discriminant Functions and Decision Surfaces; Generalized Linear Discriminant Functions; The Two-Category Linearly Separable case; Minimizing the Perception Criterion Functions; Relaxation Procedures; Non-separable Behavior; Minimum Squared-Error procedures; The Ho-Kashyap procedures. **8 Hrs**

## Unit-IV

**Support Vector Machines and Kernel based method**: Introduction, obtaining the optimal hyperplane SVM formulation with slack variables; nonlinear SVM classifiers Kernel Functions for nonlinear SVMs; Mercer and positive definite Kernels. Support Vector Regression and ε-insensitive Loss function, examples of SVM learning. **8 Hrs**

## Unit-V

**Non-Metric Methods:** Introduction; Decision Trees; CART; Recognition with Strings; Grammatical Methods.

**Unsupervised Learning and Clustering:** Introduction; Mixture Densities and Identifiably; Maximum-Likelihood Estimates; Application to Normal Mixtures; Unsupervised Bayesian Learning; Data Description and Clustering; Criterion Functions for Clustering. **7 Hrs**

**Reference Books:**


1. Richard O. Duda, Peter E. Hart, & David G.Stork, “Pattern Classification”, 2/E, Wiley-Interscience, 2012.
2. Earl Gose, Richard Johnsonbaugh, and Steve Jost, “Pattern Recognition and Image Analysis”, Pearson Education, 2007.
3. V Susheela Devi & M NarsimhaMurthy, “Pattern Recognition - An Introduction”, Universities Press, 2011.

**21UCSO613 Cloud Computing (3-0-0) 3**

**Contact Hours: 39**

**Course Learning Objectives (CLOs):** This course focuses on the following learning objectives:
Understand the cloud computing paradigm. Recognize its various forms and implement some cloud computing features.

**Course Outcomes (COs):**

| **Description of the Course Outcome:** At the end of the course the student will be able to: | | **Mapping to POs(1-12)/ PSOs (13-16)** | | |
|---|---|---|---|---|
| | | **Substantial Level (3)** | **Moderate Level (2)** | **Slight Level (1)** |
| **CO-1** | **Explain** cloud computing philosophy & computing paradigms. | - | 1,4 | 5 |
| **CO-2** | **Distinguish** and **demonstrate** cloud service models and their risks. | - | 2,4,5 | - |
| **CO-3** | **Explain** and **use** SLA life cycle and agreement structure to build simple applications. | - | 4,8 | 7 |
| **CO-4** | **Explain** and **build** simple security models using cloud security principles. | - | 1,2,4 | 5 |
| **CO-5** | **Develop** and **deploy** cloud solution for the given application scenario. | 1, 2, 3 | 4, 5 | 7 |

| **POs/PSOs** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** | **13** | **14** | **15** | **16** |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Mapping Level** | 2.3 | 2.3 | 3.0 | 2.0 | 1.5 | - | 1.0 | 2.0 | - | - | - | - | - | - | - | - |

**Pre-requisites:** Knowledge of

- Operating Systems, Computer Architecture, Computer Networks & Web Technologies.
- Programming languages- Java, Web programming

**Contents:**

**Unit-I**

**Introduction:** Paradigms in Computing, Parallel Computing, Distributed Computing, Grid Computing, Service Computing; Service Oriented Architecture (SOA), Web Services. **7 Hrs**

**Unit-II**

**Basics of Cloud Computing:** Definition, Characteristics, Architecture, Components, Service Models, and Deployment Models.
**Virtualization:** Server, Storage, Network, Desktop Hypervisor, Virtual Machine, Multi-tenancy, Opportunities and Risks. **7 Hrs**

## Unit-III

**Service Level Agreement (SLA):** Definition, Types of SLA, SLA Life Cycle, Issues Related to Cloud SLA, SLA Frameworks: WS-Agreement, WSLA, WSOL, Slang, Bilateral Protocol; Translation of SLAs into Monitoring Specifications, Dynamic Creation of Monitoring Infrastructures, Penalty Management, Runtime Prediction.

**7 Hrs**

## Unit-IV

**Cloud Security**: Cloud Security Fundamentals, Vulnerability Assessment, Security and Privacy in Cloud.

**Cloud Computing Security Architecture:** Identity Management and Access Control, Autonomic Security; VM Specific Security Techniques. **9 Hrs**

## Unit-V

**Cloud Application Programming Models:** Cloud File Systems: GFS and HDFS, BigTable, HBase and Dynamo; Map Reduce Programming Model, Hadoop: Hadoop Fundamentals, Hama and other Hadoop Related Services. Cloud Application Development Platforms: Xen Hypervisor, Amazon Web Service, Windows Azure, Google App Engine, Eucalyptus, Open Stack, Open Nebula. **9 Hrs**

**Experiments**

1. Use of Cloud Web Services
2. Experiment on Cloud Storage using mobile
3. Explore Cloud platform like Amazon Web Service, Windows Azure, Google App Engine etc.

**Reference Books:**


1. A. T. Velte, “Cloud Computing - A Practical Approach”, McGraw Hills 2010
2. Tanenbaum & V. Steen, “Distributed Systems: Principles and Paradigms”, 2nd Edition, Pearson, 2015
3. David E.Y. Sarna, “Implementing and Developing Cloud Computing Applications”, CRC Press, 2011
4. R. Krutz & R. D. Vines, “Cloud Security”, Wiley-India, 2010
5. T. White, “Hadoop: The Definitive Guide”, 4th Edition, O’Reilly Media, 2015
6. Judith Hurwitz, R.Bloor, M.Kanfman, F.Halper, “Cloud Computing for Dummies”, Wiley India Edition
7. Rajkumar Buyya, James Broberg & Andrzej Goscinski, “Cloud Computing Principles and Paradigms”, Willey 2014.
8. Dan C Marinescu, “Cloud Computing Theory and Practice”, Elsevier (MK), 2013.

**21UCSO614 Ontology & Semantic Web (3-0-0) 3**

**Contact Hours: 39**

**Course Learning Objectives (CLOs):** This course focuses on following learning perspectives:

- Theoretical and practical aspects of ontology in semantic web.
- Concepts of RDF and web ontology language.
- Various patterns for developing and reusing ontologies.
- Models for knowledge management architecture in semantic web.

**Course Outcomes (COs):**

| **Description of the Course Outcome:** At the end of the course the student will be able to: | | **Mapping to POs(1-12)/ PSOs (13-16)** | | |
|---|---|---|---|---|
| | | **Substantial Level (3)** | **Moderate Level (2)** | **Slight Level (1)** |
| **CO-1** | **Explain** the working principles, architecture of semantic web and the importance of markup languages. | 1 | - | - |
| **CO-2** | **Design** and **analyze** ontologies using web ontology language. | - | 3 | - |
| **CO-3** | **Apply** the logic with web ontology language. | 15 | - | 5 |
| **CO-4** | **Analyze** scalable architectures and semantic web services. | - | 2 | - |
| **CO-5** | **Explain** the evolution of ontologies with semantic interpretation of information. | - | 12 | - |

| **POs/PSOs** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** | **13** | **14** | **15** | **16** |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Mapping Level** | 3.0 | 2.0 | 2.0 | - | 1.0 | - | - | - | - | - | - | 2.0 | | - | 3.0 | - |

**Pre-requisites:** Knowledge of HTML and Java

**Contents:**

**Unit-I**

**The semantic web vision:** Today's web, Semantic web Technologies, A layered approach, Structured web documents in XML, Introduction to RDF, RDF syntax, RDF schema, Axiomatic statements for RDF and RDF schema, Direct inference system on RDF and RDF schema. **9 Hrs**

**Unit-II**

**Ontology and Web ontology Language:** Introducing OWL, Discovering the Various Species of OWL, Exploring the Foundations of OWL Understanding OWL Essentials, Making Simple Assertions, Inconsistency, Examining Property Characteristics, Complex Classes, Understanding Why OWL Is Different, Developing OWL Ontologies. **8 Hrs**

**Unit-III**

**Ontology building and inference by logic:** Monotonic rules - syntax and semantics, Nonmonotonic rules - syntax, Rule markup in XML, Constructing and reusing ontologies, Semantic web knowledge management architecture, Querying using SPARQL, Jena ontology API, Reasoners. **8 Hrs**

**Unit-IV**

**Ontology scalable architectures and Discovering the semantic web services:** Discovering the Roles, Creating Semantics for Enterprise Systems, Scaling Semantic Web Tools, Patterns of Architectural Usage and Applicationdevelopment using OWL API. **7 Hrs**

**Unit-V**

**Ontology evolution and Semantic knowledge:** Ontology population and enrichment, Semantic representation of multimedia content, Ontology based semantics extraction from text and Images. A survey of semantic image and video annotation tools, Overview of linked data. **7 Hrs**

**Reference Books:**


1. Grigoris Antoniou and Frank van Harmelen, "A Semantic Web Primer", 2nd Edition, MIT Press, Cambridge, MA 2008
2. Jeffrey T.Pollock, "Semantic Web For Dummies", Wiley Publishing, Inc 2009
3. John Hebeler, Matthew Fisher, Ryan Blace, & Andrew Perez-Lopez, "Semantic Web Programming", Wiley Publishing, Inc.
4. Georgios Paliouras, Constantine Spyropoulos, & George Tsatsaronis (Eds.) "Knowledge-Driven Multimedia Information Extraction and Ontology Evolution", Springer, 2011


**21UCSO615 Introduction to Software Engineering (3-0-0) 3**

**Contact Hours: 39**

**Course Learning Objectives (CLOs):** This is a 3 credit, 39 contact hours course at undergraduate level focusing on knowing the process of software system development and enables students to develop software system using engineering techniques.

**Course Outcomes (COs):**

| **Description of the Course Outcome:** At the end of the course the student will be able to: | | **Mapping to POs(1-12)/ PSOs (13-16)** | | |
|---|---|---|---|---|
| | | **Substantial Level (3)** | **Moderate Level (2)** | **Slight Level (1)** |
| **CO-1** | **Describe** the properties of various critical systems and the quality parameters. | - | - | 16 |
| **CO-2** | **Describe** the different key practices of process models. | - | - | 3,13 |
| **CO-3** | **Identify** various system requirements and **prepare** | - | - | 1,2,13 |

| | | | | |
|---|---|---|---|---|
| | system specification reports to solve real life problems in various domains and **develop** domain expertise. | | | |
| **CO-4** | **Conceptualize** the system through **design** and **modeling** the system architecture, components and processes with quality and standards. | - | - | 1,2,3,5,13,10, 12 |
| **CO-5** | Verify and validate the given system using standard tools and techniques. | - | - | 5,15,10 |
| **CO-6** | **Manage** project in terms of risk, configuration/versions, Cost and Resources**.** | - | - | 9, 10,11 |

| POs/PSOs | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Mapping Level | 1.0 | 1.0 | 1.0 | - | 1.0 | - | - | - | 1.0 | 1.0 | 1.0 | 1.0 | 1.0 | - | 1.0 | 1.0 |

**Pre-requisites:** Knowledge of

- Basics of computer systems and its uses
- Programming language (any)

**Contents:**

**Unit-I**

**Overview:** Introduction: FAQ's about software engineering, Professional and ethical responsibility. Socio-Technical systems: Emergent system properties; Legacy systems.

**Critical Systems, Software Processes:** Critical Systems: A simple safety critical system; System dependability; Availability and reliability. Software Processes: Models, Process iteration, Process activities; The Rational Unified Process; Computer Aided Software Engineering. **7 Hrs**

**Unit-II**

**Requirement Engineering:** Software Requirements: Functional and Non-functional requirements; User requirements; System requirements; The software requirements document. Requirements Engineering Process: Feasibility studies; Requirements elicitation and analysis; Requirements validation; Requirements management. **7 Hrs**

**Unit-III**

**System models:** System Models: Context models; Behavioral models; Data models; Object models;.

**Software Design and Development:** Architectural Design: Modular decomposition styles; Control styles. Object-Oriented design. UI Design Issues.

**Rapid Software Development:** Agile methods; Extreme programming; Rapid application development. Software Evolution: Program evolution dynamics; Software maintenance; **9 Hrs**

**Unit-IV**

**Verification and Validation:** Verification and Validation: Software testing: System testing; Component testing; Test case design; Test automation. Testing Techniques: **9 Hrs**

**Unit-V**

**Software Quality & Project Management:** Various Software quality parameters Project Management activities; Project planning; Project scheduling; Risk management. Managing People: Selecting staff; Motivating people; Managing people; The People Capability Maturity Model. Software Cost Estimation: Estimation techniques, **7 Hrs**

**Reference Books:**


1. Ian Somerville, "Software Engineering", 10/E, Person Education, 2016.
2. Roger Pressman, "Software Engineering, Practitioners approach", 7/E, McGraw-Hill, 2010.
3. Bharat Bhushan Agarwal & Sumit Prakash Tayal, "Software Engineering", 2nd Edition, Firewal Media Publications, 2007.
4. A.A.Puntambekar, "Software Engineering & Quality Assurance", 1st Edition, Technical Publications Pune, 2010


| 21UCSO616 | Introduction to Data Structures | 3-0-0 |
|---|---|---|

**Contact Hours: 39**

**Course Learning Objectives (CLOs):** This course focuses on the following learning perspectives:

- Working of various basic data structures and their implementation.
- Implementation issues of data structure in programming language.
- Selection of the appropriate data structure for solving a given problem.

**Course Outcomes (COs):**

| **Description of the Course Outcome:** At the end of the course the student will be able to: | | **Mapping to POs(1-12) / PSOs (13-16)** | | |
|---|---|---|---|---|
| | | **Substantial Level (3)** | **Moderate Level (2)** | **Slight Level (1)** |
| **CO-1** | **Write** programs to solve problems using Pointers and Structures/Unions. | - | 14 | 1,3,15,16 |
| **CO-2** | **Write** programs to solve simple problems using stack and **explain** its working principles. | - | 14 | 1,3,15,16 |
| **CO-3** | **Write** programs to solve problems using queue and **explain** its working principles. | - | 14 | 1,3,15,16 |
| **CO-4** | **Write** programs to solve problems using Linked Lists and **explain** its working principles. | - | 14 | 1,3,15,16 |

| | | | | |
|---|---|---|---|---|
| **CO-5** | **Write** programs to solve problems using trees and **explain** its working principles. | - | 14 | 1,3,15,16 |

| POs/PSOs | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Mapping Level | 1.0 | - | 1.0 | - | - | - | - | - | - | - | - | - | - | 2.0 | 1.0 | 1.0 |

**Pre-requisites:** Problem Solving skills and knowledge of Programming in C language.

**Contents:**

**Unit-I**

**Structures, unions and Pointers:** Motivation for using structures. Pointer, access data from memory through pointer, pointer to structures. Motivation for dynamic memory requirement. Realizing arrays using pointer and dynamic memory allocation. Importance of memory management during allocation and de-allocation of memory. **8 Hrs**

**Unit-II**

**Stack:** Realization of stack and its operations using static and dynamic structures. Application of stack in converting an expression from infix to postfix and evaluating a postfix expression. Applications of Stacks. **8 Hrs**

**Unit-III**

**Queues:** Realization of queues (FIFO, Circular Queues, Double-ended queue, Priority queue) and its operations using static and dynamic data structures. Applications of Queues. **8 Hrs**

**Unit-IV**

**Lists:** Constructing dynamic data structures using self-referential structure. Operations on lists. Doubly Linked list. Application of Lists in real time. **8 Hrs**

**Unit-V**

**Trees:** Types of trees and their properties, Realization of trees using static and dynamic data structures. Operations on Binary trees and their applications (BST). **7 Hrs**

**Reference Books:**

1) Aaron M. Tenenbaum, Yedidyah Langsam & Moshe J. Augenstein, “Data Structures using C and C++”, Pearson Education, 2006
2) E. Balagurusamy, “Programming in ANSI C”, 7th Edition, Tata McGraw-Hill, 2016
3) Behrouz A. Forouzan & Richard F. Gilberg, “Computer Science: A Structured Programming Approach Using C”, 2nd Edition, Cengage Learning, 2003.

| 21UCSL603 | Computer Networks Lab | (0-0-2) 1 |
|---|---|---|

**Contact Hours: 12**

**Course Learning Objectives (CLOs):** This course focuses on the following learning perspectives:

- Configuration of networking devices.
- Troubleshooting IPv4 and IPv6 Addressing
- DHCP and DNS Servers

**Course Outcomes (COs):**

| **Description of the Course Outcome:** At the end of the course the student will be able to: | | **Mapping to POs(1-12)/ PSOs (13-16)** | | |
|---|---|---|---|---|
| | | **Substantial Level (3)** | **Moderate Level (2)** | **Slight Level (1)** |
| **CO-1** | **Configure** the Cisco networking devices like routers, switches, and hubs etc using commands. | 13,14 | 1,2,3 | 5,15 |
| **CO-2** | **Simulate** different topologies/network using Cisco Packet Tracer application. | 13,14 | 1,2,3 | 5,15 |
| **CO-3** | **Manage** IP addresses and troubleshooting | 13,14 | 1,2,3 | 5,15 |
| **CO-4** | **Manage** applications like Web, Email, DHCP, DNS and FTP. | 13,14 | 1,2,3 | 5,15 |
| **CO-5** | **Study** on Industry relevant tools to perform traffic management. | 13,14 | 1,2,3 | 5,15 |
| **CO-6** | **Write** C program using RFCs to implement standard protocol using TCP / IP. | 13,14 | 1,2,3 | 5,15 |

| **POs/PSOs** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** | **13** | **14** | **15** | **16** |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Mapping Level** | 2.0 | 2.0 | 2.0 | - | 1.0 | - | - | - | - | - | - | - | 3.0 | 3.0 | 1.0 | - |

**Pre-requisites:** Knoweldge of Data Communication and Computer Networks (registration).

**Contents:**

| **Sl. No.** | **Term Work** |
|---|---|
| **T1** | **Building a Simple Network:**<br>Part 1: Set Up the Network Topology (Ethernet only)<br>Part 2: Configure PC Hosts<br>Part 3: Configure and Verify Basic Switch Settings<br><br>**Learning Outcomes: [CO-1,2]** |
| **T2** | **Connecting a Wired and Wireless**<br>Part 1: Connect to the Cloud<br>Part 2: Connect Router0<br>Part 3: Connect Remaining Devices<br>Part 4: Verify Connections |

| | |
|---|---|
| | Part 5: Examine the Physical Topology<br>**Learning Outcomes: [CO-1,2]** |
| **T3** | **Troubleshooting IPv4 and IPv6 Addressing**<br>Part 1: Troubleshoot First Issue<br>Part 2: Troubleshoot Second Issue<br>Part 3: Troubleshoot Third Issue<br>**Learning Outcomes:[ CO-2,3]** |
| **T4** | **Configuring IPv6 Addresses on Network Devices**<br>Part 1: Set Up Topology and Configure Basic Router and Switch Settings<br>Part 2: Configure IPv6 Addresses Manually<br>Part 3: Verify End-to-End Connectivity<br>**Learning Outcomes:[ CO-2,3]** |
| **T5** | **Designing and Implementing a Subnetted IPv4 Addressing Scheme**<br>Part 1: Design a Network Subnetting Scheme<br>Part 2: Configure the Devices<br>Part 3: Test and Troubleshoot the Network<br>**Learning Outcomes:[ CO-1,2,3]** |
| **T6** | **Web and Email**<br>Part 1: Configure and Verify Web Services<br>Part 2: Configure and Verify Email Services<br>**Learning Outcomes:[ CO-3,4]** |
| **T7** | **DHCP and DNS Servers**<br>Part 1: Configure Static IPv4 Addressing<br>Part 2: Configure and Verify DNS Records<br>**Learning Outcomes:[ CO-3,4]** |
| **T8** | **FTP Servers**<br>Part 1: Configure FTP Services on Servers<br>Part 2: Upload a File to the FTP Server<br>Part 3: Download a File from the FTP Server<br>**Learning Outcomes:[ CO-3,4]** |
| **T9** | **Troubleshooting Connectivity Issues**<br>The objective of this Packet Tracer activity is to troubleshoot and resolve connectivity issues, if possible. Otherwise, the issues should be clearly documented and so they can be escalated.<br>**Learning Outcomes:[ CO-1,2,3,4]** |
| **T10** | **Study** on Industry relevant tools to perform traffic management.<br>**Learning Outcomes:[ CO-5]** |
| **T11** | **Write** C program using RFCs to implement standard protocol using TCP / IP.<br>**Learning Outcomes:[ CO-6]** |

| 21UCSL604 | Automata Theory & Compiler Design Lab | (0-0-2) 1 |
|---|---|---|

**Contact Hours: 24**

**Course Learning Objectives (CLOs):** This laboratory course focuses on representation of patterns and syntax using lexical rules and grammars respectively, Implementation of parser & translation schemes, Implementation of assemblers, loaders, linkers & macro processor, Knowledge of system level APIs for implementation of IPC and system commands.

**Course outcomes (COs):**

| **Description of the Course Outcome:** At the end of the course the student will be able to: | | **Mapping to POs(1-12)/ PSOs (13-16)** | | |
|---|---|---|---|---|
| | | **Substantial Level (3)** | **Moderate Level (2)** | **Slight Level (1)** |
| **CO-1** | **Prepare** the grammar for the given constructs and **Write** a program using compiler writing tools to implement lexical analyzer and parser. | 13,14 | 1,2,3 | 15 |
| **CO-2** | **Write** a program to **implement** a parser. | 13,14 | 1,2,3 | 15 |
| **CO-3** | **Write** a program to implement assembler functions. | 13,14 | 1,2,3 | 15 |
| **CO-4** | **Write** a program to **Implement** various UNIX commands using system calls. | 13,14 | 1,2,3 | 15 |
| **CO-5** | **Use** IPC concepts in implementing communication protocol. | 13,14 | 1,2,3 | 15 |

| POs/PSOs | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Mapping Level | 2.0 | 2.0 | 2.0 | - | - | - | - | - | - | - | - | - | 3.0 | 3.0 | 1.0 | - |

**Prerequisites:** Knowledge of: Unix Operating System, Any programming language, and Finite automata and formal Languages

**Contents:**

1. Programs on Lex and Yacc.
2. Implementation of parser.
3. Implementation of assembler.
4. Emulation of basic commands of UNIX using system calls.
5. Application development using Inter Process Communication.

| 21UCSL605 | Minor Project - 2 | (0-0-3) 1 |
|---|---|---|

**Contact Hours: 36**

**Course Learning Objectives (CLOs):** Though the specific objectives of this course depend on the Project chosen, below are the generic objectives of this course:

Understand the domain, Analyze through Modeling and Implementation through state of the art technology available. To know Software Engineering Principles: Modeling, Estimation, Design standards and architectural issues through use of Standards etc. Also, write modular programs and handle exceptions to provide reliable solutions, to test and verify the programs for different scenarios.

**Course Outcomes (COs):**

| **Description of the Course Outcome:** At the end of the course the student will be able to: | | **Mapping to POs(1-12)/ PSOs (13-16)** | | |
|---|---|---|---|---|
| | | **Substantial Level (3)** | **Moderate Level (2)** | **Slight Level (1)** |
| **CO-1** | **Identify** and **formulate** the problem. | 11,12 | 1,2,8 | 6,7 |
| **CO-2** | **Analyze** the problem scenario and **Design** the solutions to complex engineering problem using software engineering principles or appropriate research methodology. | 11,12, 13 | 2,3,5,8,16 | 6,7 |
| **CO-3** | **Identify** and **Implement** a feasible solution using appropriate technology, tools, procedures and techniques. | 11,12,14 | 3,4,5,8,16 | - |
| **CO-4** | **Verify** and **Validate** the proposed system for correctness and to demonstrate compliance with the design and hence the stated requirements/ research gap. | 11,12,15 | 5,8,16 | - |
| **CO-5** | **Prepare** the report and **communicate** effectively through presentation. | 10,11 | 8,9 | - |

| **POs/PSOs** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** | **13** | **14** | **15** | **16** |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Mapping Level** | 2.0 | 2.0 | 2.0 | 2.0 | 2.0 | 1.0 | 1.0 | 2.0 | 2.0 | 3.0 | 3.0 | 3.0 | 3.0 | 3.0 | 3.0 | 2.0 |

**Prerequisites:** Knowledge of

- Software Engineering concepts
- Any Programming Language

**Guidelines for the Conduction of Project Work:**

A project may belong to any of the following categories:

**1. Learning Centric:** Here the output of the project activity is enhancement of the student's knowledge. Here the student chooses a work with the intension to gain the knowledge.

**2. Application**: Traditional Software Engineering project, with appreciable complexity. Scenario of demonstration had to be made clear and Completeness is important.

**3. Product Base:** The final output is a novel one which may be the assembly of several existing subsystems. Completeness is important till the user manual.

**4. Research Oriented**: A research problem of student's interest. Achievement would be a publication in the IEEE/ACM international/national conferences.

**General Instructions to Students:**

- Students are expected to perform extensive literature survey, identify problem statements and prepare synopsis in consultation with project guide/supervisor. Students are expected to submit Synopsis- Initial (Registration Phase-1) approved by project guide, to the project coordinator as per the schedule notified. A copy is to be maintained with students and the guide. This registration/ Initial synopsis contains the description of the project concept created and acts as a base line for design and Implementation of the system.
- Notification/schedules and evaluation procedures will be sent to all students in the Google groups created in the department.
- Batch size is of maximum 4 students. Mixing of divisions is not permitted unless it is the sponsored/research project and request is initiated by project supervisor/guide.
- Students may approach the faculty members of CSE department for choosing them as project guide/supervisor.
- A teacher can guide/supervise maximum of 2 UG project groups. However in special case, a DUGC (Project Coordinating Team, **PCT**, working on the behalf of DUGC) may assign additional project groups to a given teacher.
- The department will financially support presentations of publication of paper only for reputed conference publications.
- DUGC/Project coordinating team (**PCT**) may be consulted for any other/ missing information.

**Evaluation Procedures:**

a) CIE Marks for the project is to be awarded by project guide/supervisor and SEE marks are to be awarded by examiners (appointed by DUGC) and PCT based on the overall guidelines and project classifications/rubrics by looking into Software Engineering aspects & usefulness w.r.t research/innovation/technology/industry trends through formal interactions and presentations. However, each team is expected to give a formal complete presentation of their work at the end of each phase (1 to 4) to project coordinating team (PCT).

b) PCT evaluates the work and suggest the corrections and observation. All project teams are expected to incorporate these changes in their work. These

observations will be made available to Project guides and SEE examiners, which will help them to evaluate and award marks during assessment process.

c) Project guides should keep track of all interactions they do with project team members on weekly basis.

d) All Projects are evaluated and individual students are awarded a grade based on the grading criteria set.

e) Individuals' grade/marks is decided based on both CIE and SEE marks/grade.

f) A project is considered for possible award of S grade; if and only if its 'research outputs' / 'product innovation outputs' results in to a publication of a paper. In a special case, innovative or sponsored business applications with focus on recent technological trends/ Industry trends catering for societal needs may be considered. If contents of paper/research output are not at satisfactory level, then, the assessment results in to appropriate lower grades.

g) A project is considered for possible award of A grade; if and only if it demonstrate product development skills in core system or systems level applications using all aspects of software engineering product development phases like: requirements, design, implementation, testing including standards like: use of design/architectural patterns, coding standards, use of tools for design/testing, programming practices, documentation and reporting etc..,. In a special case, innovative or sponsored business applications with focus on recent technological trends / Industry trends catering for societal needs may be considered.

h) If project work contribution/ output is not at satisfactory level, then the assessment results @ appropriate lower grades. All grades/marks are awarded based on individual contributions evaluated from software engineering perspectives **specified in the rubrics**.

i) Expected important features:

- Report preparation using Latex.
- Online plagiarism check report is to be enclosed in the report.
- Use of IEEE standard. Ex: reference listing and use of PPTs for presentation etc…
- Use of software tools. Ex: for Design, version control, UI design, Testing etc…
- Conduction of workshop/ training on technology/domain to students and Preparation of training material/manual (.doc & .pdf).
- Publication of paper based on outcome of the project.
- Submission of proposal to KSCST (Govt. of Karnataka) or other agencies for funding.
- Any other features suggested by guides/coordinators from time to time.

j) Marks Weightage and Various parameters for project evaluation for both CIE and SEE level @ 6th Semester.

k)

| Sl.No. | Parameter for Assessment | Marks (%) |
|---|---|---|
| **1** | Requirements Analysis (SRS): Abstract and Detailed. | 20 |
| **2** | Design Specification; Use of: UML diagrams, | 35 |

| | | |
|---|---|---|
| | architecture diagram, ER diagram, Patterns etc… Proper cohesiveness and coupling of various components in the system design. | |
| **4** | Use of Tools and standards. | 5 |
| **5** | Implementation: Code documentation, style, robustness, maintainability, Testability, Usability (User Experience) etc… | 10 |
| **6** | Testing: for every scenario of all use cases identified. | 10 |
| **7** | Final Oral Presentation (viva-voce) (IEEE Standards for slides, oral presentation techniques etc…) | 10 |
| **8** | Project Reports- Final and Intermediate if any: preparation using LATEX and plagiarism check | 10 |

**Note:**

1. Sufficient and completeness of each parameter is to be seen while awarding marks for individual students.
2. Marks for individual students in a given project team may vary based on individuals 'learning outcomes.

- All project teams are expected to participate in the project exhibition arranged at department level. Project teams are expected to share their project experience to all their juniors and motivate them to take-up challenging work as their project work. During project exhibition, Top 2 projects from the batch will be awarded with a certificate of appreciation at the end of academic year.

**21UHUL606 Soft Skills and Aptitude (0-0-2) 1**

**Contact Hours: 24**

**Course Learning Objectives (CLOs):**

This is included with the objectives of improving the communication skills, proficiency in English language and aptitude ability of the student to enhance the employability.

**Course Outcomes (COs):**

| **Description of the Course Outcome:** At the end of the course the student will be able to: | | **Mapping to POs(1-12) / PSO(13-16)** | | |
|---|---|---|---|---|
| | | **Substantial Level (3)** | **Moderate Level (2)** | **Slight Level (1)** |
| CO-1 | **Explain** the significance of communication in the profession. | - | 10 | - |
| CO-2 | **Use** the English language with proficiency | - | 10 | 12 |
| CO-3 | **Solve** Aptitude related problems | - | 9 | 12 |
| CO-4 | **Demonstrate** the competency in the placement activities. | - | 9 | - |

| POs | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Mapping Level | - | - | - | - | - | - | - | - | 2.0 | 2.0 | - | 1.0 | - | - | - | - |

**Contents:**

Training on communication skills, proficiency in English language and aptitude ability involving the internal and external resource.

**Evaluation:**

Both the internal and external resource persons shall be engaged in imparting the related knowledge and shall have only CIE as the evaluation component. There shall be one test conducted at the end for 25 marks in Aptitude testing and there shall be one presentation by the student for 25 marks or any other suitable testing components. The arrangement for CIE evaluation is to be done by the department and maintain the relevant documents.

## CIE and SEE Evaluation (from 2023-24 batch)

## Courses with LTP 3-0-0 and 4-0-0 or 2-2-0/3-2-0

### Continuous Internal Evaluation (CIE):

- Two Internal Assessments and one Improvement test each of 20 marks and one hour duration.
- Two higher scores from three tests are taken representing 40 marks.
- Question Paper pattern for Internal Assessment: 3 questions of 10 marks each with maximum of two sub divisions. Q.3 is compulsory and one question to be answered from Q.1 and Q.2.
- Course Teacher Assessment (CTA): Minimum two components such as quiz, seminar, written assignment, any technical activity related to course each of 5marks. Total CTA marks-10
- CIE=40 (from tests)+10(from CTA) =50 marks

### Semester End Examination (SEE):

- SEE is conducted for 100 marks with 3 hours duration. It is reduced to 50 marks.
- Question Paper pattern for SEE: Five units with built in choice. Each question with maximum of three sub divisions.
- Two questions are to be set from each unit with built in choice, for example Q1 or Q2 in unit –I, Q 3 or Q 4 in unit-II and so on.
- A total of 5 full questions to be answered choosing one full question from each unit. All five units are to be answered compulsorily.
- Each question is of 20 marks.
- The Question paper is to be set for duration of 3 hours both for 3 and 4 credits courses.
- The Question paper is to be set for 100 marks for 3 and 4 credits courses.