Shirpur Education Society's

# R. C. Patel Institute of Technology, Shirpur

(An Autonomous Institute)

## Course Structure and Syllabus

## Third Year B. Tech. (Computer Engineering)

With effect from Year 2025-26

Shahada Road, Near Nimzari Naka, Shirpur, Maharashtra 425405
Ph: 02563 259 802, Web: www.rcpit.ac.in

## Third Year B.Tech Computer Engineering Semester-V (w.e.f. 2025-26)

| Sr. No. | Course Category | Course Code | Course Title | Teaching Scheme | | | Evaluation Scheme | | | | | Total | Credits | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | Continuous Assessment (CA) | | | | | | | |
| | | | | | | | TA | Term Test 1 (TT1) | Term Test 2 (TT2) | Average of (TT1 & TT2) | ESE | | | |
| | | | | L | T | P | [A] | | | [B] | [C] | [A+B+C] | | |
| 1 | PC | RCP23CCPC501 | Data Warehousing and Mining | 3 | | | 25 | 15 | 15 | 15 | 60 | 100 | 3 | 4 |
| | PC | RCP23CLPC501 | Data Warehousing and Mining Laboratory | | | 2 | 25 | | | | 25 | 50 | 1 | |
| 2 | PC | RCP23CCPC502 | Artificial Intelligence | 3 | | | 25 | 15 | 15 | 15 | 60 | 100 | 3 | 4 |
| | PC | RCP23CLPC502 | Artificial Intelligence Laboratory | | | 2 | 25 | | | | 25 | 50 | 1 | |
| 3 | PC | RCP23CCPC503 | Automata Theory and Compiler Design | 3 | | | 25 | 15 | 15 | 15 | 60 | 100 | 3 | 4 |
| | PC | RCP23CLPC503 | Automata Theory and Compiler Design Laboratory | | | 2 | 25 | | | | 25 | 50 | 1 | |
| 4 | PC | RCP23CLPC504 | Web Programming Laboratory | | | 2 | 25 | | | | 25 | 50 | 1 | 1 |
| 5# | PE | RCP23CCPE511 | Advanced Algorithms | 3 | | | 25 | 15 | 15 | 15 | 60 | 100 | 3 | 4 |
| | | RCP23CLPE511 | Advanced Algorithms Laboratory | | | 2 | 25 | | | | 25 | 50 | 1 | |
| | | RCP23CCPE512 | Advanced Operating System | 3 | | | 25 | 15 | 15 | 15 | 60 | 100 | 3 | |
| | | RCP23CLPE512 | Advanced Operating System Laboratory | | | 2 | 25 | | | | 25 | 50 | 1 | |
| | | RCP23CCPE513 | Advanced Database Management System | 3 | | | 25 | 15 | 15 | 15 | 60 | 100 | 3 | |
| | | RCP23CLPE513 | Advanced Database Management System Laboratory | | | 2 | 25 | | | | 25 | 50 | 1 | |
| | | RCP23CCPE514 | Computer Graphics and Image Processing | 3 | | | 25 | 15 | 15 | 15 | 60 | 100 | 3 | |
| | | RCP23CLPE514 | Computer Graphics and Image Processing Laboratory | | | 2 | 25 | | | | 25 | 50 | 1 | |
| 6 | MD | RCP23CCMD501 | IoT-Centric Processor Organization and Architecture | 2 | | | 25 | 15 | 15 | 15 | 60 | 100 | 2 | 3 |
| | MD | RCP23CLMD501 | IoT-Centric Processor Organization and Architecture Laboratory | | | 2 | 25 | | | | 25 | 50 | 1 | |
| 7 | SC | RCP23IPSC501 | Semester Project - III | | | 2 | 25 | | | | 25 | 50 | 1 | |
| 8 | HS | RCP23ICHSX07 | Constitution of India | 1 | | | | | | | | | A | |
| | | | Total | 15 | | 14 | 300 | | | 75 | 475 | 850 | 21 | |

# Any 1 Programme Elective Course and A Audit Course.

Prepared by:
Ms. J. S. Sonawane

Checked by:
Dr. S. M. Pardeshi

Prof. Dr. R. B. Wagh
BOS Chairman

Prof. S. P. Shukla
C.O.E.

Prof. Dr. P. J. Deore
Dean Academics & Dy. Director

Prof. Dr. J. B. Patil
Director

R. C. Patel Institute of Technology, Shirpur
Autonomous

## Third Year B.Tech Computer Engineering Semester-VI (w.e.f. 2025-26)

| Sr. No. | Course Category | Course Code | Course Title | Teaching Scheme | | | Evaluation Scheme | | | | | Total | Credits | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | Continuous Assessment (CA) | | | | ESE | | | |
| | | | | L | T | P | TA | Term Test 1 (TT1) | Term Test 2 (TT2) | Average of (TT1 & TT2) | | | | |
| | | | | | | | [A] | | | [B] | [C] | [A+B+C] | | |
| 1 | PC | RCP23CCPC601 | Machine Learning | 3 | | | 25 | 15 | 15 | 15 | 60 | 100 | 3 | 4 |
| | PC | RCP23CLPC601 | Machine Learning Laboratory | | | 2 | 25 | | | | 25 | 50 | 1 | |
| 2 | PC | RCP23CCPC602 | Software Engineering | 3 | | | 25 | 15 | 15 | 15 | 60 | 100 | 3 | 4 |
| | PC | RCP23CLPC602 | Software Engineering Laboratory | | | 2 | 25 | | | | 25 | 50 | 1 | |
| 3 | PC | RCP23CLPC603 | DevOps Laboratory | | | 2 | 25 | | | | 25 | 50 | 1 | 1 |
| 4# | PE | RCP23CCPE611 | Big Data Analytics | 3 | | | 25 | 15 | 15 | 15 | 60 | 100 | 3 | 4 |
| | | RCP23CLPE611 | Big Data Analytics Laboratory | | | 2 | 25 | | | | 25 | 50 | 1 | |
| | | RCP23CCPE612 | Game Theory | 3 | | | 25 | 15 | 15 | 15 | 60 | 100 | 3 | |
| | | RCP23CLPE612 | Game Theory Laboratory | | | 2 | 25 | | | | 25 | 50 | 1 | |
| | | RCP23CCPE613 | Computer Vision | 3 | | | 25 | 15 | 15 | 15 | 60 | 100 | 3 | |
| | | RCP23CLPE613 | Computer Vision Laboratory | | | 2 | 25 | | | | 25 | 50 | 1 | |
| 5# | PE | RCP23CCPE621 | Advanced Network Design | 3 | | | 25 | 15 | 15 | 15 | 60 | 100 | 3 | 4 |
| | | RCP23CLPE621 | Advanced Network Design Laboratory | | | 2 | 25 | | | | 25 | 50 | 1 | |
| | | RCP23CCPE622 | Distributed Databases | 3 | | | 25 | 15 | 15 | 15 | 60 | 100 | 3 | |
| | | RCP23CLPE622 | Distributed Databases Laboratory | | | 2 | 25 | | | | 25 | 50 | 1 | |
| | | RCP23CCPE623 | High Performance Computing | 3 | | | 25 | 15 | 15 | 15 | 60 | 100 | 3 | |
| | | RCP23CLPE623 | High Performance Computing Laboratory | | | 2 | 25 | | | | 25 | 50 | 1 | |
| 6 | MD | RCP23CCMD601 | Security in IOT Ecosystem | 2 | | | 25 | 15 | 15 | 15 | 60 | 100 | 2 | 3 |
| | MD | RCP23CLMD601 | Security in IOT Ecosystem Laboratory | | | 2 | 25 | | | | 25 | 50 | 1 | |
| 7 | EL | RCP23IPEL601 | Project Stage - I | | | 4 | 50 | | | | 50 | 100 | 2 | |
| 8 | HS | RCP23ITHSX06 | Environmental Science Tutorial | | 1 | | 25 | | | | | 25 | 1 | |
| 9 | SC | RCP23CLSC601 | Employability Skill Development Program-I | | | 2 | 25 | | | | 25 | 50 | 1 | |
| | | | Total | 14 | 1 | 18 | 375 | | | 75 | 525 | 975 | 24 | |

# Any 1 Programme Elective Course.

Prepared by:
Ms. J. S. Sonawane

Checked by:
Dr. S. M. Pardeshi

Prof. Dr. R. B. Wagh
BOS Chairman

Prof. S. P. Shukla
C.O.E.

Prof. Dr. P. J. Deore
Dean Academics & Dy. Director

Prof. Dr. J. B. Patil
Director

R. C. Patel Institute of Technology, Shirpur
Autonomous

# Semester - V

| Program: Computer Engineering | T.Y. B.Tech. | Semester: V |
|---|---|---|
| Data Warehousing and Mining (RCP23CCPC501) | | |
| Data Warehousing and Mining Laboratory (RCP23CLPC501) | | |

## Prerequisites:

1. Basic database concepts
2. Concepts of algorithm design and analysis

## Course Objective(s):

1. To identify the scope and essentiality of Data Mining and Warehouse.
2. To analyze data, choose relevant models and algorithms for respective applications.
3. To develop research interest towards advances in data mining.

## Course Outcomes:

On completion of the course, the learner will be able to:

| CO | Course Outcomes | Blooms Level | Blooms Description |
|---|---|---|---|
| CO1 | Design data warehouse models using dimension-modeling techniques. | L3 | Apply |
| CO2 | Analyse the data by applying Online Analytical Processing (OLAP) operations for strategic decisions. | L3 | Apply |
| CO3 | Apply preprocessing techniques to the given raw data. | L3 | Apply |
| CO4 | Apply appropriate data mining techniques on data sets to retrieve relevant information. | L3 | Apply |

# Data Warehousing and Mining (RCP23CCPC501) Course Contents

## Unit-I — 08 Hrs.

**Introduction to Data Warehouse and Dimensional modelling:** Introduction to Strategic Information, Need for Strategic Information, Features of Data Warehouse, Data warehouse versus Data Marts, Data warehouse versus Data Lake, Top-down versus Bottom-up approach. Data warehouse architecture, metadata, E-R modelling versus Dimensional Modelling, Information Package Diagram, STAR schema, STAR schema keys, Snowflake Schema, Fact Constellation Schema, Factless Fact tables, Update to the dimension tables, Aggregate fact tables.

## Unit-II — 06 Hrs.

**ETL Process and OLAP:** Major steps in ETL process, Data extraction: Techniques, Data transformation: Basic tasks, Major transformation types, Data Loading: Applying Data, OLTP Vs OLAP, OLAP definition, Dimensional Analysis, Hypercubes, OLAP operations: Drill down, Roll up, Slice, Dice and Rotation, OLAP models: MOLAP, ROLAP.

## Unit-III — 06 Hrs.

**Introduction to Data Mining, Data Exploration and Preprocessing:** Data Mining Task and Techniques, KDD process, Issues in Data Mining, Applications of Data Mining, Data Exploration: Types of Attributes, Statistical Description of Data, Data Visualization, Measuring data similarity and dissimilarity.

Data Preprocessing: Major tasks in preprocessing, Data Cleaning: Missing values, Noisy data; Data Integration: Entity Identification Problem, Redundancy and Correlation Analysis, Tuple Duplication, Data Value Conflict Detection and Resolution; Data Reduction: Attribute subset selection, Histograms, Clustering and Sampling; Data Transformation & Data Discretization: Data Transformation by Normalization, Discretization by Binning, Discretization by Histogram Analysis, Concept hierarchy generation for Nominal data.

## Unit-IV — 08 Hrs.

**Classification and Clustering:**

**Classification:** Basic Concepts of classification, Decision Tree Induction, Attribute Selection Measures using Information Gain, Tree pruning

Bayes Classification Methods: Bayes' Theorem, Naïve Bayesian Classification Model Evaluation & Selection: Metrics for Evaluating Classifier Performance, Holdout Method and Random Subsampling, Cross Validation, Bootstrap, Improving Classification Accuracy: Ensemble classification, Bagging,

Boosting and AdaBoost, Random Forests,

**Clustering:** Cluster Analysis and Requirements of Cluster Analysis

Partitioning Methods: k-Means, k-Medoids

Hierarchical Methods: Agglomerative, Divisive

Evaluation of Clustering: Assessing Clustering Tendency, Determining Number of Clusters and Measuring cluster quality: Intrinsic and Extrinsic methods

## Unit-V 05 Hrs.

**Mining Frequent Patterns and Association Rules:** Market Basket Analysis, Frequent Item sets, Closed Item sets, and Association Rule.

Frequent Item set Mining Methods: Apriori Algorithm, Association Rule Generation, FP growth, Mining frequent Itemsets using Vertical Data Format.

## Unit-VI 06 Hrs.

**Spatial and Web Mining:** Spatial Data, Spatial Vs. Classical Data Mining, Spatial Data Structures, Mining Spatial Association and Co-location Patterns, Spatial Clustering Techniques: CLARANS Extension.

**Web Mining:** Web Content Mining, Web Structure Mining, Web Usage mining, Applications of Web Mining.

## Data Warehousing and Mining Laboratory (RCP23CLPC501)

### List of Laboratory Experiments

**Suggested Experiments:(Any 08)**

1. Build Data Warehouse/Data Mart for a given problem statement
   (a) Identifying the source tables and populating sample data
   (b) Making information package diagram
   (c) Design dimensional data model i.e. Star schema, Snowflake schema and Fact Constellation schema (if applicable)
2. Perform data Pre-processing task on your dataset
3. Perform various OLAP operations such as: Slice, Dice, Drilldown, Rollup, Pivot
4. Implementation of Classification algorithm
   (a) Using Decision Tree ID3
   (b) Naïve Bayes algorithm

5. Implementation of Clustering algorithm

   (a) K-means

   (b) Hierarchical clustering (single/complete/average)

   (c) DBScan

6. Implementation of Association Rule Mining algorithm

   (a) Apriori algorithm

   (b) FP Tree algorithm

7. Demonstrate performing Classification, Clustering, Association algorithm on datasets using data mining tools (WEKA, R tool, XL Miner, etc.)

8. Implementation of PageRank algorithm

9. Implementation of HITS algorithm

10. Implementation of Spatial Clustering Algorithm — CLARANS Extensions

11. Case Study on Recent Data Mining Applications

Any other experiment based on syllabus may be included which would help the learner to understand topic/concept.

**Oral examination will be based on the entire syllabus including, the practicals performed during laboratory sessions.**

## Text Books:


1. Paulraj Ponniah, "Data Warehousing: Fundamentals for IT Professionals", Wiley India, 2nd Edition, 2013.

2. Jiawei Han and Micheline Kamber, "Data Mining: Concepts and Techniques", Morgan Kaufmann, 3rd Edition, 2012.

3. H. Dunham, "Data Mining: Introductory and Advanced Topics", Pearson Education, 1st Edition, 2006.


## Reference Books:


1. Theraja Reema, "Data Warehousing", Oxford University Press, 1st Edition, 2009.

2. P. N. Tan, M. Steinbach, and Vipin Kumar, "Introduction to Data Mining", Pearson Education 2nd Edition, , 2018.

| Program: Computer Engineering | T.Y. B.Tech. | Semester: V |
|---|---|---|
| Artificial Intelligence (RCP23CCPC502) | | |
| Artificial Intelligence Laboratory (RCP23CLPC502) | | |

## Prerequisites: Knowledge of

1. Programming Language
2. Algorithms

## Course Objective(s):

1. To provide the basic ideas and techniques underlying the design of intelligent systems.
2. To Impart the knowledge of various search techniques for problem solving.
3. Learn knowledge representation, planning and provide the knowledge to deal with uncertain and incomplete information

## Course Outcomes:

On completion of the course, the learner will be able to:

| CO | Course Outcomes | Blooms Level | Blooms Description |
|---|---|---|---|
| CO1 | Provide the basic ideas and techniques underlying the design of intelligent systems. | L2 | Understand |
| CO2 | Apply the knowledge of various search techniques for problem solving. | L3 | Apply |
| CO3 | Apply knowledge representation and planning techniques, and use suitable approaches to manage uncertain and incomplete information in AI systems. | L3 | Apply |
| CO4 | Explain the fundamental concepts of different learning paradigms and explore advanced AI techniques. | L4 | Analyze |

# Artificial Intelligence (RCP23CCPC502)
# Course Contents

## Unit-I — 05 Hrs.

**Introduction to Artificial Intelligence:** Introduction, History of Artificial Intelligence, Intelligent Systems: Categorization of Intelligent System, Components of AI Program, Foundations of AI.

**Agents and Environments:** Concept of rationality, Nature of environment, Structure of Agents, Types of Agents, PEAS representation for an Agent.

## Unit-II — 12 Hrs.

**Problem Solving:** Problem Solving Agent, Formulating Problems, Example Problems.

**Search Methods: Uninformed Search:** Breadth First Search (BFS), Depth First Search (DFS), Depth Limited Search, Depth First Iterative Deepening (DFID).

**Informed Search:**Greedy Best First Search, A* Search, Memory bounded heuristic Search.

**Local Search Algorithms and Optimization Problems:** Hill Climbing search, Simulated Annealing, Genetic algorithms, Ant Colony Optimization.

**Adversarial Search:** Game Playing, The Minimax algorithm, Alpha-Beta Pruning.

## Unit-III — 07 Hrs.

**Knowledge Representation and Reasoning:** Knowledge based Agents,The Wumpus World, Propositional Logic, First Order Logic, Inference in FOL, Conjunctive Normal Form, Forward Chaining, Backward Chaining, Unification, Resolution, Logic Programming (PROLOG),Semantic networks.

**Uncertain Knowledge and Reasoning:** Representing knowledge in an uncertain domain, The semantics of Bayesian Belief Network, Inference in Belief Network.

## Unit-IV — 04 Hrs.

**Planning:** The planning problem, Planning with state space search, Planning graphs, Partial order planning, Hierarchical planning.

## Unit-V — 07 Hrs.

**Learning:** Types of Learning, Inductive Learning.

**Artificial Neural Networks:** McCulloh Pitts Model, Perceptron, Feed Forward Network, Backpropagation Algorithm, Self-Organizing Map.

## Unit-VI — 04 Hrs.

**Expert System:** Introduction, Phases in building Expert Systems, ES Architecture, Case Study on MYCIN Rule based system.

**Advanced topic:** Introduction to Generative AI, Explainable AI.

## Artificial Intelligence Laboratory (RCP23CLPC502)

### List of Laboratory Experiments

**Suggested Experiments:(Any 10)**

1. Select a problem statement relevant to AI.
   i) Identify the problem ii) PEAS Description iii) Problem formulation
2. Identify and analyze Uninformed Search Algorithm to solve the problem. Implement BFS/DFS search algorithms to reach goal state.
3. Implement DFID search algorithms to reach goal state.
4. Identify and analyze Informed Search Algorithm to solve the problem. Implement A* search algorithm to reach goal state
5. Program to implement Local Search algorithm: Hill Climbing search
6. Program on Genetic Algorithm to solve a optimization problem in AI.
7. Program to implement learning: Perceptron Learning / Backpropagation Algorithm.
8. The laboratory will emphasize the use of PROLOG.
   (For example, Program to implement Family Tree in Prolog)
9. Implementation on any AI game: Wumpus world, Tic-tac-toe, 8-Queens Problem
10. Case study of an AI Application.

Any other experiment based on syllabus may be included which would help the learner to understand topic/concept.

**Oral examination will be based on the entire syllabus including, the practicals performed during laboratory sessions.**

## Text Books:


1. Stuart J. Russell and Peter Norvig, "Artificial Intelligence A Modern Approach", 3rd Edition, Pearson Education, 2010.
2. George F. Luger, "Artificial Intelligence", 6th Edition, Pearson Education, 2021.
3. Deepak Khemani, "A First Course in Artificial Intelligence", 6th Reprint, McGraw Hill Education, 2018f.

4. Saroj Kaushik, "Artificial Intelligence", 1st Edition, Cengage Learning, 2010.

## Reference Books:

1. A.P. Engelbrecht, "Computational Intelligence", 3rd Edition, Wiley–Blackwell, 2021.

2. Elaine Rich and Kevin Knight, "Artificial Intelligence", 3rd Edition, McGraw Hill Education, 2017.

3. Ivan Bratko, "ROLOG Programming for Artificial Intelligence", 4th Edition, Addison-Wesley 2011.

4. Hagan, Demuth, Beale, "Neural Network Design", 2nd Edition, Martin Hagan 2014.

5. Ronald J. Martin, "The age of Artificial intelligence", Independently published 2023.

| Program: Computer Engineering | T.Y. B.Tech. | Semester: V |
|---|---|---|
| Automata Theory and Compiler Design (RCP23CCPC503) | | |
| Automata Theory and Compiler Design Laboratory (RCP23CLPC503) | | |

**Prerequisites:** Programming, Algorithms and Data Structures.

## Course Objective(s):

1. To introduce the fundamental concepts of formal languages, grammar and automata theory.
2. To introduce the major concepts of language translation and compiler design and impart the knowledge of practical skills necessary for constructing a compiler.

## Course Outcomes:

On completion of the course, the learner will be able to:

| CO | Course Outcomes | Blooms Level | Blooms Description |
|---|---|---|---|
| CO1 | Understand the basic properties of formal languages and differentiate between them. | L2 | Understand |
| CO2 | Build grammar to produce strings from a specific language and apply different parsing algorithms. | L6 | Create |
| CO3 | Acquire concepts related to computational models including decidability and intractability. | L2 | Understand |
| CO4 | Understand the basics of analysis phase of compilation and its relationship with automata. | L2 | Understand |
| CO5 | Understand synthesis phases of compilation used to construct target language statements. | L2 | Understand |

# Automata Theory and Compiler Design (RCP23CCPC503) Course Contents

## Unit-I — 08 Hrs.

**Introduction to Automata Theory:** Central Concepts of Automata-theory, Deterministic Finite Automata (DFA), Non- Determinstic Finite Automata (NFA), Epsilon- NFA, NFA to DFA Conversion, Minimization of DFA.

**Introduction to Compiler Design:** Language Processors, Phases of Compilers.

## Unit-II — 06 Hrs.

**Regular Expressions and Languages:** Regular Expressions, Finite Automata and Regular Expressions, Proving Languages Not to Be Regular.

**Lexical Analysis Phase of compiler Design:** Role of Lexical Analyzer, Input Buffering, Specification of Token, Recognition of Token.

## Unit-III — 08 Hrs.

**Context Free Grammars:** Definition and designing CFGs, Derivations Using a Grammar, Parse Trees, Ambiguity and Elimination of Ambiguity, Elimination of Left Recursion, Left Factoring.

**Syntax Analysis Phase of Compilers:** Role of Parser, Top-Down Parsing, First and Follow of a Variable, Bottom-up Parsing, Introduction to LR Parsing.

## Unit-IV — 05 Hrs.

**Push Down Automata:** Definition of the Pushdown Automata, The Languages of a PDA.

**Syntax Directed Translation:** Introduction to Syntax-Directed Definitions

## Unit-V — 06 Hrs.

**Introduction to Turing Machine:** Problems that Computers Cannot Solve, The Turing machine, problems, Programming Techniques for Turing Machine, Extensions to the Basic Turing Machine.

**Undecidability:** A language That Is Not Recursively Enumerable, An Undecidable Problem That Is RE.

## Unit-VI — 06 Hrs.

**Introduction to Synthesis Phase of Compilers:**

**Intermediate Code Generation:** Variants of Syntax Trees, Three-Address Code.

**Code optimization:** Scope of optimization, DAG representation, Basic blocks, Machine-dependent and Machine independent optimization.

**Code Generation:** Issues in the Design of a Code Generator

## Automata Theory and Compiler Design Laboratory (RCP23CLPC503)

### List of Laboratory Experiments

**Suggested Experiments:(Any 08)**

1. Write a program to find epsilon closure of all states of any given NFA with epsilon transitions.
2. Design and implement a lexical analyser for given language using C and the lexical analyser should ignore redundant spaces, tabs and new line.
3. Write a program to convert CFG to CNF.
4. Write a program to implement Left recursion and left factoring.
5. Write a program to find First and Follow of a Variable.
6. To create and simulate automata using JFLAP.
7. Any real-life application of automata theory.
8. Case Study: LLVM

Any other experiment based on syllabus may be included which would help the learner to understand topic/concept.

**Oral examination will be based on the entire syllabus including, the practicals performed during laboratory sessions.**

## Text Books:


1. John E Hopcroft, Rajeev Motwani, Jeffrey D. Ullman, "Introduction to Automata Theory, Languages and Computation", 3rd Edition, Pearson, 2008.
2. Alfred V.Aho, Monica S.Lam,Ravi Sethi, Jeffrey D. Ullman, " Compilers Principles, Techniques and Tools", 2nd Edition, Pearson, 2014.


## Reference Books:


1. Elain Rich, "Automata, Computability and complexity", 1st Edition, Pearson Education, 2018.
2. K.L.P Mishra, N Chandrashekaran, 'Theory of Computer Science", 3rd Edition, PHI, 2012.
3. Peter Linz, "An introduction to Formal Languages and Automata", 6th Edition, Narosa Publishers, 2017.
4. K Muneeswaran, "Compiler Design", Oxford University Press, 2013.

| Program: Computer Engineering | T.Y. B.Tech. | Semester: V |
|---|---|---|
| Web Programming Laboratory (RCP23CLPC504) | | |

**Prerequisite:** Basics of Programming.

## Course Objectives:

1. To get familiar with the basics of Web Programming.
2. To expose students to Basics and Advanced concepts in REACT.
3. To orient students to Fundamentals of node.js and express framework.
4. To understand REST API and MongoDB for Frontend and Backend Connectivity.

## Course Outcomes:

On completion of the course, the learner will be able to:

| CO | Course Outcomes | Blooms Level | Blooms Description |
|---|---|---|---|
| CO1 | Implement interactive web page(s) using HTML5, CSS3 and JavaScript. | L6 | Create |
| CO2 | Implement Single Page Application using React.js and Node.js Framework. | L6 | Create |
| CO3 | Construct web based Node.js applications using Express. | L6 | Create |
| CO4 | Apply MongoDB for frontend and backend connectivity using REST API. | L6 | Create |

# Web Programming Laboratory(RCP23CLPC504) Course Contents

## Unit-I HTML5, CSS3 and Bootstrap5 — 04 Hrs.

**HTML5**: Introduction and Advantages of HTML5, HTML formatting, Hyperlinks, Images, tables, Lists, Elements (Block & Inline), Attributes, Page Layout, Semantic Elements, HTML5 Web Forms, HTML5 Media (Video & Audio).

**CSS3**: Introducing CSS3, Selectors, Border, Box Model, Margin & Padding, Background Images & Colors and Other Decorative (Texts, Fonts, Links, Lists, Tables), Positioning, Combinators, Pseudo-class and Pseudo-element, CSS Attribute Selectors, 2D and 3D Transformations, Transitions and Animations, @property, Flexbox, CSS3-Multi Column Layout, Media Queries.

**Bootstrap5**: Introduction to Bootstrap, Containers, Bootstrap Grids, Bootstrap Cards, Bootstrap JS (Navbar, Offcanvas, Collapse, Modal, Carousel), Flex, Bootstrap Forms.

## Unit-II JavaScript — 06 Hrs.

Introduction to JavaScript, JavaScript DOM Model, var, let, const, operators, primitive data types & strings, conditional, loop, for-each loop, operators, ternary operators, RegExp. Arrow functions, normal functions, Lexical this, Events, Handling events - Spread operator, Destructuring - named imports, default import, map, filter, reduce, date and objects. Call back system, Asynchronous, promises - Async, await, JSON Introduction, Syntax

## Unit-III React Fundamental: — 06 Hrs.

Introduction to Vue.js vs. Angular vs. React, Installation, installing libraries, Folder and file structure, Components, Component lifecycle, Props, State, Events, React Conditional, map, keys, React Router and Single page applications, Forms, Form Handling. Refs, Use effects, Hooks, Flux.

## Unit-IV Node.js — 04 Hrs.

**Node.js, Setup Development Environment:** Installation of Node.js, Working in REPL, Node JS Console, apply concepts like Datatype, Objects, Methods, Object-Oriented Programming, and Classes in the context of backend development, Creating simple Node Server, Request and Response, Routing responses, NPM JavaScript Build Processes, Event Loop and Emitters, File System Interaction, Modules, Native Node drivers.

## Unit-V Express.js: — 04 Hrs.

Introduction, Installation, Express router, REST API, Generator, Authentication, session, [illegible]ing with React, Commercial deployment

## Unit-VI Database Connectivity: 02 Hrs.

MongoDB Installation, connecting to MongoDB, CRUD Operations, Frontend Integration with React, User Authentication (JWT), Role-based Access Control, connecting MongoDB to Node-RED. Hosting Backend (e.g., Heroku, Vercel), Hosting Frontend (e.g., Netlify, Vercel).

## List of Laboratory Experiments

**Suggested Experiments:(Any 08)**

1. Develop an informative page using HTML5 layout tags with sections, images, links, and formatting.
2. Create a form in HTML5 with validations (Email, Mobile, Pin code, Password) using JavaScript.
3. Apply CSS properties: borders, margins, padding, navigation, dropdown lists.
4. Create a counter application in JavaScript.
5. Create a React application to demonstrate JSX, Components, Props, and State.
6. Create a React application to demonstrate Forms, Events, Routers, Refs, and Keys.
7. Create a React application to demonstrate Conditional Rendering.
8. Build a simple web server using HTTP module in Node.js.
9. Demonstrate Callback system, Promises, Async/Await in Node.js.
10. Demonstrate connection of Node-RED with MongoDB.
11. Build a RESTful API using MongoDB.
12. Demonstrate React Hooks: Immutability Revisited & Refs.
13. Demonstrate useEffect Hook and Side Effects.
14. Demonstrate Custom Hooks & Data Fetching.
15. Demonstrate Memoization in React Hook.
16. Project: Interactive Forms with Dynamic Tabs.
17. Job Application Form.
18. Set up file structure with basic tabs and placeholder content.

19. Add second tab to enhance tab interface.

20. Display tab headers and conditional content.

21. Implement tab navigation.

22. Implement direct tab navigation via headers.

23. Populate Personal Info tab.

24. Populate Experience and Review & Submit tabs.

25. Create states and handlers.

26. Finalize form submission and display success message.

27. Implement smarter state management.

28. Add validation infrastructure and on-blur feedback.

29. Implement navigation guards and final submission logic.

30. Implement component API design.

31. Implement the spectrum of components.

32. Task: E-Commerce product listing components.

33. Import React from 'react'.

34. Demonstrate prop conflicts.

35. Create a React app with user greeting modal.

36. Implement forward-ref in React.

37. Implement polymorphism and compound components.

38. Implement polymorphic components.

39. Task on polymorphic components.

40. Implement compound components.

41. Demonstrate React context.

42. Demonstrate React context to avoid prop drilling.

43. Task: Implement theme switching and user greeting.

44. Understand modals in React.

45. Create a React app with user greeting modal.

46. Build a custom FAQ accordion.

47. Portfolio project.

48. Create empty files as building blocks.

49. Implement header component.

50. Upgrade the existing hero component in React portfolio.

51. Create About Me, Projects, Blog, Contact, and Footer sections.

52. Project: 2048 Game.

53. Set up and display the initial game board.

54. Add random tiles (2s or 4s) to empty spots.

55. Handle player input.

56. Compress tiles to the left.

57. Implement move right functionality.

58. Implement move up and down functionality.

59. Implement tile merging.

60. Implement game over logic.

61. Add a restart game button.

62. Implement advanced patterns and smarter state in React.

63. Implement React state and visual updates.

64. Re-trigger CSS animations in React.

65. Implement React counter app.

66. Implement smarter component design in React.

67. Bypass props with lifting content up.

68. Task: Refactor a card component using composition.

69. Demonstrate controlled vs uncontrolled components.

70. Manage complex state using useReducer.

71. Organize your reducer with switch-case.

72. Implement advanced UI and safe state updates.

73. Implement React state updates with Immer.

74. Task: Build a simple profile editor.

75. Demonstrate HeadlessUI portals.

76. Implement portal-based toast notification.

**Mini Project:** Develop a website using the MERN stack with homepage and at least three validated forms using HTML5, CSS/Bootstrap, JavaScript, and React.js.

**Practical examination will be based on the entire syllabus including, the practicals performed during laboratory sessions.**

## Text Books:


1. John Dean, "Web Programming with HTML5, CSS3 and JavaScript", Jones & Bartlett Learning, 1st Edition, 2018.

2. lenn Johnson, "Programming in HTML5 with JavaScript and CSS3", Microsoft Press, 1st Edition, 2013.

3. Adam Bretz and Colin J. Ihrig, "Full Stack JavaScript Development with MEAN", SitePoint Pty. Ltd., 1st Edition, 2014.

4. Simon Holmes Clive Harber, "Getting MEAN with Mongo, Express, Angular, and Node", Manning Publications, 2nd Edition, 2019.

5. Venkat Subramaniam, "Rediscovering JavaScript, Master ES6, ES7, and ES8", The Pragmatic Bookshelf, 1st Edition, 2018.

6. Alex Banks and Eve Porcello, "Learning React Functional Web Development with React and Redux", O'Reilly, 1st Edition, 2017.

7. Andrew Mead, "Learning Node.js Development", Packt Publishing, 1st Edition, 2018.

8. Valentin Bojinov, "RESTful Web API Design with Node.js 10", Packt Publication, 3rd Edition, 2018.

## Reference Books:


1. Ethan Brown, "Web Development with Node and Express", O'Reilly, 2nd Edition, 2019.
2. Shama Hoque "Full-Stack React Projects: Learn MERN stack development by building modern web apps using MongoDB, Express, React, and Node.js", Packt Publication, 2nd Edition, 2020.


## Online Resources:


1. https://www.w3schools.com/
2. https://www.tutorialspoint.com/index.htm
3. https://www.udemy.com/course/crash-course-html-and-css/
4. https://www.udemy.com/course/reactjs-training/
5. https://www.udemy.com/course/mern-stack-course-mongodb-express-react-and-nodejs/
6. https://www.classcentral.com/course/skillshare-create-a-web-app-with-react-mongodb-express-and-nodejs-84146




| Program: Computer Engineering | T.Y. B.Tech. | Semester: V |
|---|---|---|
| Advanced Algorithms(RCP23CCPE511) | | |
| Advanced Algorithms Laboratory (RCP23CLPE511) | | |

**Prerequisites:** Data structures, Analysis of Algorithms.

**Course Objective:** To provide conceptual and practical knowledge of Advanced Algorithms.

## Course Outcomes:

On completion of the course, the learner will be able to:

| COs | Course Outcomes | Blooms Level | Blooms Description |
|---|---|---|---|
| CO1 | Analyze the chosen algorithm. | L4 | Analyze |
| CO2 | Choose appropriate data structure and algorithm for given problem statement. | L3 | Apply |
| CO3 | Design the algorithm. | L6 | Create |
| CO4 | Classify the algorithms based on P NP and NP hard parameters. | L2 | Understand |

# Advanced Algorithms(RCP23CCPE511)
# Course Contents

## Unit-I Analysis of Algorithm Based on Time — 04 Hrs.

**Asymptotic notations:** Omega, Theta, Big-O, Small-o, small Omega and Tilde.

**Amortized Analysis:** Aggregate Method, Accounting Method, Potential Method, RAM model analysis of algorithm.

## Unit-II Probabilistic and Randomized Algorithm — 05 Hrs.

Probabilistic approach to algorithm and Randomized Analysis, Indicator Random Variable (IRV), Randomized Quick Sort, Analysis of Hiring Problem, Las Vegas and Monte Carlo algorithm.

## Unit-III Advanced Data Structures — 11 Hrs.

**Balanced Search Trees:** Red-Black Tree, Randomized BST

**Heap and Operations:** Binomial Tree, Binomial Heap, Treap.

**Spatial Data Structure:** KD Tree, R Tree.

**Probabilistic Data Structure:** LogLog and HyperLogLog, Count Min sketch.

## Unit-IV Graph Based Algorithms — 07 Hrs.

**Flow Network Introduction:** Residual Network, Augmenting Path, Ford-Fulkerson Method, Edmonds-Karp Method, Push-Relable Algorithm, Relable to Front algorithm.

**Bipartite Matching:** Maximum Bipartite Matching.

## Unit-V — 06 Hrs.

**Computational Geometry:** Line Segment Properties, Convex Hull-Graham's scan algorithm.

**Geometric Searching:** Point Location in polygon using Ray Crossing.

**Online Algorithms:** Competitive Ratio, K-Server.

**Special topic:** Cuckoo search algorithm.

## Unit-VI — 06 Hrs.

**Algorithm Classes:** P, NP, NP Hardness and NP Completeness

**NP Completeness Proofs:** Satisfiability (3 sat), Reducibility, TSP.

**Approximation Algorithms:** Vertex Cover Problem, Travelling Sales Person problem.

**Network Approximation:** Randomized Rounding, Primal Dual algorithms.

## Advanced Algorithms Laboratory (RCP23CLPE511)

### List of Laboratory Experiments

**Suggested Experiments:(Any 10)**

1. To perform Amortized Analysis.
2. To implement Randomized Algorithms (Randomized Quick Sort).
3. To implement Randomized Algorithms (Hiring Problem).
4. To implement Advanced Data Structure (Red-black Tree Operations).
5. To implement Advanced Data Structure (Binomial Tree Operations).
6. To implement Advanced Data Structure (R Tree Operations).
7. To implement Advanced Data Structure (KD Tree Operations).
8. To implement Advanced Data Structure (MinHash implementation).
9. To implement Graph Based Algorithms (Ford Fulkerson Method).
10. To implement Graph Based Algorithms (Push Relable Method).
11. To implement Graph Based Algorithms (Maximum Bipartite Matching).
12. To implement Computational Geometry (Graham Scan Algorithm).
13. To implement Online Algorithms (K-Server algorithm).
14. To implement Approximation Algorithm (Approximate TSP implementation).

Any other experiment based on syllabus may be included which would help the learner to understand topic/concept.

**Oral examination will be based on the entire syllabus including, the practicals performed during laboratory sessions.**

## Text Books:


1. Thomas H. Cormen, Charles E. Leiserson, Ronald L Rivest, Clifford Stein, "Introduction to Algorithms", 4th Edition,The MIT Press, 2022 .
2. S. Sridhar, "Design and analysis of algorithms", 2nd Edition, Oxford University Press, 2014
3. Horowitz, Sahani and Rajsekaran, "Fundamentals of Computer Algorithms", 2nd Edition, University Press, 2008.
4. Harsh Bhasin, "Algorithms Design and Analysis", 1st Edition, Oxford University Press, 2015

## Reference Books:


1. Rajeev Motwani, Prabhakar Raghavan, "Randomized Algorithm", 1st Edition, Cambridge University,1995.

2. S. K. Basu, "Design Methods and Analysis of Algorithm", 2nd Edition, PHI,2013.

3. Vijay V. Vajirani, "Approximation Algorithms", 1st Edition, Springer,2003.

4. Sanjeev Arora, Boaz Barak,"Computational Complexity", Princeton University, 2007.

| Program: Computer Engineering | T.Y. B.Tech. | Semester: V |
|---|---|---|
| Advanced Operating Systems (RCP23CCPE512) | | |
| Advanced Operating Systems Laboratory (RCP23CLPE512) | | |

**Prerequisites:** Operating System and Computer Organization.

## Course Objective(s):

1. To understand the difference between distributed, multiprocessor and virtualization concepts.
2. To explore Real time operating system concepts.
3. To explore mobile operating systems.

## Course Outcomes:

On completion of the course, the learner will be able to:

| CO | Course Outcomes | Blooms Level | Blooms Description |
|---|---|---|---|
| CO1 | Understand different types of operating systems concepts to solve real life problems. | L2 | Understand |
| CO2 | Analyze system performance by applying virtualization concepts. | L4 | Analyze |
| CO3 | Understand mobile operating systems concept. | L2 | Understand |

# Advanced Operating Systems (RCP23CCPE512) Course Contents

## Unit-I — 04 Hrs.

**Introduction:** Functions of operating systems, design approaches: layered, kernel based and virtual machine approach, need for advanced operating systems, types of advanced operating systems (NOS, DOS, Multiprocessor OS, Mobile OS, RTOS, Cloud OS).

## Unit-II — 09 Hrs.

**Distributed operating Systems:** Architecture of distributed operating systems, system architecture types, issues in distributed operating systems, inherent limitation of distribute systems. Distributed mutual exclusion: classification of mutual exclusion algorithms, Lamport's, token- based algorithm, Suzuki-Kasami's Broadcast algorithm, Raymond's Tree based algorithm Distributed deadlock detection, Distributed file systems.

## Unit-III — 09 Hrs.

**Real Time Operating Systems:** Basic Model of Real time systems, Characteristics, Applications of Real time systems, Real time task scheduling, Types of tasks and their characteristics. Task Scheduling, Clock driven Scheduling, Hybrid Schedulers, Event driven Scheduling, EDF Scheduling, Rate Monotonic Algorithm, handling resource sharing Resource Handling: Resource Sharing, Priority Inversion, PIP, PCP, HLP, Scheduling real time tasks in distributed systems.

## Unit-IV — 06 Hrs.

**Multiprocessor Operating Systems:** Introduction, Basic multiprocessor system architectures, design issues, Threads Process synchronization: the test and set instruction, the swap instruction, implementation of the process waits. Processor scheduling: Issues, Co- scheduling, Smart scheduling, Affinity Based scheduling.

## Unit-V — 06 Hrs.

**Virtualization:** Introduction to Virtualization, Types of Virtualizations, Bare Metal (XEN), Hosted (KVM) Virtualization, Para virtualization, Full virtualization, Emulation, Server Virtualization, Network Virtualization and Storage Virtualization.

## Unit-VI — 05 Hrs.

**Mobile Operating System:** Mobile OS: Architecture, Android OS, iOS, Virtual OS, Cloud OS and their design issues, Application development using Android.

## Advanced Operating Systems Laboratory (RCP23CLPE512)

### List of Laboratory Experiments

**Suggested Experiments:(Any 08)**

1. Implement concurrent client-server application.
2. Simulate Lamport's logical clock.
3. Implement Ricart-Aggarwala Algorithm.
4. Demonstrate deadlock detection using Edge Chasing algorithm.
5. Demonstrate hosted virtualization using KVM.
6. Load a new operating system virtually on the client machine using the concept of bare metal virtualization by XEN.
7. Hello world, linking activities, passing data.
8. Create a simple list view with image and text.
9. Integrate a website inside an application, use of SQLite
10. Application development using Android.

Any other experiment based on syllabus may be included which would help the learner to understand topic/concept.

**Oral examination will be based on the entire syllabus including, the practicals performed during laboratory sessions.**

## Text Books:


1. Mukesh Singhal, Niranjan G. Shivaratri, "Advanced Concepts in Operating Systems", Tata McGraw-Hill Edition, 2020.
2. Andrew S. Tanenbaum and Maarten Van Steen, "Distributed Systems: Principles and Paradigms", 2nd Edition, Pearson, 2016.


## Reference Books:


1. K. C. Wang, "Embedded and Real Time Operating System", Springer, January 2022.
2. Abraham Silberschatz, Peter B. Galvin, Greg Gagne, "Operating System Concepts", [illegible] tion, 2017.

3. Rajib Mall, "Real-Time Systems: Theory and Practice", Pearson Education India, 2016.

4. Cris Wolf and Eric M Halter," Virtualization from Desktop to Enterprise", Apress, 2006.

5. K.C. Wang, "Embedded and Real-Time Operating Systems Hardcover" Springer, 2017.

| Program: Computer Engineering | T.Y. B.Tech. | Semester: V |
|---|---|---|
| Advanced Database Management System(RCP23CCPE513) | | |
| Advanced Database Management System Laboratory (RCP23CCPL513) | | |

**Prerequisite:** Basic knowledge of Database Management System.

## Course Objectives:

1. To provide an overview of advancement in SQL and Database technology, distributed database systems and document-oriented database.
2. To impart knowledge of query processing and optimization.
3. To understand the usage of advanced data models for real life applications and secure them.

## Course Outcomes:

On completion of the course, the learner will be able to:

| COs | Course Outcomes | Blooms Level | Blooms Description |
|---|---|---|---|
| CO1 | Discuss new developments in database technology, implement advanced data models for real life applications and secure them. | L2 | Understand |
| CO2 | Optimize query execution and design distributed database for better resource management. | L5 | Evaluate |
| CO3 | Demonstrate the understanding of the concepts related to document-oriented databases. | L3 | Apply |

# Advanced Database Management System(RCP23CCPE513) Course Contents

## Unit-I Advanced Databases — 07 Hrs.

Indexing and Hashing: Types of Single-Level Ordered Indexes; Multilevel Indexes; Dynamic Multilevel Indexes Using B-Trees and B+-Trees; Advanced database applications and architectures: e.g., Data Warehousing; Multimedia database; NoSQL, Native XML databases (NXD), Graph database, Object Oriented Database: Need, Types of object-oriented database, Impedance matching problem between OO languages and Relational database.

## Unit-II Query Processing and Optimization — 07 Hrs.

Query Processing: Overview, Measures of Query cost, Selection operation, Sorting, Join Operations, and other Operations, Evaluation of Expression.

Query Optimization: Translations of SQL Queries into relational algebra, Heuristic approach and cost-based optimization.

## Unit-III Advanced database architectures — 07 Hrs.

Graph Database: Introduction, Graph database architecture, Types of graph database, Graph vs Relational database, Data modeling with graph, Use cases of Graph database, ArangoDB, Giraph. Distributed Database: Introduction, Fragmentation and its types.

## Unit-IV Document Oriented Database — 06 Hrs.

Need of Document Oriented database, difference between Document Oriented Database and Traditional database, Types of encoding XML, Query execution in XML: XPath, XSLT, XQuery, JSON, BSON Representation XML, JSON Objects. Case study on document oriented database: MongoDB.

## Unit-V Advanced Data Models — 06 Hrs.

Temporal Data Models: Aspects of valid time, Bitemporal time and bi-temporal time with Examples of each.

Spatial model: Types of spatial data models - Raster, Vector and Image. MYSQL Postgres, Mobile databases.

## Unit-VI Data Security — 06 Hrs.

Introduction to Database Security Issues; Authentication and authorization, Database auditing, Discretionary Access Control Based on Granting and Revoking Privileges, Mandatory Access Control and Role-Based Access Control for Multilevel Security Introduction to Statistical Database Secur

## Advanced Database Management System Laboratory (RCP23CCPL513)

### List of Laboratory Experiments

**Suggested Experiments:(Any 08)**

1. Case study on Professional and Commercial Databases: Summary and Comparison. E.g. Oracle, MongoDB, Microsoft SQL Server
2. To implement ORDBMS features in MySQL
3. Implementation of Query monitor (QEP- Query Execution Plan, Query Statistics)
4. Simulate Query optimization by applying an SQL Query on any database
5. Implementation of B/B+ Tree
6. Perform Fragmentation (Range, List, Hash and Key) in DDBS design.
7. Implementation of simple graph database.
8. Data handing using JSON.
9. Query execution on XML database
10. Implementation of simple document-oriented database using MongoDB.
11. Case study on Database security issues and measures taken to handle those issues.(Study and document a research paper / patent / product. If possible, suggest an improvement.)

Any other experiment based on syllabus may be included which would help the learner to understand topic/concept.

**Oral examination will be based on the entire syllabus including, the practicals performed during laboratory sessions.**

## Text Books:


1. Abraham Silberschatz, Henry F. Korth, Sudarshan, "Database System Concepts", 7th Edition, Mc Graw Hill, 2021.
2. Sveta Smirnova and Alkin Tezuysal, "My SQL Cookbook", 4th Edition, O'Reilly Publication, 2022.
3. Shannon Bradshaw, Eoin Brazil, "MongoDB: The Definitive Guide - Powerful and Scalable Data Storage", 3rd Edition, O'Reilly Publication, 2020.

4. Christos Tjortjis, "Graph Databases Applications on Social Media Analytics and Smart Cities", 1st Edition, CRC Press, 2023.

## Reference Books:

1. Vinicius M. Grippa and Sergey Kuzmichev, "Learning MySQL", 2nd Edition, O'Reilly Publication, 2021.

2. Tamer OEzsu, Patrick V, "Principles of Distributed Database System", Springer Publication, 2020.

| Program: Computer Engineering | T.Y. B.Tech. | Semester: V |
|---|---|---|
| Computer Graphics and Image Processing (RCP23CCPE514) | | |
| Computer Graphics and Image Processing (RCP23CLPE514) | | |

**Prerequisites:** Knowledge of Basic Mathematics and Algorithms.

## Course Objectives:

1. This course helps the learner to understand theoretical relationships between computer graphics and image processing.
2. This course helps the learner to understand three dimensional environment representation in a computer, transformation of 2D/3D objects, basic mathematical techniques and algorithms used to build useful applications, imaging, and image processing techniques.

## Course Outcomes:

On completion of the course, the learner will be able to:

| COs | Course Outcomes | Blooms Level | Blooms Description |
|---|---|---|---|
| CO1 | Understand the fundamental of Computer Graphics and Image processing. | L2 | Understand |
| CO2 | Illustrate line drawing, circle drawing, clipping and polygon filling algorithms and demonstrate geometric representations, | L3 | Apply |
| CO3 | Demonstrate various transformations on 2D & 3D objects. | L3 | Apply |
| CO4 | Solve image enhancement and segmentation problems using spatial domain Techniques. | L4 | Analyze |

# Computer Graphics and Image Processing(RCP23CCPE514) Course Contents

## Unit-I Basics of Computer graphics and Algorithms 05 Hrs.

Basics of Computer Graphics and its applications. Random Scan Displays and systems, Raster scan displays and systems.

**Fundamentals of Digital Image Processing** Image as 2D data, Image representation in Gray scale, Binary and Color images. Fundamental steps in image processing. Coordinate conventions. Sampling and quantization. Basic relationship between pixels– neighborhood, adjacency, connectivity.

## Unit-II Filled Area Primitives and transformations 10 Hrs.

DDA line drawing, Bresenham's line drawing algorithm. Circle drawing algorithms- Midpoint Circle generation algorithm, Bezier Curves.

**Filled Area Primitives** - Boundary filling and flood filling.

**Two Dimensional Transformations** -Translation, Rotation, Scaling, Reflection and Shearing, Composite transformations, Matrix representations and homogeneous coordinates. Basic 3D transformations.

## Unit-III Clipping and Projections 08 Hrs.

Window to viewport transformation. Cohen Sutherland Line clipping algorithm. Sutherland Hodgeman Polygon clipping algorithm. Three dimensional viewing pipeline.

**Projections**- Parallel and Perspective projections. Visible surface detection algorithms- Depth buffer algorithm, Scan line algorithm.

## Unit-IV Image Enhancement in Spatial Domain 08 Hrs.

Intensity transformations: Image negative, Gray level Slicing, Thresholding, contrast stretching, histogram equalization. Basics of spatial filtering - Smoothing spatial filter Linear and nonlinear filters, and Sharpening Spatial Filters-Gradient and Laplacian.

## Unit-V Image Segmentation 08 Hrs.

Fundamentals of Image Segmentation. Thresholding - Basics of Intensity thresholding and Global Thresholding. Region based Approach - Region Growing, Region Splitting and Merging. Edge Detection - Edge Operators- Sobel and Prewitt.

## Computer Graphics and Image Processing (RCP23CLPE514)

### List of Laboratory Experiments

**Suggested Experiments:(Any 10)**

1. Digital Differential Analyzer Algorithm
2. Bresenham's Line Drawing Algorithm
3. Midpoint Circle Generation Algorithm
4. Ellipse Generation Algorithm
5. Two Dimensional Transformations
6. Coloring the Pictures using polygon filling algorithms
7. Curve Generation
8. Implementation of Image negative, Gray level Slicing and Thresholding
9. Implementation of Contrast Stretching and histogram equalization
10. Implementation of Image smoothing/ Image sharpening
11. Implementation of Edge detection using Sobel and Prewitt masks

Any other experiment based on syllabus may be included which would help the learner to understand topic/concept.

**Oral examination will be based on the entire syllabus including, the practicals performed during laboratory sessions.**

## Text Books:


1. Donald Hearn and M. Pauline Baker, "Computer Graphics", Pearson Education, 2011.
2. Rafael C. Gonzalez and Richard E. Woods, Digital Image Processing. Pearson, 4e, 2017.


## Reference Books:


1. William M. Newman and Robert F. Sproull, "Principles of Interactive Computer Graphics", McGraw Hill, 2001.
2. Zhigang Xiang and Roy Plastock, Computer Graphics (Schaum's outline Series), McGraw Hill, 2019.
3. David F. Rogers, "Procedural Elements for Computer Graphics", Tata McGraw Hill, 2001.

4. M. Sonka, V. Hlavac, and R. Boyle, "Image Processing Analysis and Machine Vision", Thomson India Edition, 4e, 2017.

## Online Resources:

1. https://onlinecourses.nptel.ac.in/noc20_cs90/preview
2. https://www.edx.org/learn/computer-graphics/the-university-of-california-san-diego-computer-graphics

| Program: Computer Engineering | T.Y. B.Tech. | Semester: V |
|---|---|---|
| IoT-Centric Processor Organization and Architecture (RCP23CCMD501) | | |
| IoT-Centric Processor Organization and Architecture Laboratory (RCP23CLMD501) | | |

## Prerequisites:

1. Basic knowledge of digital electronics.
2. Fundamentals of programming.
3. Basic understanding of computer organization.

## Course Objective(s):

1. To understand the fundamental concepts of processor organization and its relevance to IoT systems.
2. To learn memory hierarchy and interfacing techniques between processors and IoT peripherals.
3. To analyze different processor architectures optimized for IoT applications and Develop skills in programming and optimizing IoT processors.

## Course Outcomes:

On completion of the course, the learner will be able to:

| CO | Course Outcomes | Blooms Level | Blooms Description |
|---|---|---|---|
| CO1 | Analyze and evaluate different processor architectures for IoT applications. | L5 | Evaluate |
| CO2 | Design and implement memory systems for resource-constrained IoT devices and Develop efficient programs for IoT processors considering hardware constraints. | L6 | create |
| CO3 | Optimize processor performance using Pipelining and Parallel Processing concepts. | L5 | Evaluat |
| CO4 | Interface various IoT sensors and actuators with processors for optimization and security. | L3 | Apply |

# IoT-Centric Processor Organization and Architecture (RCP23CCMD501) Course Contents

## Unit-I — 03 Hrs.

**Introduction to Processor Architecture:**

- Basic computer organization
- Von Neumann vs Harvard architecture
- RISC vs CISC architecture
- IoT processor requirements and constraints
- Introduction to popular IoT processors (ARM Cortex-M, ESP32, etc.).

## Unit-II — 04 Hrs.

**Memory Systems and Hierarchy:**

- Memory hierarchy in IoT systems
- Cache organization and management
- Virtual memory concepts
- Flash memory and storage systems

## Unit-III — 06 Hrs.

**Instruction Set Architecture:**

- ISA design principles
- Instruction formats and addressing modes
- RISC-V architecture
- ARM instruction set for IoT
- Optimization techniques for IoT applications
- Assembly language programming.

## Unit-IV — 04 Hrs.

**Pipelining and Parallel Processing:**

- Basic pipeline concepts
- Pipeline hazards and solutions
- Parallel processing in IoT context
- Multi-core processors
- Hardware acceleration units

## Unit-V 05 Hrs.

**IoT Interfaces and Peripherals:**

- I/O organization
- Serial communication protocols (I2C, SPI, UART)
- Wireless interfaces (BLE, WiFi, LoRa)
- Sensor interfacing
- Real-time processing requirements
- Interrupt handling

## Unit-VI 04 Hrs.

**Advanced Topics and Optimization:**

- Security features in IoT processors
- Power management techniques
- Real-time operating systems integration
- Hardware accelerators for ML/AI
- Edge computing considerations
- Future trends in IoT processors

# IoT-Centric Processor Organization and Architecture Laboratory (RCP23CLMD501)

### List of Laboratory Experiments

**Suggested Experiments:(Any 8)**

1. Introduction to development boards (ARM/ESP32) and tool chains.
2. Assembly language programming for basic operations.
3. Memory access and management experiments.
4. Cache performance analysis.
5. Implementation of pipeline concepts.
6. I2C sensor interfacing and data acquisition.
7. SPI communication with external devices.
8. UART communication implementation.
9. WiFi/BLE module interfacing.

10. Interrupt handling and real-time processing.

11. Complete IoT system integration project.

Any other experiment based on syllabus may be included which would help the learner to understand topic/concept.

**Oral examination will be based on the entire syllabus including, the practicals performed during laboratory sessions.**

## Text Books:


1. Patterson, D. A., & Hennessy, J. L., "Computer Organization and Design RISC-V Edition: The Hardware Software Interface", 2nd Edition, Morgan Kaufmann, 2023.

2. Sarah Harris & David Harris, "Digital Design and Computer Architecture: RISC-V Edition", Morgan Kaufmann, 2024.

3. Raj Kamal, "Internet of Things: Architecture and Design Principles", McGraw Hill Education, 2023.


## Reference Books:


1. William Stallings, "Computer Organization and Architecture", 11th Edition, Pearson, 2023.

2. Perry Xiao, "IoT System Design: A Practical Approach with Microcontrollers", CRC Press, 2023.

3. ARM Limited, "ARM Architecture Reference Manual", ARM Holdings, 2024.




| Program: Computer Engineering | T.Y. B.Tech. | Semester: V |
|---|---|---|
| Semester Project - III (RCP23IPSC501) | | |

## Course Objective:

Students are expected to design, simulate/implement a project based on the knowledge acquired from current semester subjects.

## Course Outcomes:

On completion of the course, the learner will be able to:

| COs | Course Outcomes | Blooms Level | Blooms Description |
|---|---|---|---|
| CO1 | Conduct a survey of several available literatures in the preferred field of study. | L4 | Analyze |
| CO2 | Demonstrate various/alternate approaches to complete a project. | L2 | Understand |
| CO3 | Ensure a collaborative project environment by interacting and dividing project work among team members. | L3 | Apply |
| CO4 | Present their project work in the form of a technical report / paper and thereby improve the technical communication skill. | L3 | Apply |
| CO5 | Demonstrate the ability to work in teams and manage the conduct of the research study. | L2 | Understand |

**Semester Project:**

The purpose of semester project is to provide exposure to students with a variety of projects based on the knowledge acquired from the semester subjects. This activity is supposed to enrich their academic experience and bring enough maturity in student while selecting the project. Students should take this as an opportunity to develop skills in implementation, presentation and discussion of technical ideas/topics. Therefore, proper attention shall be paid to the content of semester project report which is being submitted in partial fulfillment of the requirements of the Second Year and it is imperative that a standard format be prescribed for the report.

Each student shall work on project approved by departmental committee approved by the Head of Department, a group of 03 to 05 students (max allowed: 5 students in extraordinary cases, subject to the approval of the department committee and the Head of the department) shall be allotted for each Semester Project. Each group shall submit at least 3 topics for the Semester Project. The departmental committee shall finalize one topic for every group. Semester Project Title or Theme should be based on knowledge acquired during semester. The project work shall involve sufficient work so that students get acquainted with different aspects of knowledge acquired from semester subjects.

**Student is expected to:**

- Select appropriate project title based on acquired knowledge from current semester subjects.
- Maintain Log Book of weekly work done(Log Book Format will be as per Table 1).
- Report weekly to the project guide along with log book.

**Assessment Criteria:**

- At the end of the semester, after confirmation by the project guide, each project group will submit project completion report in prescribed format for assessment to the departmental committee (including project guide).
- Assessment of the project (at the end of the semester) will be done by the departmental committee (including project guide).

**Prescribed project report guidelines:**

Size of report shall be of minimum 25 pages. Project Report should include appropriate content for:

- Introduction
- Literature Survey
- Related Theory
- Implementation details

- Project Outcomes
- Conclusion
- References

**Assessment criteria for the departmental committee (including project guide) for Continuous Assessment:**

Guide will monitor weekly progress and marks allocation will be as per Table 2.

**Assessment criteria for the departmental committee (including project guide) for End Semester Exam:**

Departmental committee (including project guide) will evaluate project as per Table 3.

Table 1: Log Book Format

| Sr | Week (Start Date:End Date) | Work Done | Sign of Guide | Sign of Coordinator |
|---|---|---|---|---|
| 1 | | | | |
| 2 | | | | |

Table 2: Continuous Assessment Table

| Sr | Exam Seat No | Name of Student | Student Attendance | Log Book Maintenance | Literature Review | Depth of Understanding | Report | Total |
|---|---|---|---|---|---|---|---|---|
| | | | 5 | 5 | 5 | 5 | 5 | 25 |

Table 3: Evaluation Table

| Sr | Exam Seat No | Name of Student | Project Selection | Design/ Simulation/ Logic | Hardware/ Programming | Result Verification | Presentation | Total |
|---|---|---|---|---|---|---|---|---|
| | | | 5 | 5 | 5 | 5 | 5 | 25 |

| Program: Computer Engineering | T.Y. B.Tech. | Semester: V |
|---|---|---|
| Constitution of India (RCP23ICHSX05) | | |

## Course Objective(s):

1. To provide basic information about Indian constitution.
2. To identify individual role and ethical responsibility towards society.
3. To understand human rights and its implications.

## Course Outcomes:

On completion of the course, the learner will be able to:

| CO | Course Outcomes | Blooms Level | Blooms Description |
|---|---|---|---|
| CO1 | Have general knowledge and legal literacy and thereby to take up competitive examinations. | L3 | Apply |
| CO2 | Understand state and central policies, fundamental duties | L2 | Understand |
| CO3 | Understand Electoral Process, special provisions. | L2 | Understand |
| CO4 | Understand powers and functions of Municipalities, Panchayats and Co-operative Societies. | L2 | Understand |
| CO5 | Understand Engineering ethics and responsibilities of Engineers. | L2 | Understand |
| CO6 | Understand Engineering Integrity & Reliability. | L2 | Understand |

# Constitution of India (RCP23ICHSX07)
# Course Contents

## Unit-I Introduction to the Constitution of India — 02 Hrs.

The Making of the Constitution and Salient features of the Constitution, Preamble to the Indian Constitution, Fundamental Rights & its limitations.

## Unit-II Directive Principles of State Policy — 02 Hrs.

Relevance of Directive Principles State Policy, Fundamental Duties.

Union Executives – President, Prime Minister, Parliament, Supreme Court of India.

## Unit-III State Executives — 03 Hrs.

Governor, Chief Minister, State Legislature, High Court of State.

Electoral Process in India, Amendment Procedures, $42^{nd}$, $44^{th}$, $74^{th}$, $76^{th}$, $86^{th}$ & $91^{st}$ Amendments.

## Unit-IV Special Provisions — 02 Hrs.

For SC & ST, Special Provision for Women, Children & Backward Classes, Emergency Provisions.

## Unit-V Human Rights — 02 Hrs.

Meaning and Definitions, Legislation Specific Themes in Human Rights- Working of National Human Rights Commission in India, Powers and functions of Municipalities, Panchayats and Co-Operative Societies.

## Unit-VI Scope & Aims of Engineering Ethics — 02 Hrs.

Responsibility of Engineers Impediments to Responsibility. Risks, Safety and liability of Engineers, Honesty, Integrity & Reliability in Engineering.

## Text Books:


1. Durga Das Basu, "Introduction to the Constitution on India", (Student Edition) Prentice-Hall EEE, $25^{th}$ Edition, 2021.
2. Charles E. Haries, Michael S. Pritchard and Michael J. Robins, "Engineering Ethics", Thompson Asia, 2003.

## Reference Books:


1. M. V. Pylee, "An Introduction to Constitution of India", Vikas Publishing, 2003.
2. M. Govindarajan, S. Natarajan, V. S. Senthilkumar, "Engineering Ethics", Prentice-Hall of India Pvt. Ltd. New Delhi, 2013.
3. Brij Kishore Sharma, pl"Introduction to the Constitution of India", $7^{th}$ Edition, PHI Learning Pvt. Ltd., New Delhi, 2015.
4. Latest Publications of Indian Institute of Human Rights, New Delhi.


## Online Resources:

1. www.nptel.ac.in
2. www.hnlu.ac.in
3. www.nspe.org
4. www.preservearticles.com

# Semester - VI

| Program: Computer Engineering | T.Y. B.Tech. | Semester: VI |
|---|---|---|
| Machine Learning(RCP23CCPC601) | | |
| Machine Learning Laboratory (RCP23CLPC601) | | |

**Prerequisites:** Data Structures, Basic Probability and Statistics, Data Mining.

## Course Objectives:

1. To introduce students to the basic concepts and techniques of Machine Learning.
2. To become familiar with Dimensionality reduction Techniques.
3. To become familiar with regression, classification and clustering tasks.

## Course Outcomes:

On completion of the course, the learner will be able to:

| COs | Course Outcomes | Blooms Level | Blooms Description |
|---|---|---|---|
| CO1 | Gain knowledge about basic concepts of Machine Learning. | L2 | Understand |
| CO2 | Apply supervised learning algorithms. | L3 | Apply |
| CO3 | Apply unsupervised learning algorithms. | L3 | Apply |
| CO4 | Design application using machine learning techniques. | L6 | Create |

# Machine Learning(RCP23CCPC601)
# Course Contents

## Unit-I Introduction to Machine Learning — 06 Hrs.

Types of Machine Learning, Steps involved in developing a Machine Learning Application, Evaluating a Learning Algorithm: Deciding what to try next, Evaluating Hypothesis, Model Selection and Train/ Validation/ Test Sets, Bias Vs variance: Regularization and Bias/ Variance, Learning Curve, Error Analysis, Handling Skewed Data: Error Matrices for Skewed Classes, Tradeoff between Precision and recall, Issues in Machine Learning, Application of Machine Learning

## Unit-II Dimensionality Reduction — 07 Hrs.

Dimensionality Reduction Techniques: Principal components analysis (Eigen values, Eigen vectors, Orthogonality), Independent Component Analysis, Singular value decomposition

## Unit-III Supervised Learning: — 14 Hrs.

**Learning with Regression and trees:** Learning with Regression: Simple Linear Regression, Multiple Linear Regression, Logistic Regression. Learning with Trees: Decision Trees, Constructing Decision Trees using Gini Index, Classification and Regression Trees (CART).

**Classification:** Classification using Bayesian Belief networks, Hidden Markov Models Support Vector Machine: Maximum Margin Linear Separators, Quadratic Programming solution to finding maximum margin separators, Kernels for learning non-linear functions. Classification using k Nearest Neighbor Algorithm

## Unit-IV Unsupervised Learning: — 07 Hrs.

**Clustering:** Basics of clustering, Hard vs Soft Clustering, Density Based Clustering: DBSCAN, Expectation maximization (EM) for soft clustering, Semi-supervised learning with EM using labelled and unlabeled data, Radial Basis functions.

## Unit-V Applications of Machine Learning: — 05 Hrs.

Recommender Systems, Machine Learning for Image Recognition, Sentiment Analysis, Machine Learning for video surveillance

## Machine Learning Laboratory (RCP23CLPC601)

### List of Laboratory Experiments

**Suggested Experiments:(Any 08)**

1. To implement Linear Regression.
2. To implement Logistic Regression.
3. To implement CART decision tree algorithm.
4. To implement Support Vector Machine.
5. To implement Bayesian Classification.
6. To implement PCA.
7. To implement K-Nearest Neighbour.
8. To implement Radial basis functions.
9. Mini project based on any machine learning application.

Any other experiment based on syllabus may be included which would help the learner to understand topic/concept.

**Oral examination will be based on the entire syllabus including, the practicals performed during laboratory sessions.**

## Text Books:


1. Ethem Alpaydm, "Introduction to Machine Learning", 4th Edition, MIT Press, 2020.
2. Peter Harrington, "Machine Learning In Action", 1st Edition, DreamTech Press, 2012.
3. Tom M. Mitchell, "Machine Learning", 1st Edition, McGraw Hill, 2017.
4. Andreas C. Müller and Sarah Guido, "Introduction to Machine Learning with Python: A Guide for Data Scientists", O'reilly, 2016.
5. Kevin P. Murphy, "Machine Learning: A Probabilistic Perspective", MIT Press, 2012.


## Reference Books:


1. Aurélien Géron, "Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow", 2nd Edition, 2019.
2. Witten Ian H., Eibe Frank, Mark A. Hall, and Christopher J. Pal. "Data Mining: Practical machine learning tools and techniques", 1st Edition, Morgan Kaufmann, 2016.

3. Han Kamber, "Data Mining Concepts and Techniques", 3rd Edition, Morgann Kaufmann Publishers, 2012.

4. Mehryar Mohri, Afshin Rostamizadeh, and Ameet Talwalkar, "Foundations of Machine Learning", The MIT Press, 2012.

5. H. Dunham, "Data Mining: Introductory and Advanced Topics", 1st Edition, Pearson Education, 2006.

| Program: Computer Engineering | T.Y. B.Tech. | Semester: VI |
|---|---|---|
| Software Engineering (RCP23CCPC602) | | |
| Software Engineering Laboratory (RCP23CLPC602) | | |

## Prerequisites:

1. Concepts of Object Oriented Programming & Methodology.
2. Knowledge of developing applications with front end & back end connectivity.

## Course Objective:

1. To explore the essential phases and critical aspects of an overall software development process in order to design a high-quality software solution in cost-effective manner for a real-world problem.

## Course Outcomes:

On completion of the course, the learner will be able to:

| COs | Course Outcomes | Blooms Level | Blooms Description |
|---|---|---|---|
| CO1 | Understand basic concepts of Software Engineering process and models. | L2 | Understand |
| CO2 | Identify requirements, analyse and design for real world software projects. | L4 | Analyze |
| CO3 | Plan, schedule and track the progress of the project using various software project management tools. | L3 | Apply |
| CO4 | Identify risks and create mitigation plans for dealing with real world risks, manage the configuration and change in software. | L4 | Analyze |
| CO5 | Apply testing principles on software projects. | L3 | Apply |

# Software Engineering (RCP23CCPC602)
# Course Contents

## Unit-I — 08 Hrs.

**Introduction to Software Engineering and Process Models:** Nature of Software, Software Engineering, Software Process, CMM, Generic Process Model.

**Prescriptive Process Models**: The Waterfall Model, V Model.

**Incremental Process Model**: Incremental Model, RAD Model

**Evolutionary Process Models**: Prototyping Paradigm, The Spiral Model

**Concurrent Process Models**: Concurrent Process Model

**Agile Methodology**: Agility Principals, Agile Process Models: Extreme Programming (XP), Adaptive Software Development (ASD), Dynamic Systems Development Method (DSDM), Scrum, Crystal, Feature Driven Development (FDD), Agile Modeling (AM), Kanban Model, LEAN models.

## Unit-II — 07 Hrs.

**Requirement Analysis**: Requirement Elicitation, Software Requirement Specification (SRS).

**Requirement Models**: Scenario Based Models, Class Based Models, Behavioural Models and Flow Models.

## Unit-III — 06 Hrs.

**Design Engineering and Analysis**: Design Principles, Design Concepts, Effective Modular Design-Cohesion and Coupling.

**Translating the requirement models into the design model.**

**Designs** Architectural Design, Component Level Design, User Interface Design.

## Unit-IV — 08 Hrs.

**Software Project Management**:

**Project Management Concepts**: Management Spectrum, 3Ps

**Process and Project Metrics**: Metrics in the Process and Project Domains, software measurement, metrics for software quality.

**Software Project Estimation**: LOC, FP, Empirical Estimation Models COCOMO I, COCOMO II, Specialized Estimation Techniques.

**Software Project Scheduling**: Work Breakdown Structure, Network Diagram, Gantt Chart, PERT, CPM, Stakeholders and Communication plan, Introduction to Project Management Information System (PMIS).

## Unit-V 05 Hrs.

**Software Risk Management**: Risk Identification, Risk Assessment, Risk Projection, Risk Refinement, RMMM Plan.

**Software Configuration Management**: SCM, SCM Repositories, SCM Process, Change Control and Version Control.

## Unit-VI 05 Hrs.

**Software Testing Fundamentals**: Strategic Approach to Software Testing, Unit Testing, Integration Testing, Verification, Validation Testing, System Testing, Test Strategies for WebApps **Software Testing Techniques**: White Box Testing, Basis Path Testing, Control Structure Testing and Black Box Testing. TDD.

## Software Engineering Laboratory (RCP23CLPC602)

### List of Laboratory Experiments

**Suggested Experiments:(Any 8)**

1. Prepare detailed statement of problem for the selected / allotted mini project and identify suitable process model for the same with justification.
2. Develop Software Requirement Specification (SRS) document in IEEE format for the project.
3. Identify scenarios & develop UML Use Case and Class Diagram for the project.
4. Draw DFD (upto 2 levels) and prepare Data Dictionary for the project.
5. Develop Activity / State Transition diagram for the project.
6. Develop Sequence and Collaboration diagram for the project.
7. Use project management tool to prepare schedule and estimation for the project.
8. Prepare RMMM plan for the project.
9. Change specification and make different versions using any SCM Tool.
10. Develop test cases for the project using testing techniques.

Any other experiment based on syllabus may be included which would help the learner to understand topic/concept.

**Oral examination will be based on the entire syllabus including, the practicals performed during laboratory sessions.**

## Text Books:


1. Roger Pressman, "Software Engineering: A Practitioner's Approach", 8th Edition, McGraw-Hill Publications, 2015.
2. Ian Sommerville, "Software Engineering", 9th Edition, Pearson Education, 2011.
3. Ali Behfrooz and Fredeick J. Hudson, "Software Engineering Fundamentals", Oxford University Press.
4. Ugrasen Suman, "Software Engineering – Concepts and Practices", Cengage Learning, 2022.


## Reference Books:


1. Pankaj Jalote, "Software Engineering: A Precise Approach", Wiley Publications, 2010.
2. Pankaj Jalote, "An Integrated Approach to Software Engineering", Springer, 2005.
3. Jibitesh Mishra and Ashok Mohanty, "Software Engineering", Pearson, 2012.
4. Rajib Mall, "Fundamentals of Software Engineering", 4th Edition, Prentice Hall India, 2014.

| Program: Computer Engineering | T.Y. B.Tech. | Semester: VI |
|---|---|---|
| DevOps Laboratory (RCP23CLPC603) | | |

## Prerequisites:

1. Knowledge of Linux Operating system, installation and configuration of services and command line basics.
2. Software Development Life cycle.

## Course Objective(s):

1. To understand the fundamentals of DevOps engineering.
2. To be proficient with DevOps terminologies, concepts, benefits, and deployment options to meet real world software development requirements.

## Course Outcomes:

On completion of the course, the learner will be able to:

| CO | Course Outcomes | Blooms Level | Blooms Description |
|---|---|---|---|
| CO1 | Interpret and apply various principles, phases and activities of Agile methodology and DevOps principles to meet software development requirements. | L3 | Apply |
| CO2 | Understand and implement DevOps principles for CI/CD. | L3 | Apply |
| CO3 | Apply testing process for application development and configuration management tools. | L3 | Apply |

# DevOps Laboratory ( (RCP23CLPC603)
# Course Contents

## Unit-I — 02 Hrs.

**Introduction to DevOps:** Phases of Software Lifecycle, Minimum Viable Product (MVP) & Cross-functional Teams, Lean, ITIL, Agile development methodologies, DevOps as a prominent culture to achieve agility in the software development process, DevOps Stakeholders, Goals, DevOps and Agile, DevOps Tools.

## Unit-II — 03 Hrs.

**Version Control:** Introduction, Overview of Version Control Systems, Role of Version Control System, Types of Control Systems and their Supporting Tools, Importance of version control in CICD pipeline.

## Unit-III — 06 Hrs.

**Continuous Integration:** Introduction to Jenkins (With Master -Slave Architecture), Choosing a launch method, Administering Jenkins slaves, Labels, groups and load balancing. Creating Views and Jobs in Jenkins: The Jenkins user interface, Jobs in Jenkins, Creating Views, Managing Views and Jobs in Jenkins: Managing Views in Jenkins, Navigating a job's project page, Job Execution, The Job Execution Configuration Panel, The Status Panel, Console Panel.

## Unit-IV — 04 Hrs.

**Continuous Deployment:** Overview of Docker, Benefits of Docker Workflow, Process Simplification, Architecture, Docker Containers, Docker Workflow, Anatomy of Dockerfile, Building an Image, Running an Image, Custom base Images, Storing Images.

## Unit-V — 06 Hrs.

**Continuous Testing:** Introducing WebDriver and WebElements, Selenium Testing Tools, Differences between Selenium 2 and Selenium 3, Setting up a project in Eclipse with Maven and TestNG using Java, WebElements, Locating WebElements using WebDriver, Interacting with WebElements, Different Available WebDrivers, Using Java 8 Features with Selenium. Introducing Java 8 Stream API, Using Stream API with Selenium WebDriver. .

## Unit-VI — 08 Hrs.

**Continuous Management:** The Parts of an Infrastructure System, Infrastructure Platforms, Infrastructure Resources, Compute Resources, Storage Resources, Network Resources.

**Puppet:** Puppet Architecture, The Puppet Server, setting up the Puppet Agent, Performing Ad-Hoc Op-

timizations.

**Ansible:** Ansible Architecture, Ansible and Infrastructure Management, Local Infrastructure

**Development:** Ansible and Vagrant.Introduction to open-source tools for data gathering and management, AWS

## DevOps Laboratory (RCP23CLPC603)

### List of Laboratory Experiments

**Suggested Experiments:(Any 08)**

1. To understand Version Control System / Source Code Management, install git and create a GitHub account.
2. Perform various GIT operations on local and Remote repositories using GIT Cheat-Sheet.
3. To understand Continuous Integration, install and configure Jenkins with Maven/Ant/Gradle to setup a build Job.
4. To Build the pipeline of jobs using Maven / Gradle / Ant in Jenkins, create a pipeline script to Test and deploy an application over the tomcat server.
5. To understand Jenkins Master-Slave Architecture and scale your Jenkins standalone implementation by implementing slave nodes.
6. To Setup and Run Selenium Tests in Jenkins Using Maven.
7. To understand Docker Architecture and Container Life Cycle, install Docker and execute docker commands to manage images and interact with containers.
8. To learn Dockerfile instructions, build an image for a sample web application using Dockerfile.
9. To install and Configure Pull based Software Configuration Management and provisioning tools using Puppet /Ansible.
10. To learn Software Configuration Management and provisioning using Puppet Blocks (Manifest, Modules, Classes, Function).
11. To provision a LAMP/MEAN Stack using Puppet Manifest.

Any other experiment based on syllabus may be included which would help the learner to understand topic/concept.

**Practical examination will be based on the entire syllabus including, the practicals performed during laboratory sessions.**

## Text Books:


1. John Dean, "Web Programming with HTML5, CSS3 and JavaScript", Jones & Bartlett Learning, 2019.
2. Karl Matthias & Sean P. Kane, "Docker: Up and Running", 3rd Edition, O'Reilly Publication, 2022.
3. Craig Berg, "DevOps For Beginners: A Complete Guide To DevOps Best Practices" 2020.
4. Mikael Krief, "Learning DevOps: A comprehensive guide to accelerating DevOps culture adoption with Terraform, Azure DevOps, Kubernetes, and Jenkins", Packt Publication, 2nd Edition, 2022.
5. Gene Kim, Jez Humble, et.el, "The DevOps Handbook: How to Create World-Class Agility.
6. Reliability, & Security in Technology Organizations", IT Revolution Press; 2nd edition 2021.
7. Mark Reed, "DevOps: The Ultimate Beginners Guide to Learn DevOps Step-By Step",LLC Publication, 2020.
8. Deepak Gaikwad, Viral Thakkar, "DevOps Tools from Practitioner's Viewpoint",Wiley, 2019.


## Reference Books:


1. Mark S. Merkow, "Practical Security for Agile and DevOps", CRC Press Taylor & Francis,2022.
2. Emily Freeman, "DevOps for Dummies", 3rd Edition, Wiley Publication, 2019.
3. Martin Alfke, Felix Frank, "Puppet 5 Essentials", 3rd Edition, O'Reilly Publication, 2017.

| Program: Computer Engineering | T.Y. B.Tech. | Semester: VI |
|---|---|---|
| Big Data Analytics (RCP23CCPE611) | | |
| Big Data Analytics Laboratory (RCP23CLPE611) | | |

**Prerequisite:** Database Management Systems, Data Warehousing and Mining.

## Course Objective(s):

1. Analyze Big Data characteristics & evaluate appropriate analytical approaches for different business scenarios.
2. Design MapReduce solutions and implement stream processing techniques for large-scale data handling.
3. Construct Big Data mining solutions and apply analytics frameworks to solve industry problems.

## Course Outcomes:

On completion of the course, the learner will be able to:

| CO | Course Outcomes | Blooms Level | Blooms Description |
|---|---|---|---|
| CO1 | Differentiate between various Big Data frameworks and choose appropriate tools for specific data processing needs. | L4 | Analyze |
| CO2 | Develop MapReduce programs and optimize stream processing algorithms for real-time data analysis. | L6 | Create |
| CO3 | Examine complex datasets and apply mining algorithms to extract meaningful patterns. | L5 | Evaluate |
| CO4 | Create scalable Big Data applications and integrate recommendation systems and social network analytics. | L6 | Create |

# Big Data Analytics (RCP23CCPE611)
# Course Contents

## Unit-I — 04 Hrs.

**Introduction to Big Data and Big Data Analytics:** What is Big Data?, Big Data Characteristics, Drivers, Big Data Characteristics,Traditional vs Big Data, Types of Big Data, Big Data Challenges, Examples of Big Data in Real Life, Big Data Applications. Introduction to Big Data Analytics, Types of Big Data Analytics, Big Data Analytics Process Models, Advantages of Big Data Analytics, Challenges in Big Data Analytics.

## Unit-II — 08 Hrs.

**Big Data Frameworks - Hadoop:** Hadoop Core Components, Hadoop Ecosystem, Hadoop Physical Architecture, Working of Hadoop, Advantages and Limitations of Hadoop.

**HDFS:** Overview of HDFS, HDFS Architecture, HDFS Commands

**Hive:** Features of Hive, Hive Architecture, Hive Vs Traditional Database, HiveQL

**HBase:** Introduction to HBase, HBase Architecture, HBase Vs RDBMS, HBase Schema, Indexing in HBase, HBase Shell Queries

**Pig:** Features, Execution Modes of Pig, Pig Vs SQL, Pig Architecture, Pig Data Model

**Zookeeper:** Features, Architecture of Zookeeper, Working of Zookeeper

## Unit-III — 04 Hrs.

**MapReduce Paradigm :** The Map Tasks, Grouping by Key, The Reduce Tasks, Combiners, Details of MapReduce Execution, Coping With Node Failures. Algorithms Using MapReduce: Word count by MapReduce, Matrix-Vector Multiplication by MapReduce,Relational-Algebra Operations, Computing Selections by MapReduce, Computing Projections by MapReduce, Union ,Intersection, and Difference by MapReduce.

## Unit-IV — 08 Hrs.

**Mining Big Data Streams - The Stream Data Model:** A DataStream-Management System, Examples of Stream Sources, Issues in Stream Processing, Sampling Data in a Stream: Sampling Techniques, Data Ingestion

**Spark:** Features of Spark, Components of Spark, Architecture of Spark, RRD in Spark, SparkQL, Schedulers in Spark, Shared Variables in Spark.

**Kafka:** Features of Kafka, Components of Kafka, Kafka Cluster Architecture, Kafka Workflow

**Filtering Streams:** The Bloom Filter **Counting Distinct Elements in a Stream:** The Count-Distinct Problem, The Flajolet-Martin Algorithm,

**Counting Ones in a Window :** The Cost of Exact Counts, The Datar-Gionis-In[illegible]

gorithm, Query Answering in the DGIM Algorithm.

## Unit-V 08 Hrs.

**Big Data Mining Algorithms - Frequent Pattern Mining Algorithms:** Handling Larger Datasets in Main Memory,Park, Chen, and Yu (PCY) Algorithm, The SON Algorithm and MapReduce.

**Clustering Algorithms:** CURE Algorithm. Canopy Clustering, Clustering with MapReduce.

**Classification Algorithms:** Overview SVM classifiers, Parallel SVM, K-Nearest Neighbor classifications for Big Data, One Nearest Neighbor

## Unit-VI 07 Hrs.

**Big Data Analytics Applications - Link Analysis:** PageRank Definition, Structure of the web, dead ends, Using Page rank in a search engine, Efficient computation of Page Rank, Topic sensitive Page Rank, Link Spam, Hubs and Authorities, HITS Algorithm.

**Recommendation Engines:** A Model for Recommendation Systems, Content-Based Recommendations, Collaborative Filtering.

**Mining Social- Network Graphs:** Social Networks as Graphs, Types , Clustering of Social Network Graphs, Direct Discovery of Communities.

# Big Data Analytics Laboratory (RCP23CLPE611)

### List of Laboratory Experiments

**Suggested Experiments:(Any 8)**

1. Installation of Hadoop on a single node cluster.
2. Execution of HDFS Commands.
3. Implementation of Map Reduce program to count words in a text file, matrix multiplication
4. Execute HIVE commands to load, insert, retrieve, update, or delete data in the tables.
5. Execute HBASE commands to perform basic CRUD operations and joins.
6. To create RDD, perform various operations and find occurrence of each word.
7. To create SparkQL and execute various SQL commands.
8. Perform Sentiment analysis using Spark Streaming.
9. Implementation of DGIM algorithm

10. Implementation of PCY, SON algorithm.

11. Implementation of PageRank, HITS algorithm

12. Implementation of Recommendation System.

Any other experiment based on syllabus may be included which would help the learner to understand topic/concept.

**Oral examination will be based on the entire syllabus including, the practicals performed during laboratory sessions.**

## Text Books:


1. Seema Acharya, Subhashini Chellappan, Big Data and Analytics , 2nd Edition, Wiley, 2022.

2. Radha Shankarmani, M. Vijayalakshmi, Big Data Analytics , 2nd Edition, Wiley, 2016.

3. Tom White, Hadoop: The Definitive Guide , 4th Edition,O'Reilly, 2015.


## Reference Books:


1. Jure Leskovec, Anand Rajaraman, Jeffrey D. Ullman, Mining of Massive Datasets , 3rd Edition,Cambridge University Press, 2020.

2. Rajkumar Buyya, Rodrigo N. Calheiros, Amir Vahid Dastjerdi, Big Data Principles and Paradigms, Morgan Kaufmann, 2016.

3. Vignesh Prajapati, Big Data Analytics with R and Hadoop, Packt Publishing Limited, 2013.

| Program: Computer Engineering | T.Y. B.Tech. | Semester: VI |
|---|---|---|
| Game Theory (RCP23CCPE612) | | |
| Game Theory Laboratory (RCP23CLPE612) | | |

**Prerequisite:** Analysis of Algorithms.

## Course Objectives:

1. To explain and predict how individuals behave in a specific strategic situation, and therefore help improve decision making.
2. To explain in depth, the standard equilibrium concepts in Game Theory.
3. To illustrate the concepts, real-world examples and case studies.
4. To design Repeated Games with public information.
5. To design static and Dynamic games with incomplete information.

## Course Outcomes:

On completion of the course, the learner will be able to:

| COs | Course Outcomes | Blooms Level | Blooms Description |
|---|---|---|---|
| CO1 | Identify strategic situations and represent them as games. | L1 | Remember |
| CO2 | Solve simple games using various techniques. | L3 | Apply |
| CO3 | Recommend and prescribe which strategies to implement. | L3 | Apply |
| CO4 | Develop Static and Dynamic Games. | L6 | Create |
| CO5 | Develop Repeated Games. | L6 | Create |

# Game Theory (RCP23CCPE612)
# Course Contents

## Unit-I Introduction to Game Theory — 07 Hrs.

Games and solutions - Game theory and mechanism design - Examples from networks - Strategic form games - Matrix and continuous games - Iterated strict dominance - Rationalizability - Nash Equilibrium - existence and uniqueness - Mixed and correlated equilibrium – Super modular games - Potential/congestion games - Existence and Properties of Nash Equilibria.

## Unit-II Extensive-Form Games — 04 Hrs.

SDefinition - Strategies and Equilibria in Extensive Form Games - Backward Induction and Subgame Perfection and its Critiques.

## Unit-III Repeated Games — 07 Hrs.

Infinitely/finitely repeated games - Pareto Perfection and Renegotiation - Proofness in Repeated Games - Repeated Games with incomplete Public Information - Trigger strategies - Fork Theorem with Imperfect Public Information.

## Unit-IV Static Games with incomplete information — 06 Hrs.

Mixed and Behavioral strategies - Bayesian Nash equilibrium - Applications in auctions - Different auction formats - Revenue and efficiency properties of different auctions - Bayesian Games.

## Unit-V Mechanism Design — 08 Hrs.

Mechanism Design Principle - Single Agent - Several Agents, social choice functions and mechanisms, incentive compatibility and revelation theorem, the Gibbard-Satterthwaite impossibility theorem, quasilinear mechanisms, Vickrey- Clarke-Groves (VCG) mechanisms.

## Unit-VI Dynamic Games with incomplete information — 07 Hrs.

Introduction - Perfect Bayesian Equilibrium in Multi-stage games - Extensive-Form and Strategic-Form Refinements - Reputation Effects - Sequential Bargaining under Incomplete Information.

# Game Theory Laboratory (RCP23CLPE612)

## List of Laboratory Experiments

**Suggested Experiments:(Any 08)**

1. Construct and evaluate Nash equilibrium for Prisoner's Dilemma using nashpy, without using nashpy
2. Construct and evaluate the Game tree for the Battle of Sexes in Gambit
3. Construct and Evaluate Nash equilibrium for Hunting Game using nashpy, without using nashpy
4. Matching Pennies: Two payers, each having a penny, are asked to choose from among two strategies – heads (H) and tails (T). The row player wins if the two pennies match, while the column player wins if they do not. Find Nash Equilibrium nashpy, without using nashpy.
5. Routing congestion game: player 1 is interested in getting good service, hence would like theothers to choose different routes, while player 2 is interested only in disrupting player 1's serviceby trying to choose the same route. Find Nash Equilibrium nashpy, without using nashpy
6. SENATE RACE: An incumbent senator (from a rightist party) runs against a challenger (from a leftist party). They first choose a political platform, leftist or rightist, where the senator has to move first. If both choose the same platform, the incumbent wins, otherwise the challenger wins. Assume that the value of winning is 10, the value of compromising their political views (by choosing a platform not consistent with them) is 5, and the payoff is the sum of these values. Use Gambit to illustrate a Game tree.
7. BACKWARD INDUCTION: Six stones lie on the board. Black and White alternate to remove either one or two stones from the board, beginning with White. Whoever first faces an empty board when having to move loses. The winner gets $1, \texttt{theloserloses1}$. What are the best strategies for the players? Use Gambit to illustrate a Game tree
8. CENTIPEDE GAME: In this game, two players alternately face two stacks of money. To make a move, a player has the choice either to pass, in which case both stacks grow slightly and the other player now must make a move facing slightly larger stacks, or to take the larger stack, in which case the other player gets the smaller one and the game ends. If it didn't end before, the game ends after a fixed number of rounds, in which case both players share the accumulated money evenly. Use Gambit to illustrate a Game tree. Implement the CENTIPEDE GAME and find out Nash equilibrium
9. Mini Project

Any other experiment based on syllabus may be included which would help the learner to [illegible] and topic/concept.

**Oral examination will be based on the entire syllabus including, the practicals performed during laboratory sessions.**

## Text Books:


1. Ana Espinola-Arredondo, Felix Muñoz-Garcia, "Game Theory-An Introduction with Step-by-Step Examples", Springer Nature Link, 2023.
2. Fudenberg, Drew, Jean Tirole, "Game Theory", Cambridge, MA: MIT Press, 1991
3. Osborne, M. J., and Rubinstein, A., "A Course in Game Theory", Cambridge, MA: MIT Press, 1994.
4. D. Fudenberg and J. Tirole, "Game Theory", The MIT Press, 2005


## Reference Books:


1. Nisan, Noam, Tim Roughgarden, Eva Tardos, Vijay V. Vazirani, "Algorithmic Game Theory", Cambridge, UK: Cambridge University Press, 2007.
2. Shoham, Y. and Leyton-Brown, K., "Multiagent Systems: Algorithmic, Game Theoretic, and Logical Foundations", Cambridge University Press, 2008.

| Program: Computer Engineering | T.Y. B.Tech. | Semester:VI |
|---|---|---|
| Computer Vision(RCP23CCPE613) | | |
| Computer Vision Laboratory (RCP23CLPE613) | | |

**Prerequisites:** A sound knowledge of Computer Graphics and Image Processing.

## Course Objectives:

1. To introduce fundamental concepts of computer vision and equip students with foundational knowledge in image formation, photometric processing, and geometric transformations.
2. To develop practical skills in feature extraction, recognition methodologies, and morphological image processing techniques for object segmentation and detection.
3. To explore advanced applications in feature-based alignment, 3D vision, and recognition, including face detection, pose estimation, and scene understanding.

## Course Outcomes:

On completion of the course, the learner will be able to:

| COs | Course Outcomes | Blooms Level | Blooms Description |
|---|---|---|---|
| CO1 | Understand and apply key principles of image formation and photometric processing to analyze images and digital camera functions. | L2 | Understand |
| CO2 | Implement and evaluate feature extraction techniques and morphological operations for conditioning, labeling, and matching objects in images | L5 | Evaluate |
| CO3 | Apply segmentation and detection algorithms to identify and isolate objects and regions of interest in complex images. | L3 | Apply |
| CO4 | Demonstrate competence in advanced feature-based alignment, 3D object recognition, and scene understanding for real-world applications in computer vision. | L6 | Create |

# Computer Vision(RCP23CCPE613)
# Course Contents

## Unit-I Introduction — 04 Hrs.

What is computer Vision, Image Formation: Geometric Primitives, 2D transformation, 3 D transformation, 3D rotation, 3D to 2D Projection, Lens Distortion

**Photometric Image Formation:** Lighting, reflectance and shading

**The digital camera:** sampling and quantizition.

## Unit-II Recognition Methodology — 06 Hrs.

Conditioning, Labeling, Grouping, Extracting, Matching.

**Morphological Image Processing:** Introduction, Dilation, Erosion, Opening, Closing,Hit-or-Miss transformation, Morphological algorithm operations on binary images, Morphological algorithm operations on gray-scale images, Thinning, Thickining , Region growing, region shrinking.

## Unit-III Feature Extraction and Model Fitting: — 08 Hrs.

Edges - Canny, LOG, DOG, Line detectors (Hough Transform), Corners – Harris and Hessian Affine, Orientation Histogram, SIFT, SURF, HOG, GLOH, Deformation, RANSAC, Scale-Space Analysis-Image Pyramids and Gaussian derivative filters, Gabor Filters and DWT.

## Unit-IV Object Segmentation and Detection: — 08 Hrs.

Active contours, Split and merge, Mean shift and mode finding, Normalized cuts, Graph cuts and energy-based methods, Application: Medical image segmentation.

## Unit-V Feature-based alignment — 06 Hrs.

2D and 3D feature-based alignment, Pose estimation, Geometric intrinsic calibration, Calibration patterns, Vanishing points, Application: Single view metrology, Rotational motion, Radial distortion.

## Unit-VI Recognition: — 07 Hrs.

Object detection, Face detection, Pedestrian detection, Face recognition, 3D shape models, Application: Personal photo collections, Instance recognition, Category recognition, Context and scene understanding.

## Computer Vision Laboratory(RCP23CLPE613)

### List of Laboratory Experiments

**Suggested Experiments:(Any 08)**

1. Image Acquisition and Display
2. Image Transformations
3. Image Denoising and enhancement techniques
4. Feature Detection using OpenCV- Corner, Edge, Pyramid
5. Object Detection
6. Object Tracking
7. Pattern Recognition
8. Face Recognition
9. Perform medical image segmentation
10. Implementation of Image smoothing/ Image sharpening
11. Mini Project Based Learning
12. Research Article Review

Any other experiment based on syllabus may be included which would help the learner to understand topic/concept.

**Oral examination will be based on the entire syllabus including, the practicals performed during laboratory sessions.**

## Text Books:


1. Richard Szeliski, 'Computer Vision: Algorithms and Applications", 2nd Edition, Springer Verlag London Limited, 2022.
2. Rafael C. Gonzalez and Richard E. Woods, 'Digital Image Processing", Pearson, 4th Edition, 2017.


## Reference Books:


1. David Forsyth, Jean Ponce, "Computer Vision: A Modern Approach, Pearson Education", 2nd Edition, 2015.

2. Katsushi Ikeuchi, "Computer Vision: A Reference Guide", 2nd edition, Springer Publishing, 2021.

3. Christopher M. Bishop, "Pattern Recognition and Machine Learning", Springer, 2006.

4. M. Sonka, V. Hlavac, and R. Boyle, "Image Processing, Analysis, and Machine Vision, Thomson India Edition, 4th Edition, 2017.

| Program: Computer Engineering | T.Y. B.Tech. | Semester: VI |
|---|---|---|
| Advanced Network Design (RCP23CCPE621) | | |
| Advanced Network Design Laboratory (RCP23CLPE621) | | |

**Prerequisites:** Data Structures, Computer Networks, Operating Systems.

## Course Objective(s):

1. To develop a comprehensive understanding of advanced network design principles and their strategic implications.
2. To acquire proficiency in the practical implementation of network designs, including physical infrastructure, configurations, and lifecycle management.
3. To apply analytical skills to assess network data traffic, security threats, and performance metrics for informed decision-making.

## Course Outcomes:

On completion of the course, the learner will be able to:

| CO | Course Outcomes | Blooms Level | Blooms Description |
|---|---|---|---|
| CO1 | Analyze the need for advanced networks and standards in various scenarios, evaluating specific design requirements for topology, cabling, and physical component placement. | L4 | Analyze |
| CO2 | Develop frameworks and standards for the implementation of network designs, incorporating best practices in configuration management. Also apply human factor concepts to enhance usability and troubleshoot network issues. | L6 | Create |
| CO3 | Integrate IPv6 in network design, design and implement security systems, and analyze real-world examples of advanced network design. Also compare performance metrics in Internet routing and VoIP. | L6 | Create |



# Advanced Network Design (RCP23CCPE621)
# Course Contents

## Unit-I — 05 Hrs.

**Introduction to Advanced Network Design:** Overview of Advanced Network Design Principles: Definition of advanced network design, Importance of strategic network planning. Review of Basic Networking Concepts: OSI Model revisited, TCP/IP fundamentals, Network addressing and subnetting. Emerging Trends in Networking: Software-Defined Networking (SDN), Network Function Virtualization (NFV), Internet of Things (IoT) in networking.

## Unit-II — 09 Hrs.

**Physical Network Design and Infrastructure:** Physical Network Design: Topology design considerations, Cabling and physical component placement, Redundancy and fault tolerance in physical design.Advanced Router Configuration: In-depth configuration of routers, Routing protocols (EIGRP, OSPF, BGP), Router optimization and scalability.

## Unit-III — 06 Hrs.

**Configuring and Managing the Network Infrastructure:** Network Configuration Best Practices: Implementation of network designs, Configuration management and version control Network Lifecycle Management: Maintenance and troubleshooting strategies, Network monitoring and performance optimization.

## Unit-IV — 06 Hrs.

**Analyzing Network Data Traffic:** Traffic Analysis Fundamentals: Packet capture and analysis tools, Understanding network protocols, Quality of Service (QoS) Implementation, QoS requirements and strategies, Traffic classification and shaping

## Unit-V — 06 Hrs.

**Network Security and IPv6:** Threats and Vulnerabilities in Networks: Common network security threats, Vulnerability assessment and risk analysis Firewall and Intrusion Detection/Prevention Systems, Designing and placing firewalls strategically, Intrusion detection and prevention strategies Virtual Private Networks (VPNs) and IPv6: Implementing VPNs for security, Integration of IPv6 in network design

## Unit-VI — 07 Hrs.

**Internet Routing and VOIP:** Internet Routing with BGP: BGP essentials and best practices, Internet routing considerations Voice over IP (VoIP) Basics: Introduction to VoIP technologies, design

considerations for VoIP in networks Case Studies and Best Practices: Examining real-world examples of advanced network design, best practices for implementing advanced network solutions

## Advanced Network Design Laboratory (RCP23CLPE621)

### List of Laboratory Experiments

**Suggested Experiments:(Any 8)**

1. Implement advanced network design principles through case study analysis.
2. Design a physical network layout using simulation tools like Cisco Packet Tracer or GNS3.
3. Configure routers for EIGRP, OSPF, and BGP, optimizing routing tables.
4. Set up a network infrastructure adhering to industry best practices.
5. Develop troubleshooting skills and perform routine maintenance tasks.
6. Analyze network traffic using tools like Wireshark.
7. Implement Quality of Service (QoS) strategies for network traffic.
8. Simulate and analyze common network security threats using tools like Metasploit.
9. Design and implement effective firewall and intrusion detection/prevention systems.
10. Configure and deploy Virtual Private Networks (VPNs) for secure communication

Any other experiment based on syllabus may be included which would help the learner to understand topic/concept.

**Oral examination will be based on the entire syllabus including, the practicals performed during laboratory sessions.**

## Text Books:


1. Olivier Bonaventure, "Computer Networking: Principles, Protocols and Practice," No Starch Press, 2013.
2. Kevin Dooley, "Designing Large Scale LANs," O'Reilly Media, 2011.
3. Mani Subramanian, "Network Management: Principles and Practice," 2nd Edition, Addison-Wesley, 2012.
4. Ilya Grigorik, "High-Performance Browser Networking," O'Reilly Media, 2013.
5. William Stallings, "Network Security Essentials," Pearson,2016.

6. William A. Flanagan, "VoIP and Unified Communications: Internet Telephony and the Future Voice Network," Wiley, 2012.

## Reference Books:

1. Laura Chappell, "Wireshark Network Analysis," Protocol Analysis Institute,2012.

2. Silvia Hagen, "IPv6 Essentials," O'Reilly Media,2006.

3. Iljitsch Van Beijnum, "BGP: Building Reliable Networks with the Border Gateway Protocol," 2nd Edition, O'Reilly Media, 2002.

4. Tim Szigeti, "End-to-End QoS Network Design," Cisco Press, 2013.

5. Jeffrey S. Beasley and Piyasat Nilkaew, "A Practical Guide to Advanced Networking," Pearson, 2017.

6. Gary A. Donahue, "Network Warrior," O'Reilly Media, 2011.

| Program: Computer Engineering | T.Y. B.Tech. | Semester: V |
|---|---|---|
| Distributed Databases (RCP23CCPE622) | | |
| Distributed Databases Laboratory(RCP23CLPE622) | | |

**Prerequisites:** Database Management Systems, Advanced Database Management Systems.

## Course Objectives:

1. To provide students with a comprehensive understanding of the design and management of distributed databases
2. To familiarize students with the technical challenges and solutions in distributed database systems

## Course Outcomes:

On completion of the course, the learner will be able to:

| COs | Course Outcomes | Blooms Level | Blooms Description |
|---|---|---|---|
| CO1 | Design distributed databases, including data fragmentation and query processing, while optimizing for performance for real life applications. | L6 | Create |
| CO2 | To apply distributed transaction management protocols to ensure consistency and reliability in distributed systems under various failure conditions. | L3 | Apply |

# Distributed Databases (RCP23CCPE622)
# Course Contents

## Unit-I Concept and Overview of Distributed Database System

**05 Hrs.**

What is Distributed Database System (DDBS), Features of DDBS, promises of DDBS, Design issue in DDBS, Distributed DBMS architecture: Client/server System, Peer¬to-¬Peer, Mutli¬Database system, Types of Distributed Databases: Homogeneous vs. Heterogeneous

## Unit-II Distributed Database Design **07 Hrs.**

Distributed database design concept, Objective of Data Distribution, Data Fragmentation (Horizontal, Vertical and Hybrid), Fragment Allocation, Transparencies in Distributed Database Design, Data Replication Concept and Techniques

## Unit-III Overview of Query Processing **08 Hrs.**

Query Processing Problem, Objectives of Query Processing, Complexity of Relational Algebra Operations, Characterization of Query Processors, Layers of Query Processing.

**Query Decomposition and Data Localization:** Query Decomposition- Normalization, Analysis, Elimination of Redundancy, Rewriting, Localization concept.

## Unit-IV Distributed Query Optimization **08 Hrs.**

Query Optimization, Join Ordering in Distributed Queries- Join Ordering, Semijoin Based Algorithms, Join versus Semijoin, Distributed Query Optimization – Dynamic Approach, Static Approach, Semijoin-based Approach, Hybrid Approach.

## Unit-V Distributed Transaction, Concurrency control and Recovery

**07 Hrs.**

Distributed Transaction Management - Definition of a Transaction, Properties of Transactions, Model for Transaction management.

Distributed Concurrency control: Locking-Based Concurrency Control Algorithms - Distributed 2PL, Timestamp-Based Concurrency Control Algorithms.

Distributed Deadlock and Recovery: Distributed Deadlock prevention, avoidance, detection and recovery, Two-Phase and Three-Phase Commit Protocol.

## Unit-VI Distributed Database Security and Emerging Trends

**04 Hrs.**

Security Challenges in Distributed Databases, NoSQL distributed databases, NewSQL, and cloud-native databases, Database as a service (DBaaS).

## Distributed Databases Laboratory(RCP23CLPE622)

### List of Laboratory Experiments

**Suggested Experiments:(Any 08)**

1. Creation of centralized homogeneous database.
2. Perform Horizontal fragmentation and allocation in DDBS design.
3. Perform Vertical fragmentation and allocation in DDBS design.
4. Perform Hybrid fragmentation and allocation in DDBS design.
5. Implementation of Replication transparency.
6. Implementation of Query monitor (QEP- Query Execution Plan, Query Statistics)
7. Implementation of query optimization.
8. Implementations of deadlock detection in DDBS. (Use of table lock/unlock)
9. Simulation of two phase / three phases commit protocol.
10. Case study on Distributed Database Security and Emerging Trends.

Any other experiment based on syllabus may be included which would help the learner to understand topic/concept.

**Oral examination will be based on the entire syllabus including, the practicals performed during laboratory sessions.**

## Text Books:


1. Chhanda Ray, "Distributed Database System", Pearson Education India, 2009.
2. Abraham Silberschatz, Henry F. Korth, Sudarshan, "Database System Concepts", 7th Edition, Mc Graw Hill, 2021.
3. Seed K. Rahimi and Frank S. Haug, "Distributed Database Management System", Wiley India, 2015.

## Reference Books:


1. M. Tamer Ozsu, Patrick Valduriez, "Principles of Distributed Database", Pearson Education India, 2020.

2. Elmasri and Navathe, "Fundamentals of Database Systems", 7th Edition, Pearson education, 2016.

| Program: Computer Engineering | T.Y. B.Tech. | Semester: VI |
|---|---|---|
| High Performance Computing (RCP23CCPE623) | | |
| High Performance Computing Laboratory (RCP23CLPE623) | | |

**Prerequisite:** IoT-Centric Processor Organization and Architecture.

## Course Objectives:

1. To learn concepts of parallel processing as it pertains to high-performance computing.
2. To design, develop and analyze parallel programs on high-performance computing resources using parallel programming paradigm.

## Course Outcomes:

On completion of the course, the learner will be able to:

| COs | Course Outcomes | Blooms Level | Blooms Description |
|---|---|---|---|
| CO1 | Understand different parallel processing approaches and platforms involved in achieving High Performance Computing | L2 | Understand |
| CO2 | Explore Programming for High Performance Computing. | L3 | Apply |
| CO3 | Understand the principles of Grid and Cloud Computing with practical examples and applications. | L2 | Understand |
| CO4 | Analyze the performance measures in high performance computing. | L4 | Analyze |
| CO5 | Discover the advanced topic in GPU including libraries and framework. | L4 | Analyze |

# High Performance Computing (RCP23CCPE623) Course Contents

## Unit-I Introduction to Parallel Processing — 04 Hrs.

Parallel processing, Levels of Parallelism, Models: SIMD, MIMD, SIMT, SPMD, Data Flow Models, Demand-driven Computation. Loosely coupled and tightly coupled. Parallel Architecture: Interconnection network, processor Array, Multiprocessor, Challenges in Parallel Computing, Performance Metrics, Distributed vs. Parallel architectures.

## Unit-II Introduction to High Performance Computing — 06 Hrs.

Principles of HPC, HPC Architectures, HPC vs Parallel Processing, Data partitioning Techniques: Block, cyclic, and block-cyclic partitioning, Domain Decomposition: Spatial, temporal, and functional decomposition, Load balancing, Case Study: Partitioning strategies for matrix multiplication. Communication Models: Shared memory vs. message passing. Point-to-Point Communication: Send/Receive operations in MPI. Collective Communication: Broadcast, scatter, gather, and reduction operations in MPI (MPI-Reduce).

## Unit-III GPU and CUDA Programming — 08 Hrs.

Overview of GPU, evolution of GPU, CPU vs. GPU, overview of CUDA: Features, Benefits, Architecture. Programming Model CUDA: Kernels and kernel launches, Thread and block indexing, CUDA Memory Management: Memory Hierarchy and Memory Management, Case Studies: simulation, data analytics and machine learning.

## Unit-IV Grid and Cloud Computing — 07 Hrs.

Data & Computational Grids, Grid Architectures and its relation to various Distributed Technologies, Examples of The Grid Computing, Cloud Computing, High Performance Cloud Computing, Cloud Tensor Processing Units (TPUs). Case studies: Analyzing impact of TPUs.

## Unit-V Performance Optimization — 08 Hrs.

Speedup, Efficiency and Scalability, Amdahl's Law, Gustafson's Law, Weak vs. Strong Scaling, Performance Bottlenecks, Data Races and Determinism, Data Race Avoidance, Profiling and performance analysis tools for GPUs, Techniques for optimizing GPU performance (warp divergence, loop unrolling, vectorization), Memory bandwidth optimization techniques, Advanced GPU programming concepts (shared memory atomics, warp shuffling). Case Studies: Scientific Computing with CUDA/Real-life application.



## Unit-VI Advanced Topics in GPU: 06 Hrs.

Introduction to GPU accelerated libraries (cuBLAS, cuDNN, cuGraph), GPU computing frameworks (TensorFlow, PyTorch) and their integration with GPUs, Introduction to GPU clusters and distributed GPU computing, Cluster Setup & its Advantages. Case studies: Realworld applications of GPU computing.

## High Performance Computing Laboratory (RCP23CLPE623)

### List of Laboratory Experiments

**Suggested Experiments:(Any 08)**

1. Set up the CUDA environment, install the CUDA Toolkit, and write a basic CUDA program to understand the CUDA development environment.
2. Implement vector addition using CUDA to introduce students to parallelism, thread management, and memory allocation in GPU programming.
3. Develop a CUDA program for matrix multiplication to understand parallelism and optimization techniques in GPU computing.
4. Apply CUDA for image processing tasks, like blurring and edge detection, to learn how to process images efficiently using GPU parallelism.
5. Implement parallel reduction operations (e.g., sum, min, max) to grasp the concept of efficient parallel reduction.
6. Explore parallel sorting algorithms using CUDA, comparing their performance with CPU based sorting and optimizing CUDA sorting.
7. Employ CUDA for solving real-world problems to understand the power of GPU parallelism.
8. Experiment with CUDA to implement concurrent data structures using locks and atomic operations to learn how to manage data concurrently.
9. Optimize the reduction step in machine learning algorithms using CUDA, focusing on techniques for efficient large-scale data processing.
10. Integrate CUDA-accelerated code with data science frameworks like TensorFlow or PyTorch to develop and run GPU-accelerated machine learning models for practical applications.
11. Perform the Log Analysis-Based Resource and Execution Time Improvement

Any other experiment based on syllabus may be included which would help the learner to understand topic/concept.

**Oral examination will be based on the entire syllabus including, the practicals performed during laboratory sessions.**

## Text Books:


1. Edson Borin, Lúcia Maria A. Drummond, Jean-Luc Gaudiot, Alba Melo, Maicon Melo Alves, Philippe Olivier Alexandre Navaux, "High Performance Computing in Clouds: Moving HPC Applications to a Scalable and Cost Effective Environment", Springer, ISBN-13 978-3031297687, 2023.

2. Alexander Heifetz, "High Performance Computing for Drug Discovery and Biomedicine", Springer Nature, ISBN, 1071634496, 9781071634493, 2023.

3. Richard Ansorge, "Programming in Parallel with CUDA", Cambridge University Press, ISBN-13 978-1108479530, 2022.

4. Robert Robey, Yuliana Zamora, "Parallel and High Performance Computing", Manning publisher, ISBN-13 978-1617296468, 2021.

5. Sergey A. Babkin, "The Practice of Parallel Programming", CreateSpace Publisher ISBN-13:978-1451536614, Online Edition 2021.

6. Dr. Brian Tuomanen, "Hands-On GPU Programming with Python and CUDA", Packt Publishing, ISBN-13 978-1788993913, 2018.


## Reference Books:


1. David B. Kirk and Wenmei W. Hwu, Morgan Kaufmann, "Programming Massively Parallel Processors: A Handson Approach", 4th Edition, 2022.

2. Jason Sanders and Edward Kandrot, Addison-Wesley, "CUDA by Example: An Introduction to General-Purpose GPU Programming", Addison-Wesley, 1st Edition, 2010.

3. Georg Hager, Gerhard Wellein, "Introduction to High Performance computing for Scientist and Engineers", CRC press, 2019.

| Program: Computer Engineering | T.Y. B.Tech. | Semester: VI |
|---|---|---|
| Security in IoT Ecosystem (RCP23CCMD601) | | |
| Security in IoT Ecosystem Laboratory (RCP23CLMD601) | | |

**Prerequisites:** Computer Networks with IoT Fundamentals, IoT-Centric Processor Organization and Architecture.

## Course Objective(s):

1. To understand and apply classic cryptographic techniques and modular arithmetic principles essential for modern cryptographic systems.
2. To gain knowledge of modern cryptographic algorithms, including symmetric and asymmetric encryption, and their practical applications in ensuring data confidentiality.
3. To explore mechanisms for ensuring integrity and availability in cryptographic systems, focusing on hashing, message authentication, and digital signatures.
4. To analyze network security protocols and IoT-specific authentication mechanisms that enhance security in connected environments.

## Course Outcomes:

On completion of the course, the learner will be able to:

| CO | Course Outcomes | Blooms Level | Blooms Description |
|---|---|---|---|
| CO1 | Demonstrate proficiency in applying classic cryptographic techniques and modular arithmetic to solve cryptographic problems. | L3 | Apply |
| CO2 | Implement and evaluate modern symmetric and asymmetric encryption algorithms to secure data transmission in various applications. | L5 | Evaluate |
| CO3 | Utilize secure hash functions and digital signatures to ensure data integrity and authenticity in cryptographic communications. | L3 | Apply |
| CO4 | Design and assess secure network architectures for IoT environments, implementing robust authentication protocols to safeguard user and device identities.. | L6 | Create |

# Security in IoT Ecosystem (RCP23CCMD601)
# Course Contents

## Unit-I — 04 Hrs.

**Modular Arithmetic and Classic Cryptographic Techniques:**

**Classic Cryptographic Techniques:** Caesar Cipher, Railfence Cipher, Vigenère Cipher, and Affine Cipher, One-Time Pad and Enigma Machine, Transposition Ciphers.

**Modular Arithmetic and Number Theory:** Modular exponentiation, Fermat's Little Theorem and applications, Chinese Remainder Theorem, Euler's Theorem, Euclid's Algorithm, Groups , Rings and Fields..

## Unit-II — 05 Hrs.

**Modern Cryptographic Techniques:**

**Fundamentals:** Block Ciphers and Stream Ciphers, Block Cipher Modes of Operations.

**Symmetric Encryption Algorithms:** Historical Symmetric Encryption Techniques (DES), Transition from DES to 3DES, Advanced Encryption Standard, ChaCha20, Lightweight Cryptography.

**Asymmetric Encryption Algorithms:** RSA Algorithm, Elliptic Curve Cryptography, Curve 25519.

## Unit-III — 04 Hrs.

**Digital Trust Mechanisms:**

**Secure Hash Algorithms:** Cryptographic Hashes, SHA-256

**Message Authentication Codes (MAC) and HMAC:** MAC, HMAC, CMAC

**Digital Signatures:** DSA, ECDSA

**Fault Tolerance and High Availability in Cryptographic Systems:** Fault Tolerance in Cryptography, Redundancy Mechanisms, Failover Systems

## Unit-IV — 04 Hrs.

**Network Security:**

**Secure Network Protocols:** SSL/TLS, IPSec, HTTPS

**Firewalls and Intrusion Detection Systems (IDS/IPS):** Firewalls,Intrusion Detection Systems (IDS), Prevention Systems (IPS)

**VPNs and Zero Trust Architecture:** Threat Detection with Machine Learning and AI

## Unit-V — 04 Hrs.

**IOT Network Authentication:**

**IoT Overview and Architecture:** IoT ecosystem and architecture components, unique securit

challenges in IoT environments

**Symmetric Key-Based Authentication in IoT:** Lightweight encryption methods for Wireless Sensor Networks (WSNs)

**Asymmetric Key-Based Authentication Protocols (ECC in IoT):** ECC for secure key exchange in resource-constrained environments, Importance of ECC for energy efficiency in IoT devices.

**Quantum-Resistant Solutions for IoT:** Lattice-based cryptographic protocols for quantum resistance, Application of lattice-based algorithms in IoT for future-proof security

## Unit-VI

05 Hrs.

**IOT User and Device Level Authentication:**

**Mutual Authentication Protocols:** Ensuring two-way authentication in IoT devices, Prevention of unauthorized access through mutual verification.

**Biometric-Based Robust Access Control Models:** Use of biometrics (fingerprints, facial recognition) for secure access, Challenges and advantages of biometric authentication in IoT.

**Gadget-Free Authentication:** Context-aware and password-less authentication methods, Use of geolocation, user behavior, and ambient data for seamless access.

**Physically Unclonable Function (PUF)-Based Authentication:** PUF for tamper-resistant security, Applications of PUF in IoT.

**Hardware-Based Encryption and Secure Elements:** Secure hardware modules (TPM, HSM) for encrypted data storage, Key management and tamper-proof elements for enhanced device security.

## Security in IoT Ecosystem Laboratory (RCP23CLMD601)

### List of Laboratory Experiments

**Suggested Experiments:(Any 8)**

1. **Modular Arithmetic Operations:**

   a) Create a program to perform modular exponentiation and demonstrate its application in Fermat's Little Theorem.

   b) Solve a set of problems using the Chinese Remainder Theorem and demonstrate how to recover the original number from modular equations.

2. **Implementation of Classic Ciphers:**

   a) Write a program to implement and encrypt/decrypt messages using the Caesar Cipher, Rail-fence Cipher, and Vigenère Cipher.

   b) Analyze the strengths and weaknesses of each cipher by attempting frequency analysis on sample ciphertexts.



3. **Symmetric Encryption Implementation:**

   a) Implement DES and AES encryption algorithms in a programming language of choice (e.g., Python). Test the performance and security of both algorithms using various input sizes.

   b) Explore the block cipher modes of operation (ECB, CBC, CFB) by implementing them and analyzing their impact on the security of the encrypted data.

4. **Asymmetric Encryption Experiment:**

   a) Write a program to implement the RSA algorithm and conduct key generation, encryption, and decryption. Evaluate its security based on key sizes.

   b) Implement Elliptic Curve Cryptography (ECC) for key exchange and demonstrate its use in a simple messaging application.

5. **Hash Function Analysis:**

   Implement SHA-256 and create a program to generate hash values for various input data. Analyze collision resistance by testing for any two different inputs that produce the same hash.

6. **Digital Signature Generation and Verification:**

   Implement the Digital Signature Algorithm (DSA) and create a workflow for signing a message and verifying the signature. Discuss potential vulnerabilities and how to mitigate them.

7. **Secure Network Protocols Testing:**

   a) Set up a secure communication channel using SSL/TLS and demonstrate a secure client-server application. Analyze the security features provided by the protocol.

   b) Configure and test a VPN solution, documenting the steps for secure remote access and discussing its advantages and limitations.

8. **Intrusion Detection System Setup:**

   a) Install and configure an Intrusion Detection System (IDS) such as Snort. Simulate different attack scenarios and observe how the IDS detects and responds to them.

   b) Create a simple firewall configuration to control incoming and outgoing traffic, testing its effectiveness against common network attacks.

9. **IoT Device Security Simulation:**

   a) Simulate a lightweight encryption method for securing data transmission in a Wireless Sensor Network (WSN) and evaluate its performance in terms of energy consumption and security.

   b) Implement an asymmetric key-based authentication protocol using ECC in an IoT simulation environment. Analyze the efficiency of key exchange in resource-constrained devices.

10. **Quantum-Resistant Protocol Evaluation:**

    Research and implement a basic lattice-based cryptographic protocol. Simulate its application in an IoT scenario and evaluate its effectiveness against potential quantum attacks.

11. **Mutual Authentication Protocol Implementation:**

    a) Design and implement a mutual authentication protocol between two IoT devices, documenting the process and evaluating its security effectiveness.

    b) Test the protocol against various attack vectors (e.g., replay attacks, man-in-the-middle attacks) to evaluate its robustness.

12. **Biometric Authentication System Development:**

    Create a simple biometric authentication system using facial recognition or fingerprint scanning. Test its effectiveness and discuss the challenges faced during implementation.

Any other experiment based on syllabus may be included which would help the learner to understand topic/concept.

**Oral examination will be based on the entire syllabus including, the practicals performed during laboratory sessions.**

## Text Books:


1. William Stallings, "Cryptography and Network Security: Principles and Practice", Pearson 2020, 8th Edition.

2. Behrouz A. Forouzan, "Cryptography and Network Security", McGraw-Hill Education, 2017, 4th Edition.

3. Madhusanka Liyanage, An Braeken, Pardeep Kumar, Mika Ylianttila, "IoT Security", Wiley, 2020


## Reference Books:


1. Brucer Schneier, "Applied Cryptography: Porotocols, Algorithms and Source Code in C", Wiley, 2015,

2. Charles Pfleeger, Shari Lawrence Pfleeger & Jonathan Margulies, "Security in Computing", 5th Edition, Prentice Hall, 2018.

| Program: Computer Engineering | T.Y. B.Tech. | Semester: VI |
|---|---|---|
| Project Stage-I (RCP23IPEL601) | | |

## Course Objectives:

1. To understand the basic concepts and principles of project development.
2. To formulate/identify the problem statement.
3. To implement the solution as per the problem statement.
4. To develop the team building, writing, logical reasoning and management skills.
5. To provide the connections between the designs and concepts across different disciplinary boundaries.
6. To encourage students to become independent personnel, critical thinkers and lifelong learners.

## Course Outcomes:

On completion of the course, the learner will be able to:

| COs | Course Outcomes | Blooms Level | Blooms Description |
|---|---|---|---|
| CO1 | To identify the problem statement and produce solution of the problem considering cultural, social, environmental and economic factors using appropriate tool and method. | L4 | Analyze |
| CO2 | Demonstrate project based learning that allows students to transfer existing ideas into new applications. | L2 | Understand |
| CO3 | Develop an ability to work in teams and manage to conduct the project development activity. | L3 | Apply |
| CO4 | Integrate different perspectives from relevant disciplines which help them to get internships, jobs and admission for higher studies. | L3 | Apply |
| CO5 | Present the project development in the form of technical writing, understand what constitutes to plagiarism and how to use proper referencing styles. | L2 | Understand |

**Syllabus:**

Domain knowledge (any beyond) needed from the following areas for the effective implementation of the project:

Database Management System, Networking and Internet of Things, Embedded Systems, Data science and Big data, Web and Application Development, Robotics, AI and Machine Learning, etc.

The above areas can be updated based on the technological innovations and development needed for specific project.

**Guidelines:** The main purpose of this activity is to improve the students' technical skills, communication skills by integrating writing, presentation and teamwork opportunities.

- Each student shall work on project stage-I as approved by the departmental committee, a group of 03 to 04 students (max allowed: 5 students in extraordinary cases, subject to the approval of the departmental committee) shall be allotted for each Project stage-I. The departmental committee shall include Head of Department, project coordinator and guide(s).
- Each group shall submit at least 3 topics for the Project stage-I. The departmental committee shall finalize one topic for every group.
- Each group is required to maintain separate log book for documenting various activities of the project stage-I (Refer Table 4).
- Each group will be reviewed twice in a semester and marks will be allotted based on the various points mentioned in the evaluation scheme.
- In the first review of this semester, each group is expected to complete 30% of project stage-I.
- In the second review of this semester, each group is expected to complete 50% of project stage-I.
- Interaction with alumni mentor will also be appreciated for the improvement of project stage-I.

**Assessment Criteria:**

- At the end of the semester, after confirmation by the project guide, each project group will submit project completion report in prescribed format for assessment to the departmental committee.
- Assessment of the project (at the end of the semester) will be done by the departmental committee.
- Oral examination shall be conducted by Internal and External examiners. Students have to give presentation and demonstration based on their project stage-I.

**Prescribed project report guidelines:**

Size of report shall be of minimum 30 pages (excluding cover and front pages). Project stage-I report should include appropriate content for:

- **Abstract**
- **Introduction**
  - Backgroumd
  - Motivation
  - Problem Statement
  - Objectives
  - Scope
- **Literature Survey**
  - Review of Existing System(s)
  - Limitations of Existing System(s)
- **Proposed System**
  - Analysis/Framework/ Algorithm
  - Details of H/W and S/W required
  - Design details
  - Methodology (your approach to solve problem)
- **Implementation Plan for Project Stage-II**
- **Conclusion**
- **References**

**Assessment criteria for the departmental committee for Continuous Assessment:**

Guide will monitor weekly progress and marks allocation will be as per Table 5.

**Assessment criteria for the departmental committee for End Semester Exam:**

Departmental committee will evaluate project as per Table 6.

Table 4: Log Book Format

| Sr | Week (Start Date:End Date) | Work Done | Sign of Guide | Sign of Coordinator |
|---|---|---|---|---|
| 1 | | | | |
| 2 | | | | |

Table 5: Contimous Assessment Sheet

| Sr | P.R.N. | Name of Student | Student Attendance (5) | Log Book Maintenance (5) | Literature Review (5) | Depth of Understanding (5) | Report (5) | Total (25) |
|---|---|---|---|---|---|---|---|---|
| | | | 5 | 5 | 5 | 5 | 5 | 25 |

Table 6: Evaluation Sheet

| Sr | P.R.N. | Name of Student | Project Stage - I Selection (5) | Design/ Simulation/ Logic (5) | PCB/ Hardware/ Programming (5) | Result Verification (5) | Presentation (5) | Total (25) |
|---|---|---|---|---|---|---|---|---|
| | | | 5 | 5 | 5 | 5 | 5 | 25 |

| Program: Computer Engineering | T.Y. B.Tech. | Semester: VI |
|---|---|---|
| Environmental Science Tutorial (RCP23ITHSX06) | | |

**Prerequisite:** Interest in Environment and its impact on Human.

## Course Objective(s):

1. To familiarise students with environment related issues such as depleting resources, pollution, ecological problems and the renewable energy scenario.
2. To give overview of Green Technology options.

## Course Outcomes:

On completion of the course, the learner will be able to:

| CO | Course Outcomes | Blooms Level | Blooms Description |
|---|---|---|---|
| CO1 | Understand how human activities affect environment . | L2 | Understand |
| CO2 | Understand the various technology options that can make a difference. | L2 | Understand |

# Environmental Science Tutorial (RCP23ITHSX06) Course Contents

## Unit-I

01 Hrs.

**Air Pollution:** Air Quality Index, Case study on Smog

## Unit-II

01 Hrs.

**Water Pollution:** Presentation on Water Pollution (Industrial, Sewage, etc.) explaining any specific case.

## Unit-III

01 Hrs.

**Noise Pollution :**Decibel limits for hospital, library, silence zone, List effects of noise pollution on human health, Measure decibel level in college library, canteen, classroom

## Unit-IV

01 Hrs.

**Biodiversity loss :** Case study on effect of pollution on biodiversity loss.

## Unit-V

01 Hrs.

**Deforestation :** Debate for and against "To promote Economic growth Deforestation is required."

## Unit-VI

01 Hrs.

**Renewable Energy sources :** Presentation on different Renewable Energy Technologies

## Unit-VII

01 Hrs.

**Climate change:** Report on major Impact of Global warming on Environment giving real examples.

## Unit-VIII

01 Hrs.

**Green Technology :** Advantages and Examples of Green Building for Sustainable development, Sustainable Software Design, Data Center Energy Efficiency, Thin-Client and Energy Efficiency.

## Text Books:


1. R. Rajagopalan, "Environmental Studies From Crisis to Cure".
2. Erach Bharucha, "Textbook of Environmental Studies For Undergraduate Courses".
3. Mohammad Dastbaz, Colin Pattinson, Babak Akhgar, Morgan and Kaufman, Elsevier., "Green Information Technology A Sustainable Approach".


## Reference Books:


1. Paulina Golinska, Marek Fortsch, Jorge Marx-Gómez, "Information Technologies in Environmental Engineering", New Trends and Challenges, Springer, 2011.

| Program: Computer Engineering | T.Y. B.Tech. | Semester: VI |
|---|---|---|
| Employability Skill Development Program-I (RCP23CLSC601) | | |

## Course Objective(s):

1. Apply basic programming constructs such as variables, conditionals, loops, and recursion to solve simple problems.
2. Implement and manipulate core data structures like arrays, strings, sets, and maps in real-world scenarios.
3. Design and analyze basic algorithms using brute force, sorting, and observation techniques.
4. Solve problems using algorithmic strategies such as greedy, constructive, and backtracking approaches.
5. Utilize mathematical concepts like GCD, modular arithmetic, and binary systems in algorithm development.

## Course Outcomes:

On completion of the course, the learner will be able to:

| CO | Course Outcomes | Blooms Level | Blooms Description |
|---|---|---|---|
| CO1 | Apply fundamental programming concepts such as variables, control structures, loops, and recursion to solve basic computational problems. | L3 | Apply |
| CO2 | Demonstrate effective use of basic data structures like arrays, strings, sets, and maps to store, retrieve, and manipulate data. | L3 | Apply |
| CO3 | Analyze and implement simple algorithms using sorting, brute force, and ad-hoc techniques to solve real-world problems. | L4 | Analyze |
| CO4 | Develop algorithmic solutions using advanced techniques like greedy methods, constructive logic, and backtracking. | L6 | Create |
| CO5 | Evaluate and apply mathematical concepts such as number theory, modular arithmetic, and binary operations in designing efficient algorithm. | L5 | Evaluate |

# Employability Skill Development Program-I (RCP23CLSC601) Course Contents

## Unit-I

05 Hrs.

**Programming Basics:**

- Basic Programming Concepts
  - Variables, Data Types, Expressions
  - Input/Output
  - Inbuilt Functions (Python and general)
  - Control Flow: Conditional Statements (if, else, elif)
  - Loops (for, while) — including Python-specific loops
  - Recursion basics

## Unit-II

05 Hrs.

**Data Structures Fundamentals:**

- Arrays: 1D Arrays, Frequency Arrays
- Strings
- Sets
- Maps (Dictionaries)
- Groupings and Subsequences

## Unit-III

05 Hrs.

**Algorithms – Foundations:**

- Brute Force Approaches
- Sorting Algorithms
- Simple Algorithms & Implementation Practice
- Observation Techniques
- Ad-hoc Problem Solving

## Unit-IV

**Algorithmic Techniques:** 05 Hrs.

- Greedy Algorithms
- Constructive Algorithms
- Ad-hoc constructive techniques
- Combined with observation strategies
- Recursion & Backtracking
- Suffix Sum and Prefix Sum techniques

## Unit-V

**Mathematics for Programming:** 06 Hrs.

- Basic Math & Arithmetic
- Number Theory
- Greatest Common Divisor (GCD)
- Modular Arithmetic
- Divisibility Rules
- Integer Division
- Number Systems & Binary
- Geometry basics
- Recurrence Relations

## Reference Books:


1. John Zelle, "Python Programming: An Introduction to Computer Science", 3rd Edition, Franklin, Beedle & Associates Inc., 2016.
2. Michael T. Goodrich, Roberto Tamassia, and Michael H. Goldwasser, "Data Structures and Algorithms in Python", 1st Edition, Wiley, 2013.
3. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein, "Introduction to Algorithms", 4th Edition, The MIT Press, 2022.
4. Kenneth H. Rosen, "Discrete Mathematics and Its Applications", 7th Edition, McGraw-Hill Education, 2012.