



# AVANTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

(Autonomous)

(Approved by A.I.C.T.E., New Delhi & Permanently Affiliated to JNTU-GV, Vizianagaram)

NAAC “A+” Accredited Institute

Cherukupally (Village), Near Tagarapuvalasa Bridge, Vizianagaram (Dist), AP, Pin-531162.

[www.aietta.ac.in](http://www.aietta.ac.in), [principal@aietta.ac.in](mailto:principal@aietta.ac.in)

---

## DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

### **Institute Vision:**

To develop highly skilled professionals with strong ethical and human values.

### **Institute Mission:**

**M1:** To impart quality education through effective teaching–learning practices, industrial exposure, and professional training.

**M2:** To produce competent and highly knowledgeable graduates with a positive attitude and problem-solving ability.

**M3:** To instill self-confidence, leadership qualities, and social responsibility to face professional and societal challenges.

### **Department Vision:**

To emerge as a center of excellence in Computer Science and Engineering education and research, producing competent, industry-ready professionals with innovation, ethical values, and social responsibility.

### **Department Mission:**

**M1:** To impart strong foundational knowledge in computer science, programming, and emerging technologies through effective teaching–learning practices.

**M2:** To develop problem-solving, analytical, and research skills by encouraging innovation, projects, and industry interaction.

**M3:** To produce industry-ready professionals with proficiency in modern tools, teamwork, and professional communication.

**M4:** To inculcate ethical values, social responsibility, and lifelong learning skills to meet professional and societal challenges.

### **Program Educational Objectives (PEOs):**

**PEO1:** Graduates will apply strong foundations in computer science, mathematics, and engineering principles to analyze and solve real-world problems.

**PEO2:** Graduates will design, develop, and deploy software systems and applications using modern Engineering tools, technologies, and industry-standard practices.



# AVANTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

**(Autonomous)**

(Approved by A.I.C.T.E., New Delhi & Permanently Affiliated to J.N.T.U-GV, Vizianagaram)

NAAC "A+" Accredited Institute

Cherukupally (Village), Near Tagarapuvalasa Bridge, Vizianagaram (Dist) -531162.

[www.aietta.ac.in](http://www.aietta.ac.in), [principal@aietta.ac.in](mailto:principal@aietta.ac.in)

---

**PEO3:** Graduates will pursue successful careers in industry, entrepreneurship, higher education, or research through continuous learning and skill enhancement.

**PEO4:** Graduates will demonstrate professional ethics, effective communication, teamwork, and leadership while contributing responsibly to society.

## **Program Specific Outcomes (PSOs):**

**PSO1:** Apply core computing concepts to design efficient software and systems.

**PSO2:** Use modern tools and emerging technologies to solve real-world problems ethically.

## **Program Outcomes (POs):**

**PO1: Engineering Knowledge:** Apply knowledge of mathematics, natural science, computing, engineering fundamentals and an engineering specialization as specified in WK1 to WK4 respectively to develop to the solution of complex engineering problems.

**PO2: Problem Analysis:** Identify, formulate, review research literature and analyze complex engineering problems reaching substantiated conclusions with consideration for sustainable development. (WK1 to WK4)

**PO3: Design/Development of Solutions:** Design creative solutions for complex engineering problems and design/develop systems/components/processes to meet identified needs with consideration for the public health and safety, whole-life cost, net zero carbon, culture, society and environment as required. (WK5)

**PO4: Conduct Investigations of Complex Problems:** Conduct investigations of complex engineering problems using research-based knowledge including design of experiments, modelling, analysis & interpretation of data to provide valid conclusions. (WK8).

**PO5: Engineering Tool Usage:** Create, select and apply appropriate techniques, resources and modern engineering & IT tools, including prediction and modelling recognizing their limitations to solve complex engineering problems. (WK2 and WK6)

**PO6: The Engineer and The World:** Analyze and evaluate societal and environmental aspects while solving complex engineering problems for its impact on sustainability with reference to economy, health, safety, legal framework, culture and environment. (WK1, WK5, and WK7).

**PO7: Ethics:** Apply ethical principles and commit to professional ethics, human values, diversity and inclusion; adhere to national & international laws. (WK9)



# AVANTHI INSTITUTE OF ENGINEERING AND TECHNOLOGY

**(Autonomous)**

(Approved by A.I.C.T.E., New Delhi & Permanently Affiliated to J.N.T.U-GV, Vizianagaram)

NAAC “A+” Accredited Institute

Cherukupally (Village), Near Tagarapuvalasa Bridge, Vizianagaram (Dist) -531162.

[www.aietta.ac.in](http://www.aietta.ac.in), [principal@aietta.ac.in](mailto:principal@aietta.ac.in)

---

**PO8: Individual and Collaborative Team work:** Function effectively as an individual, and as a member or leader in diverse/multi-disciplinary teams.

**PO9: Communication:** Communicate effectively and inclusively within the engineering community and society at large, such as being able to comprehend and write effective reports and design documentation, make effective presentations considering cultural, language, and learning differences

**PO10: Project Management and Finance:** Apply knowledge and understanding of engineering management principles and economic decision-making and apply these to one's own work, as a member and leader in a team, and to manage projects and in multidisciplinary environments.

**PO11: Life-Long Learning:** Recognize the need for, and have the preparation and ability for i) independent and life-long learning ii) adaptability to new and emerging technologies and iii) critical thinking in the broadest context of technological change. (WK8).

**JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY GURJADA VIZIANAGARAM****VIZIANAGARAM – 535 003, Andhra Pradesh, India****B.Tech CSE (R23-COURSE STRUCTURE & SYLLABUS)****B.Tech. – III Year I Semester**

| S.No  | Category                                | Title                                                                                                                                                                                                   | L  | T | P  | Credits |
|-------|-----------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----|---|----|---------|
| 1     | Professional Core                       | Data Warehousing and Data Mining                                                                                                                                                                        | 3  | 0 | 0  | 3       |
| 2     | Professional Core                       | Compiler Design                                                                                                                                                                                         | 3  | 0 | 0  | 3       |
| 3     | Professional Core                       | Design and Analysis of Algorithms                                                                                                                                                                       | 3  | 0 | 0  | 3       |
| 4     | Professional Elective-I                 | 1. Object Oriented Analysis and Design<br>2. Software Testing Methodologies<br>3. Advanced Computer Architecture<br>4. Computer Graphics<br>5. 12 week MOOC Swayam/ NPTEL course recommended by the BoS | 3  | 0 | 0  | 3       |
| 5     | Open Elective-I                         | 1. Data Structures<br>2.Database Management Systems<br>3. Computer Organization<br>4.Scripting Languages<br>5. Operating Systems                                                                        | 3  | 0 | 0  | 3       |
| 6     | Professional Core                       | Data Mining Lab                                                                                                                                                                                         | 0  | 0 | 3  | 1.5     |
| 7     | Professional Core                       | Compiler Design Lab                                                                                                                                                                                     | 0  | 0 | 3  | 1.5     |
| 8     | Skill Enhancement course                | Full Stack development-2                                                                                                                                                                                | 0  | 1 | 2  | 2       |
| 9     | Engineering Science                     | Tinkering Lab/ SWAYAM Plus - Android Application Development (with Flutter)                                                                                                                             | 0  | 0 | 2  | 1       |
| 10    | Evaluation of Community Service Project |                                                                                                                                                                                                         | -  | - | -  | 2       |
| Total |                                         |                                                                                                                                                                                                         | 15 | 1 | 10 | 23      |



**B.Tech. III Year II Semester**

[illegible]



| III Year I Semester | DATA WAREHOUSING & DATA MINING | L | T | P | C |
|---------------------|--------------------------------|---|---|---|---|
|                     |                                | 3 | 0 | 0 | 3 |

### Course Objectives:

Students undergoing this course are expected to:

1. Understand the concepts of Data Ware housing and Data Mining
2. Master data mining techniques in various applications like social, scientific and environmental context.
3. Develop skill in selecting the appropriate data mining algorithm for solving practical problems.

### Course Outcomes:

After completion of the course, students should be able to:

1. Understand Data Warehouse fundamentals, Data Mining concepts, principles and its functionalities
2. Preprocess the data using various Data Preprocessing Techniques for mining applications
3. Design and deploy appropriate classification techniques to solve real world problems and further be able to assess the strengths and weaknesses of various methods and algorithms to analyze their behavior.
4. Demonstrate Association analysis techniques for generating association rules from data.
5. Use different Clustering techniques to cluster data and Cluster the high dimensional data for better organization of the data

### UNIT –I: Data Mining Systems and Knowledge Discovery Process:

**Data Warehouse and OLAP Technology:** An Overview- What Is a Data Warehouse. A Multidimensional Data Model - Need for Online Analytical Processing - OLTP V/s OLAP -OLAP Operations in Multidimensional Data Model. Data Warehouse Architecture, Data Warehouse Implementation, From Data Warehousing to Data Mining.

**Need and Usage of Data Mining Technologies** - Overview of Knowledge Discovery Process from Databases—What Motivated Data Mining - Why Is It Important - Data Mining Functionalities—What Kinds of Patterns Can Be Mined? Are All of the Patterns Interesting Classification of Data Mining Systems, Data Mining Task Primitives, Integration of a Data Mining System with a Database or Data Warehouse System, Major Issues in Data Mining.

### UNIT–II: Data Pre-processing:

**Data Exploration:** Data Objects and attribute types -Statistical description of data- Descriptive Data Summarization-Data Visualization - Data similarity and dissimilarity measures.

**Data Pre-processing:** Why Pre-process the Data -Data Cleaning-Data Integration-Data Reduction-Data Transformation and Data Discretization.

### UNIT–III: Classification:

Basic issues regarding classification and predication - General Approach to solving a classification problem- Decision Tree Classification, Attribute Selection Measures, Tree Pruning- Bayesian



Classification – Rule Based Classification – Support Vector Machines.

**Classification Model Evaluation and Selection** - Accuracy and Error measures, Holdout, Random Sampling, Cross Validation, Bootstrap, Comparing Classifier performance using ROC Curves.

**UNIT–IV: Mining Frequent Patterns and Association Rules:**

Basic Concepts-Problem Definition- Market Basket Analysis- Frequent Itemsets- Closed Itemsets and Association Rules - Frequent Pattern Mining - Efficient and Scalable Frequent Itemset Mining Methods- the Apriori Algorithm for finding Frequent Itemsets Using Candidate Generation - Generating Association Rules from Frequent Itemsets - A pattern growth approach for mining Frequent Itemsets- FP-Growth Algorithm

**UNIT V: Cluster Analysis:**

Basics and Importance of Cluster Analysis- Clustering techniques- Different Types of Clusters- Partitioning Methods (K-Means, K Medoids) -Strengths and Weaknesses. Hierarchical Methods (Agglomerative, Divisive) - Density-Based Methods (DBSCAN, OPTICS)-

**Text Books:**

- i. Introduction to Data Mining: Pang-Ning Tan & Michael Steinbach, Vipin Kumar, Pearson.
- ii. Data Mining concepts and Techniques, 3/e, Jiawei Han, Michel Kamber, Elsevier.

**References:**

- i. Data Mining Techniques and Applications: An Introduction, Hongbo Du, Cengage Learning.
- ii. Data Mining :VikramPudi and P. Radha Krishna,Oxford.
- iii. Data Mining and Analysis - Fundamental Concepts and Algorithms; Mohammed J.Zaki, Wagner Meira, Jr, Oxford
- iv. Data Warehousing Data Mining & OLAP, Alex Berson, Stephen Smith,TMH.

**E-resources:**

- i. [http://onlinecourses.nptel.ac.in/noc18\\_cs14/preview](http://onlinecourses.nptel.ac.in/noc18_cs14/preview) (NPTEL course by Prof.Pabitra Mitra)
- ii. [http://onlinecourses.nptel.ac.in/noc17\\_mg24/preview](http://onlinecourses.nptel.ac.in/noc17_mg24/preview)  
(NPTEL course by Dr. Nandan Sudarshanam & Dr. Balaraman Ravindran)
- iii. [http://www.saedsayad.com/data\\_mining\\_map.htm](http://www.saedsayad.com/data_mining_map.htm)



| III Year I Semester | COMPILER DESIGN | L | T | P | C |
|---------------------|-----------------|---|---|---|---|
|                     |                 | 3 | 0 | 0 | 3 |

### Course Objectives:

The objectives of this course is to acquire knowledge on the

1. Design of lexical analyzers, Syntax analyzers, Intermediate code generators
2. Usage of Lex/Bison tools in writing compilers
3. Issues in the code generation, code generation generation algorithms.

### Course Outcomes:

The students should be able to:

1. Understand the basics of language processing and implement lexical analyzer for any language.
2. Describe the different types of parsing and implement parser for any language.
3. Explain the different intermediate code representations and use Syntax directed
4. Apply the basics of data flow analysis, optimizations, and run time environment required for handling recursive procedures.
5. Discuss the issues in the code generation and code generation algorithms.

### UNIT - I:

**Introduction and Lexical Analysis:** Language Processors, the structure of a compiler, the science of building a compiler, phases of a compiler. Lexical Analysis: The role of the lexical analyzer, Identifying tokens, Transition diagrams for recognizing tokens, Input buffering, The lexical analyzer generator Lex, Finite automata, Conversion from regular expressions to automata, design of a lexical analyzer generator, Optimization of DFA-based pattern matchers.

### UNIT - II:

**Syntax Analysis:** Introduction, Context-Free Grammars, BNF(Backus-Naur Form), EBNF(Extended Backus-Naur Form). Preprocessing of grammars: left recursion elimination, left factoring. Top-Down Parsing: Recursive-descent parsers, LL(1) parsers. Bottom-Up parsing: Introduction to LR parsers, Simple LR, Canonical LR, and Look ahead LR. Extending parsers to handle ambiguous grammars, Parser generators Yacc/Bison.

### UNIT – III:

#### **Syntax-Directed Translation, Semantic Analysis, Intermediate Code Generation**

Syntax-Directed Definitions, Evaluation orders for SDD's, Applications of Syntax-Directed Translation, Syntax-Directed Translation Schemes, and Implementing L-Attributed SDD's.

Intermediate code generation: Variants of Syntax trees, Three-address code, Types and declarations, Type checking, Control flow, Back patching, Switch-Statements, Intermediate Code for Procedures.

### UNIT - IV:

#### **Code Optimization, Run-time Environment**

Run-Time Environments: Storage organization, Activation record, Stack allocation, Access to nonlocal data on the stack, Heap management, Introduction to garbage collection, Introduction to



trace-based collection. Machine-Independent optimizations: The principal sources of optimization, Basic blocks and flow graphs, Introduction to data-flow analysis, Foundations of data-flow analysis, Constant propagation.

**UNIT - V: Target Code Generation**

Code Generation: Issues in the design of a Code Generator, The target language, Addresses in the target code, A simple code generator.

Machine-dependent Optimizations: Peephole optimization, Register allocation and assignment, Dynamic Programming code generation.

**Text Books:**

- i. Compilers: Principles, Techniques and Tools, Second Edition, Alfred V. Aho, MonicaS. Lam, Ravi Sethi, Jeffrey D. Ullman, Pearson.
- ii. Compiler Construction-Principles and Practice, Kenneth C Loudon, Cengage Learning.

**Reference Books:**

- i. Modern Compiler Implementation in C, Andrew W Appel, Revised edition, Cambridge University Press.
- ii. The Theory and Practice of Compiler writing, J. P. Tremblay and P. G. Sorenson, TMH
- iii. lex & yacc, 2nd Edition by John Levine, Doug Brown, Tony Mason

**E-resources:**

- i. <https://www.edx.org/course/compilers>
- ii. <https://nptel.ac.in/courses/106/108/106108113/>



| III Year I Semester | DESIGN AND ANALYSIS OF<br>ALGORITHMS | L | T | P | C |
|---------------------|--------------------------------------|---|---|---|---|
|                     |                                      | 3 | 0 | 0 | 3 |

**Course Objectives:**

The objectives of this course is to acquire knowledge on the

1. Analyze the asymptotic performance of algorithms
2. Write rigorous correctness proofs for algorithms
3. Demonstrate a familiarity with major algorithms and data structures

**Course Outcomes: The students should be able to:**

1. Measure the performance and calculate the Time & Space complexities of algorithms.
2. Design effective algorithms based on strategies like Decrease and Conquer, Divide and Conquer Transform and Conquer,
3. Develop efficient algorithms using Greedy method.
4. Discuss various problems suitable to Dynamic programming.
5. Find an optimal solution by applying different Backtracking and Branch & Bound technique.

**UNIT - I:**

**Introduction:** What is an Algorithm, Algorithm Specification, Pseudo code Conventions Recursive Algorithm, Performance Analysis, Space Complexity, Time Complexity, Amortized Complexity, Asymptotic Notation, Practical Complexities' Performance Measurement.

**UNIT - II:**

**Decrease-and-Conquer:** Insertion Sort Algorithms for Generating Combinatorial Objects Decrease-by-a-Constant-Factor Algorithms Variable-Size-Decrease Algorithms

**Divide and Conquer:** Merge Sort, Quick Sort, Multiplication of Large Integers and Strassen's Matrix Multiplication

**Transform and conquer:** Pre-sorting Balanced Search Trees, Heaps and Heap sort

**UNIT - III:**

**The Greedy Method:** The General Method, Knapsack Problem, Job Sequencing with Deadlines Minimum-cost Spanning Trees, Prim's Algorithm, Kruskal's Algorithms, An Optimal Randomized Algorithm, Optimal Merge Patterns, Single Source Shortest Paths.

**UNIT - IV:**

**Dynamic Programming:** The General Method Multistage graph ,All - Pairs Shortest Paths, , String Edition, 0/1 Knapsack, Reliability Design, optimal binary search trees.

**UNIT - V:**

**Backtracking:** The General Method, The S-Queens Problem, Sum of Subsets, Graph Coloring Hamiltonian Cycles

**Branch and Bound:** The Method, Least cost (LC) Search, The 15-Puzzle: an Example, Control Abstraction for LC-Search, Bounding, FIFO Branch-and-Bound, LC Branch and Bound, 0/1 Knapsack problem, LC Branch-and Bound Solution, FIFO Branch-and-Bound Solution, Traveling Salesperson problem.

**Text Books:**

- i. Fundamentals of computer algorithms E. Horowitz S. Sahni, University Press

**JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY GURJADA  
VIZIANAGARAM**



**VIZIANAGARAM – 535 003, Andhra Pradesh, India  
B.Tech CSE (R23-COURSE STRUCTURE & SYLLABUS)**

- ii. Introduction to the design and analysis of Algorithms Anany Levitin pearson ,3<sup>rd</sup> edition
- iii. Introduction to Algorithms Thomas H Cormen PHILearning

**Reference Books:**

- i. The Design and Analysis of Computer Algorithms, Alfred V Aho John E Hopcroft Jeffrey D Ullman
- ii. Algorithm Design, Jon Kleinberg, Pearson
- iii. Algorithms, by Dasgupta, Papadimitrou and Vazirani, McGraw-Hill Education, 2006.



| III Year I Semester | OBJECT ORIENTED ANALYSIS AND<br>DESIGN<br>(PROFESSIONAL ELECTIVE-I) | L | T | P | C |
|---------------------|---------------------------------------------------------------------|---|---|---|---|
|                     |                                                                     | 3 | 0 | 0 | 3 |

**Course Objectives:** The main objective is the students to

1. Become familiar with all phases of OOAD.
2. Master the main features of the UML.
3. Master the main concepts of Object Technologies and how to apply them at work and develop the ability to analyze and solve challenging problem in various domains.

**Course Outcomes: The students should be able to:**

1. Select the basic elements of modeling such as Things, Relationships and Diagrams depending on the views of UML Architecture and SDLC.
2. Apply basic and Advanced Structural Modeling Concepts for designing real time applications.
3. Design Class and Object Diagrams that represent Static Aspects of a Software System.
4. Analyze Dynamic Aspects of a Software System using Use Case, Interaction and Activity Diagrams.
5. Apply techniques of State Chart Diagrams and Implementation Diagrams to model behavioral aspects and Runtime environment of Software Systems.

#### **UNIT I:**

**Introduction:** The Structure of Complex systems, The Inherent Complexity of Software, Attributes of Complex System, Organized and Disorganized Complexity, Bringing Order to Chaos, Designing Complex Systems. **Case Study:** System Architecture: Satellite-Based Navigation

#### **UNIT II:**

**Introduction to UML:** Importance of modeling, principles of modeling, object oriented modeling, conceptual model of the UML, Architecture, and Software Development Life Cycle. **Basic Structural Modeling:** Classes, Relationships, common Mechanisms, and diagrams. **Case Study:** Control System: Traffic Management.

#### **UNIT III:**

**Class & Object Diagrams:** Terms, concepts, modeling techniques for Class & Object Diagrams. **Advanced Structural Modeling:** Advanced classes, advanced relationships, Interfaces, Types and Roles, Packages. **Case Study:** AI: Cryptanalysis.

#### **UNIT IV:**

**Basic Behavioral Modeling-I:** Interactions, Interaction diagrams Use cases, Use case Diagrams, Activity Diagrams. **Case Study:** Web Application: Vacation Tracking System



#### UNIT V:

**Advanced Behavioral Modeling:** Events and signals, state machines, processes and Threads, time and space, state chart diagrams. **Architectural Modeling:** Component, Deployment, Component diagrams and Deployment diagrams. **Case Study:** Weather Forecasting

#### Text Books:

- i. Grady BOOCH, Robert A. Maksimchuk, Michael W. ENGLE, Bobbi J. Young, Jim Conallen, Kellia Houston , “Object- Oriented Analysis and Design with Applications”, 3rd edition, 2013, PEARSON.
- ii. Grady Booch, James Rumbaugh, Ivar Jacobson: The Unified Modeling Language User Guide, Pearson Education.

#### Reference Books:

- i. Meilir Page-Jones: Fundamentals of Object Oriented Design in UML, Pearson Education.
- ii. Pascal Roques: Modeling Software Systems Using UML2, WILEY- Dreamtech India Pvt. Ltd.
- iii. AtulKahate: Object Oriented Analysis & Design, The McGraw-Hill Companies.
- iv. Applying UML and Patterns: An introduction to Object – Oriented Analysis and Design and Unified Process, Craig Larman, Pearson Education.



| III Year I Semester | SOFTWARE TESTING<br>METHODOLOGIES<br>(PROFESSIONAL ELECTIVE-I) | L | T | P | C |
|---------------------|----------------------------------------------------------------|---|---|---|---|
|                     |                                                                | 3 | 0 | 0 | 3 |

### Course Objectives:

The objectives of this course is to acquire knowledge on the

1. Describe the principles and procedures for designing test cases.
2. Provide supports to debugging methods.
3. Acts as the reference for software testing techniques and strategies.

### Course Outcomes:

The students should be able to:

1. Define Software testing terminology and methodology
2. Discuss and Classify various testing techniques for conducting different types of software testing
3. Apply different software testing techniques.
4. Construct test cases by understanding test suite management and software quality management.
5. Demonstrate modern software testing tools and testing of Object Oriented Software and Web based software.

### UNIT-I:

**Introduction:** Purpose of Testing, Dichotomies, Model for Testing, Consequences of Bugs, Taxonomy of Bugs. **FLOW GRAPHS AND PATH TESTING:** Basics Concepts of Path Testing, Predicates, Path Predicates and Achievable Paths, Path Sensitizing, Path Instrumentation, Application of Path Testing.

### UNIT-II:

**Transaction Flows Testing:** Transaction Flows, Transaction Flow Testing Techniques.

**Dataflow Testing:** Basics of Dataflow Testing, Strategies in Dataflow Testing, Application of Dataflow Testing.

**Domain Testing:** Domains and Paths, Nice & Ugly Domains, Domain testing, Domains and Interfaces Testing, Domain and Interface Testing, Domains and Testability.

### UNIT-III:

**Paths, Path Products and Regular Expressions:**

Path Products & Path Expression, Reduction Procedure, Applications, Regular Expressions & Flow Anomaly Detection.

**Syntax Testing:** Why, What and How, A Grammar for formats, Test Case Generation, Implementation and Application and Testability Tips.

**Based Testing:** Overview, Decision Tables, Path Expressions, KV Charts, and Specifications.



**UNIT-IV:**

**State, State Graphs And Transition Testing:** State Graphs, Good & Bad State Graphs, State Testing, and Testability Tips.

**Graph Matrices and Application:**

Motivational overview, matrix of graph, relations, power of a matrix, node reduction algorithm.

**UNIT-V:**

**Software Testing Tools:** Introduction to Testing, Automated Testing, Concepts of Test Automation, Introduction to list of tools like Winrunner, LoadRunner, Jmeter, Selenium About Win Runner Using Win runner, Mapping the GUI, Recording Test, Working with Test, Enhancing Test, Checkpoints, Test Script Language, Putting it all together, Running and Debugging Tests, Analyzing Results, Batch Tests, Rapid Test Script Wizard.

**Text Books:**

- i. Software testing techniques – Boris Beizer, Dreamtech, second edition.
- ii. Software Testing- Yogesh Singh, Cambridge

**Reference Books:**

- i. The Craft of software testing - Brian Marick, Pearson Education.
- ii. Software Testing, 3rd edition, P.C. Jorgensen, Aurbach Publications (Dist.by SPD).
- iii. Software Testing, N. Chauhan, Oxford University Press.
- iv. Introduction to Software Testing, P. Ammann & J. Offutt, Cambridge Univ. Press.
- v. Effective methods of Software Testing, Perry, John Wiley, 2nd Edition, 1999.
- vi. Software Testing Concepts and Tools, P. Nageswara Rao, dreamtech Press
- vii. Win Runner in simple steps by Hakeem Shittu, Genixpress, 2007.
- viii. Foundations of Software Testing, D. Graham & Others, Cengage Learning.



| III Year I Semester | ADVANCED COMPUTER<br>ARCHITECTURE<br>(PROFESSIONAL ELECTIVE-I) | L | T | P | C |
|---------------------|----------------------------------------------------------------|---|---|---|---|
|                     |                                                                | 3 | 0 | 0 | 3 |

### Course Objectives:

1. Understand fundamentals of computer design
2. Understand instruction level parallelism
3. Understand memory hierarchy in computer systems

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

- 1: Describe fundamentals concepts of computer design
- 2: Explain different types of Parallelisms in Computer Architectures.
3. Design memory hierarchy methods
- 4: Explain the Thread level parallelism
- 5: Explain the different types of Storage systems and bench marking the storage device.

### UNIT I

**Introduction:** Fundamentals of Computer design- Technology trends- cost- measuring and reporting performance quantitative principles of computer design. Instruction level parallelism (ILP) - over coming data hazards- reducing branch costs –high performance instruction delivery- hardware based speculation- limitation of ILP

### UNIT II

**ILP software approach** - compiler techniques- static branch protection - VLIW approach - H.W support for more ILP at compile time- H.W verses S.W Solutions

### UNIT III

**Memory hierarchy design-** cache performance- reducing cache misses penalty and miss rate – virtual memory- protection and examples of VM.

### UNIT IV

**Multiprocessors and thread level parallelism-** symmetric shared memory architecturesdistributed shared memory- Synchronization- multi threading.



## **UNIT V**

**Storage systems-** Types – Buses - RAID- errors and failures- bench marking a storage device- designing a I/O system.

## **TEXT BOOK(S)**

i. Computer Architecture and parallel Processing, Kai Hwang and A.Briggs International Edition McGraw-Hill.

## **REFERENCES BOOK(S)**

i. Computer Architecture A quantitative approach 3rd edition John L. Hennessy & David A. Patterson Morgan Kufmann (An Imprint of Elsevier)

ii. Advanced Computer Architectures, Dezso Sima, Terence Fountain, Peter Kacsuk, Pearson.

iii. Parallel Computer Architecture, A Hardware / Software Approach, David E. Culler, Jaswinder Pal singh with Anoop Gupta, Elsevier

## **E-Resources:**

- i. <https://nptel.ac.in/courses/106/105/106105163/>



| III Year I Semester | COMPUTER GRAPHICS | L | T | P | C   |
|---------------------|-------------------|---|---|---|-----|
|                     |                   | 0 | 0 | 3 | 1.5 |

**Course Objectives:**

1. To introduce fundamental concepts and theory of computer graphics.
2. To develop, design and implement two and three dimensional graphical structures
3. To learn Creation, Management and Transmission of Multimedia objects.

**Course Outcomes:**

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

1. Develop various 2D output primitives like lines, circles, polygons.
2. Generate 3D representations by using various projection transformations.
3. Illustrate various illumination and shading models.
4. Apply OpenGL graphic software and rendering techniques for 3D object representations.
5. Create images using fractals and iterated functions

**UNIT-I:**

**Introduction to Graphics:** Application areas of Computer Graphics, overview of graphics systems, video-display devices, graphics monitors and work stations and input devices.

**2D Primitives:** Output primitives-Line, Circle and Ellipse drawing algorithms, Attributes of output primitives, Two dimensional Geometric transformations, Two dimensional viewing Line, Polygon, Curve and Text clipping algorithms.

**UNIT-II:**

**3D Concepts:** Parallel and Perspective projections - Three dimensional object representations – Polygons, Curved lines, Splines, Quadric Surfaces, - Visualization of data sets - 3D transformations – Viewing -Visible surface identification.

**UNIT-III:**

**Illumination and Shading:** Background, simple lighting model, shading models, intensity representation, color models, texture synthesis

**UNIT-IV:**

**Graphics Hardware and Software:** Graphics programming using OpenGL-Basic graphics primitives, Drawing three dimensional objects, Drawing three dimensional scenes.

**Rendering:** Introduction to shading models, Flat and Smooth shading, Adding texture to faces, Adding shadows of objects, Building a camera in a program, Creating shaded objects.

**UNIT V:**

**Fractals:** Fractals and Self similarity, Peano curves, Creating image by iterated functions, Mandelbrot sets, Julia Sets, Random Fractals

**Overview of Ray Tracing:** Intersecting rays with other primitives, Adding Surface texture, Reflections and Transparency, Boolean operations on Objects.



**Text Books:**

- i. Donald Hearn, Pauline Baker, Computer Graphics – C Version, Pearson Education.
- ii. F.S. Hill, Computer Graphics using OPENGL, Pearson Education.

**Reference Books:**

- i. James D. Foley, Andries Van Dam, Steven K. Feiner, John F. Hughes, Computer Graphics Principles and practice in C, Pearson Education.



| III Year I Semester | DATA STRUCTURES<br>(OPEN ELECTIVE-I) | L | T | P | C |
|---------------------|--------------------------------------|---|---|---|---|
|                     |                                      | 3 | 0 | 0 | 3 |

**Course Objectives:**

1. To teach efficient storage mechanisms of data for an easy access.
2. To develop application using data structures.
3. To improve the logical ability

**Course Outcomes:** On completion of this course, the student will be able to:

1. Compare the performances of various Searching and Sorting techniques in terms of time and space complexities.
2. Illustrate the applications of Stacks.
3. Implement various types of Queues and their efficient operations.
4. Demonstrate the advantages of dynamic memory allocation via linked lists.
5. Implement the basic operations, search and traversals on Trees.

**UNIT-I**

Time and space complexity, Data Structures – Introduction to Data Structures, abstract data types, Searching and Sorting – Sorting- selection sort, bubble sort, insertion sort, quick sort, merge sort, shell sort, radix sort, Searching-linear and binary search methods, comparison of sorting and searching methods.

**UNIT –II**

Stacks-Operations, array and linked representations of stacks, stack applications -infix to postfix conversion, postfix expression evaluation, recursion implementation.

**UNIT-III**

Queues-operations, array, and linked representations. Circular Queue operations, Dequeues, applications of queues.

**UNIT-IV**

Linear list – singly linked list implementation, insertion, deletion and searching operations on linear list, circular linked list implementation, Double linked list implementation, insertion, deletion and searching operations. Applications of linked lists.

**UNIT-V**

Trees – Definitions, tree representation, properties of trees, Binary tree, Binary tree representation, binary tree properties, binary tree traversals, binary tree implementation, applications of trees.

**Text Books:**

1. Fundamentals of Data structures in C, 2nd Edition, E.Horowitz, S.Sahni and Susan Anderson-Freed,



Universities Press.

2 Data structures A Programming Approach with C, D.S.Kushwaha and A.K.Misra, PHI.

**References:**

1. Data structures: A Pseudocode Approach with C, 2nd edition,
2. R.F.Gilberg And B.A.Forouzan, Cengage Learning.
3. Data structures and Algorithm Analysis in C, 2nd edition, M.A.Weiss, Pearson.
4. Data Structures using C, A.M.Tanenbaum, Y. Langsam, M.J.Augenstein, Pearson.
5. Data structures and Program Design in C, 2nd edition, R.Kruse, C.L.Tondo and B.Leung, Pearson



|                     |                                                   |   |   |   |   |
|---------------------|---------------------------------------------------|---|---|---|---|
| III Year I Semester | DATA BASE MANAGEMENT SYSTEMS<br>(OPEN ELECTIVE-I) | L | T | P | C |
|                     |                                                   | 3 | 0 | 0 | 3 |

### Course Objectives:

1. Train in the fundamental concepts of database management systems, database modeling and design, SQL, PL/SQL, and System implementation techniques.
2. Enable students to model ER diagram for any customized applications.
3. To learn the principles of systematically designing and using large scale Database Management Systems for various applications.

### Course Outcomes:

1. Understand the usage of Key Constraints on Database.
2. Describe ER model and normalization for database design.
3. Create, maintain, and manipulate a relational database using SQL.
4. Understand efficient data storage and retrieval mechanism, recovery techniques.
5. Design and build database system for a given real world problem.

### UNIT-I:

**An Overview of Database Management:** Introduction- Importance of Database System, Data Independence- Relation Systems and Others- Summary, Database system architecture, Introduction- The Three Levels of Architecture-The External Level- the Conceptual Level- the Internal Level- Mapping- the Database Administrator-The Database Management Systems- Client/Server Architecture.

### UNIT-II:

**The E/R Models:** The Relational Model, Relational Calculus, Introduction to Database Design, Database Design and ER Diagrams-Entities Attributes, Entity Sets-Relationship and Relationship Sets-Conceptual Design with the ER Models.

**The Relational Model:** Integrity Constraints Over Relations- Key Constraints –Foreign Key Constraints-General Constraints, Relational Algebra and Calculus, Relational Algebra- Selection and Projection- Set Operation, Renaming – Joins- Division- More Examples of Queries, Relational Calculus - Tuple Relational Calculus, Domain Relational Calculus.

### UNIT-III:

**Queries, Constraints, Triggers:** The Form of Basic SQL Query, Union, Intersect, and Except, Nested Queries, Aggregate Operators, Null Values, Complex Integrity Constraints in SQL, Triggers and Active Database.

**Schema Refinement (Normalization) :** Purpose of Normalization or Schema Refinement, Concept of



Functional Dependency, Normal Forms Based on Functional Dependency(1NF, 2NF and 3NF), Concept of Surrogate Key, Boyce-Codd Normal Form(BCNF), Lossless Join and Dependency Preserving Decomposition, Fourth Normal Form(4NF).

#### **UNIT-IV:**

##### **Transaction Management and Concurrency Control:**

Transaction, Properties of Transactions, Transaction Log, Transaction Management with SQL using Commit Rollback and Save Point, Concurrency Control for Lost Updates, Uncommitted Data, Inconsistent Retrievals, and the Scheduler.

**Concurrency Control with Locking Methods:** Lock Granularity, Lock Types, Two Phase Locking For Ensuring Serializability, Deadlocks, Concurrency Control with Time Stamp Ordering: Wait/Die and Wound/Wait Schemes, Database Recovery Management : Transaction Recovery.

#### **UNIT-V:**

**Overview of Storages and Indexing:** Data on External Storage- File Organization and Indexing – Clustered Indexing – Primary and Secondary Indexes, Index Data Structures, Hash-Based

**Indexing** – Tree- Based Indexing, Comparison of File Organization.

#### **Text Books:**

1. Introduction to Database Systems, CJ Date, Pearson.
2. Data base Management Systems, Raghurama Krishnan, Johannes Gehrke, TATAMcGraw Hill 3rd Edition.

#### **References Books:**

1. Data base Systems design, Implementation, and Management, Peter Rob & Carlos Coronel 7th Edition.
2. Fundamentals of Database Systems, Elmasri Navrate Pearson Education.
3. Database Systems - The Complete Book, H G Molina, J D Ullman, J Widom Pearson.
4. Data base System Concepts, 5/e, Silberschatz, Korth, TMH.



| III Year I Semester | COMPUTER ORGANIZATION<br>(OPEN ELECTIVE-I) | L | T | P | C |
|---------------------|--------------------------------------------|---|---|---|---|
|                     |                                            | 3 | 0 | 0 | 3 |

### Course Objectives:

1. Gives a view of computer system from user's perspective, representation of data.
2. Understand the architecture of a modern computer with its various processing units. Also the Performance measurement of the computer system.
3. Illustration of data paths and control flow for sequencing in CPUs, Microprogramming of control unit of CPU.

### Course Outcomes:

1. Understand the architecture of modern computer.
2. Able to calculate the effective address of an operand by addressing modes.
3. Apply different instruction types.
4. Determine the importance of memory management system of computer.
5. Design the roles and functions of processing unit and micro programmed control.

### UNIT -I:

**Basic Structure of Computers:** Functional unit, Basic Operational concepts, Bus structures, System Software, Performance, The history of computer development.

**Data Representation:** Data types, Complements, Fixed Point Representation, Floating- Point Representation, Other Binary Codes, Error Detection codes.

### UNIT -II:

#### Machine Instruction and Programs:

Instruction and Instruction Sequencing: Register Transfer Notation, Assembly Language Notation, Basic Instruction Types, Addressing Modes, Basic Input/output Operations, The role of Stacks and Queues in computer programming equation. Component of Instructions: Logic Instructions, shift and Rotate Instructions.

### UNIT -III:

**Type of Instructions:** Arithmetic and Logic Instructions, Branch Instructions, Addressing Modes, Input/output Operations.

**Input/output Organization:** Accessing I/O Devices, Interrupts: Interrupt Hardware, Enabling and Disabling Interrupts, Handling Multiple Devices, Direct Memory Access, Buses: Synchronous Bus, Asynchronous Bus, Interface Circuits, Standard I/O Interface: Peripheral Component Interconnect (PCI) Bus, Universal Serial Bus (USB).

### UNIT -IV:

**The Memory Systems:** Basic memory circuits, Memory System Consideration, Read-Only Memory: ROM, PROM, EPROM, EEPROM, Flash Memory, Cache Memories: Mapping Functions, INTERLEAVING



**Secondary Storage:** Magnetic Hard Disks, Optical Disks.

**UNIT -V:**

**Processing Unit:** Fundamental Concepts: Register Transfers, Performing an Arithmetic or Logic Operation, Fetching a Word from Memory, Execution of Complete Instruction, Hardwired Control,

**Micro programmed Control:** Microinstructions, Micro program Sequencing, Wide Branch Addressing Microinstructions with next – Address Field.

**Text Books:**

- i. Computer Organization, Carl Hamacher, Zvonks Vranesic, Safea Zaky, 5th Edition, McGraw Hill.
- ii. Computer Architecture and Organization, John P. Hayes, 4<sup>th</sup> Edition, McGrawHill.

**Reference Books:**

- i. Computer Organization and Architecture – William Stallings Sixth Edition, Pearson/PHI.
- ii. Structured Computer Organization–AndrewS.Tanenbaum,4th Edition PHI/Pearson.
- iii. Fundamentals or Computer Organization and Design, - Sivaraama Dandamudi Springer Int.Edition.
- iv. “Computer Organization and Design: The Hardware/Software Interface” by David A. Patterson and John L.Hennessy.
- vi. J.P. Hayes, "Computer Architecture and Organization", McGraw-Hill,1998.

**E-Resources:**

- i. <https://nptel.ac.in/courses/106/106/106106092/>
- ii. <https://nptel.ac.in/courses/106/105/106105163/>



| III Year I Semester | SCRIPTING LANGUAGES<br>(OPEN ELECTIVE-I) | L | T | P | C |
|---------------------|------------------------------------------|---|---|---|---|
|                     |                                          | 3 | 0 | 0 | 3 |

### Course Objectives:

1. Introduces scripting languages such as Perl, Ruby, PHP and TCL.
2. Design, code, and test applications using scripting languages.
3. An ability to create PHP scripts to store and manipulate user data.

### Course Outcomes:

1. Acquire programming skills in RUBY scripting language.
2. Ability to create and run scripts using PERL.
3. To gain some fluency programming in Perl and PHP and related languages.
4. To improve knowledge of advanced concepts in PHP.
5. Gain knowledge of the strengths and weakness of Perl, TCL and Ruby; and select an appropriate language for solving a given problem.

### UNIT-I:

**Introduction:** Ruby, Rails, the structure and Execution of Ruby Programs, Package Management with RUBYGEMS.

**Ruby and Web:** Writing CGI scripts, cookies, Choice of Web servers, SOAP and web services

**Ruby Tk:** Simple Tk Application, widgets, Binding events, Canvas, scrolling

### UNIT-II:

#### Introduction to PERL and Scripting:

Scripts and Programs, Origin of Scripting , Scripting Today, Characteristics of Scripting Languages, Uses for Scripting Languages, Web Scripting, and the universe of Scripting Languages. PERL- Names and Values, Variables, Scalar Expressions, Control Structures, arrays, list, hashes, strings, pattern and regular expressions, subroutines.

### UNIT-III:

#### Advanced PERL:

Finer points of looping, pack and unpack, file system, eval, data structures, packages, modules, objects, interfacing to the operating system, Creating Internet ware applications, Dirty Hands Internet Programming, security Issues.

**PHP Basics :** PHP Basics- Features, Embedding PHP Code in your Web pages, Outputting the data to the browser, Data types, Variables, Constants, expressions, string interpolation, control structures, Function, Creating a Function, Function Libraries, Arrays, strings and Regular Expressions.

### UNIT-IV:

**Advanced PHP Programming:** PHP and Web Forms, Files, PHP Authentication and Methodologies - Hard Coded, File Based, Database Based, IP Based, Login Administration, Uploading Files with PHP, Sending Email using PHP, PHP Encryption Functions, the Mcrypt package and Building Web sites for the



World.

**UNIT -V:**

TCL: Structure, syntax, Variables and Data in TCL, Control Flow, Data Structures, input/output, procedures , strings , patterns, files, Advance TCL, eval, source, exec and uplevel commands, Name spaces, trapping errors, event driven programs, making applications internet aware, Nuts and Bolts Internet Programming, Security Issues, CInterface.

**Tk:** Visual Tool Kits, Fundamental Concepts of Tk, Tk by example, Events and Binding, Perl-Tk.

**Text Books:**

- i. The World of Scripting Languages, David Barron, Wiley Publications.
- ii. Ruby Programming language by David Flanagan and Yukihiro Matsumoto O'Reilly
- iii. Beginning PHP and MySQL, 3rd Edition, Jason Gilmore, Apress Publications(Dream tech)

**Reference Books:**

- i. Open Source Web Development with LAMP using Linux, Apache, MySQL, Perl and PHP, J.Lee and B.Ware (Addison Wesley) Pearson Education.
- ii. Perl by Example, E.Quigley, Pearson Education.
- iii. PHP 6 Fast and Easy Web Development, Julie Meloni and Matt Telles, Cengage Learning Publications.



| III Year I Semester | OPERATING SYSTEMS<br>(OPEN ELECTIVE-I) | L | T | P | C |
|---------------------|----------------------------------------|---|---|---|---|
|                     |                                        | 3 | 0 | 0 | 3 |

### Course Objectives:

1. Provide knowledge about the services rendered by operating systems.
2. Present detail discussion on processes, threads and scheduling algorithms.
3. Discuss various file-system implementation issues and memory management techniques.

### Course Outcomes:

1. Understand the importance of operating systems and different types of system calls.
2. Analyze the communication between processes and various process scheduling algorithms.
3. Understand the process synchronization, different ways for deadlocks handling.
4. Analyze various memory mapping techniques and different page replacement methods.
5. Evaluate various file allocation and disk scheduling algorithms.

### UNIT-I: Operating Systems Overview:

**Introduction:** what is an operating system, Types of operating systems, operating systems concepts, operating systems services, Introduction to System call, System call types, Operating System Generation.

### UNIT-II: Process Management:

**Process concept:** Process Concept, Process Scheduling, Operations on Processes, Inter process Communication.

**Multithreaded Programming:** Overview, Multithreading models, Threading Issues.

**Process scheduling:** Basic Concepts, Scheduling Criteria, Scheduling Algorithms.

### UNIT-III: Synchronization:

**Process Synchronization:** The Critical-Section Problem, Synchronization Hardware, Semaphores, Classic Problems of Synchronization, Monitors, Synchronization examples.

**Principles of deadlock** – System Model, Deadlock Characterization, Deadlock Prevention, Detection and Avoidance, Recovery from Deadlock.

### UNIT-IV: Memory Management:

**Memory Management strategies:** Swapping, Contiguous Memory Allocation, Segmentation, Paging, Structure of the Page Table.

**Virtual Memory Management:** Virtual Memory, Demand Paging, Page-Replacement Algorithms, Thrashing.

**UNIT-V: File system Interface-** The concept of a file, Access Methods, Directory and Disk structure, File system mounting.

**File System implementation:** File system structure, allocation methods, free-space management.

**Mass-storage structure:** Overview of Mass-storage structure, Disk scheduling, Device drivers.



**Text Books:**

1. Silberschatz A, Galvin P B, and Gagne G, Operating System Concepts, 9th edition, Wiley, 2013.
2. Tanenbaum A S, Modern Operating Systems, 3rd edition, Pearson Education, 2008. (for Interprocess Communication and File systems).

**References:**

- i. Tanenbaum A S, Woodhull A S, Operating Systems Design and Implementation, 3rd edition, PHI, 2006.
- ii. Dhamdhare D M, Operating Systems A Concept Based Approach, 3rd edition, Tata McGraw-Hill, 2012.
- iii. Stallings W, Operating Systems -Internals and Design Principles, 6th edition, Pearson Education, 2009.
- iv. Nutt G, Operating Systems, 3rd edition, Pearson Education, 2004.



| III Year I Semester | DATA MINING LAB | L | T | P | C   |
|---------------------|-----------------|---|---|---|-----|
|                     |                 | 0 | 0 | 3 | 1.5 |

**Course Objectives:**

1. Practical exposure on implementation of well known data mining tasks and their effective use in discovering interesting hidden patterns from large datasets.
2. Exposure to real time data sets for analysis and prediction.
3. Focus is on the main process of data mining such as data preparation, classification, clustering, association analysis, and pattern evaluation

**Software Requirements:** WEKA Tool and R Programming/Python Programming

**Course Outcomes:** After undergoing the course students will be able to:

1. Create summary statistics for the given datasets.
2. Analyze various datasets and perform Data Pre-processing.
3. Apply various data mining algorithms on the give data set to select the appropriate one.
4. Develop skills and apply data mining tools for solving practical problems.
5. Handling a small data mining project for a given practical domain.

**LIST OF EXPERIMENTS:**

1. Study of WEKA tool and applying data mining techniques on following data sets in ARFF or CSV file Format.
2. Implementation / Usage of WEKA for classification of datasets such as customer's data, weather forecasting data, agricultural data etc.
3. Experiment to summarize and visualization of various datasets.
4. Experiment to demonstrate various data pre-processing techniques
5. Experiment to select prominent feature subsets of various datasets.
6. Experiment to Evaluate Information Gain of an attribute in the student database
7. Demonstration of classification rule process using j48 decision tree algorithm
8. Demonstration of classification rule process using ID3 decision tree algorithm
9. Experiment to predict the class using the Bayesian classification
10. Experiment to predict the class using the k-Nearest Neighbor classification
11. Experiment to implement weight & bias updating using the Back propagation Neural Network
12. Demonstration of clustering process using k-means algorithm
13. Demonstration of mining frequent patterns using Apriori algorithm
14. Demonstration of mining frequent patterns using FP-Growth algorithm
15. Experiment to compare the performance of various data mining algorithms on the give database.



| III Year I Semester | COMPILER DESIGN LAB | L | T | P | C   |
|---------------------|---------------------|---|---|---|-----|
|                     |                     | 0 | 0 | 3 | 1.5 |

**Course Objectives:**

The objectives of this course is to acquire knowledge on the

1. Implementation of a compiler for a basic language
2. Lex/Yacc specifications for designing frontend of a compiler
3. Generate MIPS instruction set

**Course Outcomes:**

The students should be able to:

1. Understand the different phases of compilation and the working of compilers like gcc, clang etc
2. Implement lexical analyzer for any language
3. Implement parser for any language
4. Implement 3-address code generator for simple programming constructs
5. Implement MIPS code generator by considering simple programming constructs

**List of experiments**

1. Check the output of different compilers gcc, g++, clang, clang++, javac, python etc by running respective language programs with different flags. (purpose to understand preprocessor, optimizations, linker)
2. The Language called TinyCStr is described as follows
  - a) Every TinyCStr program has one or more functions and syntax of function declaration and function definition is similar to C, one function must be main.
  - b) Every TinyCStr function has zero or statements
  - c) The possible statements are declaration, assignment, conditional statements (if, else, for, while) except switch.
  - d) TinyCStr supports primitive data types of C and a string datatype
    - i. Implement a lexical analyser for TinyCStr using flex/lex
    - ii. Implement a parser for TinyCStr using bison/yacc and generate AST (Abstract Syntax Tree)
    - iii. Generate a 3-address code from the AST
    - iv. Generate MIPS instructions from 3-address code and run it on SPIM simulator
3. Write a program illustrating code optimization techniques:
  - i) Constant folding
  - ii) Copy propagation
  - iii) Common sub expression elimination
  - iv) Loop unrolling
  - v) Dead code elimination

**Text Books:**

- i. flex & bison by John Levine Released August 2009 Publisher(s): O'Reilly Media, Inc.
- ii. Compilers: Principles, Techniques and Tools, Second Edition, Alfred V. Aho, Monica S. Lam, Ravi Sethi, Jeffrey D. Ullman, Pearson.

**Reference Books:**

- i. LLVM Cookbook, Mayur pandey

**E-resources:**



**JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY GURJADA VIZIANAGARAM**

**VIZIANAGARAM – 535 003, Andhra Pradesh, India**

**B.Tech CSE (R23-COURSE STRUCTURE & SYLLABUS)**

- i. <https://llvm.org/>
- ii. <https://gcc.gnu.org/>
- iii. [https://www.dsi.unive.it/~gasparetto/materials/MIPS\\_Instruction\\_Set.pdf](https://www.dsi.unive.it/~gasparetto/materials/MIPS_Instruction_Set.pdf)



|                            |                                                                  |          |          |          |          |
|----------------------------|------------------------------------------------------------------|----------|----------|----------|----------|
| <b>III Year I Semester</b> | <b>FULL STACK DEVELOPMENT - 2<br/>(SKILL ENHANCEMENT COURSE)</b> | <b>L</b> | <b>T</b> | <b>P</b> | <b>C</b> |
|                            |                                                                  | <b>0</b> | <b>1</b> | <b>2</b> | <b>2</b> |

### **Course Objectives:**

The main objectives of the course are to

1. Make use of router, template engine and authentication using sessions to develop application in ExpressJS.
2. Build a single page application using RESTful APIs in ExpressJS
3. Apply router and hooks in designing ReactJS application

### **Course Outcomes:**

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

1. Apply core concepts of ExpressJS to build web applications
2. Design and develop web applications using ExpressJS by connecting to MongoDB with Mongoose
3. Creating reusable components using both function-based and class-based approaches.
4. Implement dynamic user interfaces in ReactJS
5. Install and configure MongoDB (including Atlas) and perform essential database operations

### **Experiments covering the Topics:**

1. ExpressJS – Routing, HTTP Methods, Middleware, Templating, Form Data
2. ExpressJS – Cookies, Sessions, Authentication, Database, RESTful APIs
3. ReactJS – Render HTML, JSX, Components – function & Class, Props and States, Styles, Respond to Events
4. ReactJS – Conditional Rendering, Rendering Lists, React Forms, React Router, Updating the Screen
5. ReactJS – Hooks, Sharing data between Components, Applications – To-do list and Quiz
6. MongoDB – Installation, Configuration, CRUD operations, Databases, Collections and Records

### **Sample Experiments:**

#### **1. ExpressJS – Routing, HTTP Methods, Middleware.**

- a. Write a program to define a route, Handling Routes, Route Parameters, Query Parameters and URL building.
- b. Write a program to accept data, retrieve data and delete a specified resource using http methods.
- c. Write a program to show the working of middleware.

#### **2. ExpressJS – Templating, Form Data**

- a. Write a program using templating engine.
- b. Write a program to work with form data.



**3. ExpressJS – Cookies, Sessions, Authentication**

- a. Write a program for session management using cookies and sessions.
- b. Write a program for user authentication.

**4. ExpressJS – Database, RESTful APIs**

- a. Write a program to connect MongoDB database using Mongoose and perform CRUD operations.
- b. Write a program to develop a single page application using RESTful APIs.

**5. ReactJS – Render HTML, JSX, Components – function & Class**

- a. Write a program to render HTML to a web page.
- b. Write a program for writing markup with JSX.
- c. Write a program for creating and nesting components (function and class).

**6. ReactJS – Props and States, Styles, Respond to Events**

- a. Write a program to work with props and states.
- b. Write a program to add styles (CSS & Sass Styling) and display data.
- c. Write a program for responding to events.

**7. ReactJS – Conditional Rendering, Rendering Lists, React Forms**

- a. Write a program for conditional rendering.
- b. Write a program for rendering lists.
- c. Write a program for working with different form fields using react forms.

**8. ReactJS – React Router, Updating the Screen**

- a. Write a program for routing to different pages using react router.
- b. Write a program for updating the screen.

**9. ReactJS – Hooks, Sharing data between Components**

- a. Write a program to understand the importance of using hooks.
- b. Write a program for sharing data between components.

**10. MongoDB – Installation, Configuration, CRUD operations**

- a. Install MongoDB and configure ATLAS
- b. Write MongoDB queries to perform CRUD operations on document using insert(), find(), update(), remove()

**11. MongoDB – Databases, Collections and Records**

- a. Write MongoDB queries to Create and drop databases and collections.
- b. Write MongoDB queries to work with records using find(), limit(), sort(), createIndex(), aggregate().

**12. Augmented Programs: (Any 2 must be completed)**

- a. Design a to-do list application using NodeJS and ExpressJS.



- b. Design a Quiz app using ReactJS.
- c. Complete the MongoDB certification from MongoDB University website.

**Text Books:**

- i. Pro MERN Stack: Full Stack Web App Development with Mongo, Express, React, and Node, Vasana Subramanian, 2<sup>nd</sup> edition, APress, O'Reilly.
- ii. Node.js in Action, Mike Cantelon, Mark Harter, T.J. Holowaychuk, Nathan Rajlich, Manning Publications. (Chapters 1-11)
- iii. React Quickly, Azat Mardan, Manning Publications (Chapters 1-8, 12-14)

**Web Links:**

- i. ExpressJS - <https://www.tutorialspoint.com/expressjs>
- ii. ReactJS - <https://www.w3schools.com/REACT> (and) <https://react.dev/learn#>
- iii. MongoDB - <https://learn.mongodb.com/learning-paths/introduction-to-mongodb>



| III Year I Semester | TINKERING LAB<br>(ENGINEERING SCIENCE) | L | T | P | C |
|---------------------|----------------------------------------|---|---|---|---|
|                     |                                        | 0 | 0 | 2 | 1 |

### Course Objectives:

1. The Tinkering Lab course aim so foster an environment where students can explore, experiment, and innovate through hands-on learning.
2. The lab encourages creativity, problem-solving, prototyping, and teamwork.
3. It provides interdisciplinary exposure to mechanical, electronic, and digital fabrication tools, sensors, IoT, coding, and product design.

### Course Outcomes:

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

1. Apply knowledge from various engineering disciplines to build and test prototypes
2. Use tools, equipment, and software for basic mechanical, electronic, and digital fabrication.
3. Understand the basics of product design, prototyping, and iteration
4. Work collaboratively on real-world problem statements and inter disciplinary projects
5. Develop problem-solving skills through tinkering and experimentation

### Module– I

**Introduction to Tinkering:** The principles of creativity, innovation, and tinkering. What is tinkering? Importance of creativity and experimentation in engineering, Overview of Tinkering Labs: Spaces, tools, and possibilities, Safety and best practices in the lab.

#### Lab Work

- Simple individual projects using hand tools(e.g., wood/metal cutting, joining,etc.).
- Demonstrations of sample projects from different engineering streams.

### Module– II

**Introduction to Mechanical Prototyping:** Basic mechanical fabrication tools and techniques: 3D modeling and CAD tools (Introduction to software like Solid Works, AutoCAD), Introduction to 3D printing and laser cutting, Using hand tools, drills, and mechanical prototyping kits.

#### Lab Work

- Create simple mechanical prototypes using 3D printing and basic fabrication tools.
- Build and test basic structural models using wood, plastic, and metal materials.
- Introduction to Electronics Prototyping: Basic electronics and circuit design. Introduction to bread boarding, resistors, capacitors, transistors, and ICs., Use of Arduino/Raspberry Pi or similar microcontrollers, Working with sensors and actuators, Introduction to simple programming for hardware integration.

#### Lab Work

- Build a simple electronic project like an automatic light sensor or basic robot.
- Implement hands-on projects using Arduino, basic wiring, and circuit design.
- Digital Fabrication and IoT: Digital tools for fabrication and integrate IoT-based systems, Introduction to IoT: Sensors, communication protocols (Bluetooth, Wi-Fi, etc.),Working with embedded systems, IoT platforms, and cloud-based services, Introduction to digital fabrication



techniques (Laser cutting, CNC machines).

### **Lab Work**

- Build a basic IoT project(e.g.,smart home device or weather station using sensors).
- Fabricate components of IoT projects using3D printers or laser cutters.

### **Module– III**

**Design Thinking and Ideation:** Process of design thinking for problem-solving, Steps of design thinking: Empathize, Define, Ideate, Prototype, and Test, Case studies of successful product design and innovation, Understanding user needs and translating them into tangible solutions.

### **Lab Work**

- Brain storm ideas for final projects.
- Work in teams to develop multiple solutions to a given challenge.

### **Module– IV**

**Rapid Prototyping and Iteration:** Prototype and test ideas iteratively, Concepts of rapid prototyping: fail-fast, iterate often, Tools for prototyping across disciplines (CAD, electronics simulation, 3D printing), Testing and evaluating prototypes for functionality, usability, and cost.

### **Lab Work**

- Create initial prototypes for the final project.
- Conduct iterative testing and modifications.

### **Module– V**

**Collaborative Final Project:** Application of interdisciplinary knowledge to solve real-world problems, Work in interdisciplinary teams(mix of mechanical, electronics, computer science,etc.), Solving real-world problems by building a functional prototype, Document project progress, design decisions, and iterations.

### **Lab Work**

- Develop and finalize a working prototype that addresses a selected challenge.
- Present the project to peers and faculty for evaluation.

### **Text Books**

- i. Designing for Growth: A Design Thinking Tool Kit for Managers by Jeanne Liedtka and Tim Ogilvie.
- ii. The Art of Tinkering by Karen Wilkinson and Mike Petrich.
- iii. Make: Electronics: Learning by Discovery by Charles Platt.

### **Reference Links**

- i. Arduino, Raspberry Pi, and IoT tutorial platforms.
- ii. 3D modeling and CAD tutorials on platforms like Tinker cad or Fusion360.
- iii. MIT's Fab Lab and Fab Academy resources.

### **Tools and Equipment:**

- i. 3D printers, laser cutters, CNC machines.
- ii. Arduino and Raspberry Pi kits.
- iii. Basic electronic components: sensors, LEDs, motors, breadboards.
- iv. Mechanical tools: drills, saws, hand tools,etc.



| III Year II Semester | COMPUTER NETWORKS | L | T | P | C |
|----------------------|-------------------|---|---|---|---|
|                      |                   | 3 | 0 | 0 | 3 |

### Course Objectives:

1. To introduce the fundamental various types of computer networks.
2. To understand state-of-the-art in network protocols, architectures, and applications.
3. To explore the various layers of OSI Model.

### Course Outcomes:

The students are able to

1. Understand OSI and TCP/IP reference models with an emphasis to Physical Layer, Data Link Layer and Network Layer.
2. Analyze the issues related to data link, medium access and transport layers by using channel allocation and connection management schemes. Analyze MAC layer protocols and LAN technologies.
3. Solve problems related to Flow control, Error control, Congestion control and Network Routing.
4. Design and compute subnet masks and addresses for networking requirements.
5. Understand how internet works,

### UNIT-I:

**Introduction:** Network Hardware and software Reference models- The OSI Reference Model- the TCP/IP Reference Model - A Comparison of the OSI and TCP/IP Reference Models, Examples of Networks: Novell Networks, Arpanet, Internet, Network Topologies WAN, LAN, MAN.

**Physical Layer:** Guided Transmission Media, Digital Modulation and Multiplexing: frequency division multiplexing, wave length division multiplexing, synchronous time division multiplexing, statistical time division multiplexing.

### UNIT-II:

**The Data Link Layer** - Design Issues, Services Provided to the Network Layer – Framing – Error Control – Flow Control, Error Detection and Correction – Error-Correcting Codes – Error Detecting Codes, Elementary Data Link Protocols, Sliding Window Protocols.

**Channel allocation methods:** TDM, FDM, ALOHA, Carrier sense Multiple access protocols, Collision Free protocols – IEEE standard 802 for LANS – Ethernet, Token Bus, Token ring, Bridges and IEEE 802.11 and 802.16. Data link layer switching, virtual LANs.

### UNIT-III:

**Network layer Routing Algorithms:** Design Issues, Routing Algorithms-Shortest path, Flooding, Flow based Distance vector, Link state, Hierarchical, Broadcast routing, Congestion Control algorithms-General principles of congestion control, Congestion prevention policies, Choke packets, Load shedding, and Jitter Control.

**Internet Working:** Tunneling, internetworking, Fragmentation, Network layer in the internet

– IP protocols, IP address, Subnets, Internet control protocols, OSPF, BGP, Internet multicasting, Mobile IP, IPV6.



#### UNIT IV:

**The Transport Layer:** Elements of transport protocols – addressing, establishing a connection, releasing connection, flow control and buffering and crash recovery, End to end protocols: UDP, Real Time Transport Protocol.

**The Internet Transport Protocol: TCP-** reliable Byte Stream (TCP) end to end format, segment format, connection establishment and termination, sliding window revisited, adaptive retransmission, TCP extension, Remote Procedure Call.

#### UNIT – V:

**Application Layer:** WWW and HTTP: Architecture- Client (Browser), Server, Uniform Resource Locator HTTP: HTTP Transaction, HTTP Operational Model and Client/Server Communication, HTTP Generic Message Format, HTTP Request Message Format, HTTP Response Message Format.

**The Domain Name System:** The DNS Name Space, Resource Records, Name Servers, Electronic Mail: Architecture and Services, The User Agent, Message Formats, Message Transfer, Final Delivery.

#### Text Books:

- i. Data Communications and Networks – Behrouz A. Forouzan, Third Edition TMH.
- ii. Computer Networks, 5ed, David Patterson, Elsevier
- iii. Computer Networks: Andrew S Tanenbaum, 4th Edition. Pearson Education/PHI
- iv. Computer Networks, Mayank Dave, CENGAGE

#### References:

- i. Tanenbaum and David J Wetherall, Computer Networks, 5th Edition, Pearson Edu, 2010
- ii. Computer Networks: A Top Down Approach, Behrouz A. Forouzan, Firouz Mosharraf, McGraw Hill Education
- iii. An Engineering Approach to Computer Networks-S.Keshav, 2nd Edition, Pearson Education
- iv. Understanding communications and Networks, 3rd Edition, W.A. Shay, Thomson The TCP/IP Guide, by Charles M. Kozierok,
- v. Free online Resource, <http://www.tcpipguide.com/free/index.htm>



| III Year II Semester | ARTIFICIAL INTELLIGENCE | L | T | P | C |
|----------------------|-------------------------|---|---|---|---|
|                      |                         | 3 | 0 | 0 | 3 |

### Course Objectives:

1. To have a basic proficiency in a traditional AI language including an ability to write simple to intermediate programs and an ability to understand code written in that language
2. To have an understanding of the basic issues of knowledge representation and blind and heuristic search, as well as an understanding of other topics such as minimax, resolution, etc. that play an important role in AI programs
3. To have a basic understanding of some of the more advanced topics of AI such as learning, natural language processing, agents and robotics, expert systems, and planning.

### Course Outcomes:

1. Outline problems that are amenable to solution by AI methods, and which AI methods may be suited to solving a given problem
2. Apply the language/framework of different AI methods for a given problem
3. Implement basic AI algorithms- standard search algorithms or dynamic programming
4. Design and carry out an empirical evaluation of different algorithms on problem formalization, and state the conclusions that the evaluation supports
5. Design Expert Systems using fuzzy logic theory

### UNIT- I:

**Introduction:** history, intelligent systems, foundations of AI, applications, tic-tac-toe game playing, development of AI languages, current trends.

### UNIT -II:

**Problem solving:** state-space search and control strategies: Introduction, general problem solving, characteristics of problem, exhaustive searches, heuristic search techniques, iterative deepening A\*, constraint satisfaction.

**Problem reduction and game playing:** Introduction, problem reduction, game playing, alpha beta pruning, two-player perfect information games.

### UNIT –III:

**Logic concepts:** Introduction, propositional calculus, propositional logic, natural deduction system, axiomatic system, semantic tableau system in propositional logic, resolution refutation in propositional logic, predicate logic.

### UNIT -IV:

**Knowledge representation:** Introduction, approaches to knowledge representation, knowledge representation using semantic network, extended semantic networks for KR, knowledge representation using frames.

**Advanced knowledge representation techniques:** Introduction, conceptual dependency theory, script



structure, CYC theory, case grammars, semantic web

**UNIT–V:**

**Expert system and applications:** Introduction phases in building expert systems, expert system versus traditional systems Uncertainty measure: probability theory: Introduction, probability theory, Bayesian belief networks, certainty factor theory, dempster-shafer theory ,Fuzzy sets and fuzzy logic: Introduction, fuzzy sets, fuzzy set operations, types of membership functions, multi valued logic, fuzzy logic, linguistic variables and hedges, fuzzy propositions, inference rules for fuzzy propositions, fuzzy systems.

**Text Books:**

- i. Artificial Intelligence- Saroj Kaushik, CENGAGE Learning
- ii. Artificial intelligence, A modern Approach , 2nded, Stuart Russel, Peter Norvig, PEA

**Reference Books:**

- i. Artificial Intelligence- Deepak Khemani, TMH, 2013
- ii. Introduction to Artificial Intelligence, Patterson, PHI
- iii. Artificial intelligence, structures and Strategies for Complex problem solving, -George F Luger, 5thed, PEA

**e-Resources:**

- i. <https://nptel.ac.in/courses/106/105/106105077/>
- ii. <http://aima.cs.berkeley.edu/>



| III Year II Semester | CRYPTOGRAPHY & NETWORK<br>SECURITY | L | T | P | C |
|----------------------|------------------------------------|---|---|---|---|
|                      |                                    | 3 | 0 | 0 | 3 |

### Course Objective:

This course aims at training students to master the:

1. The concepts of classical encryption techniques and concepts of finite fields and number theory
2. Working principles and utilities of various cryptographic algorithms including secret key cryptography, hashes and message digests, and public key algorithms
3. Design issues and working principles of various authentication protocols and PKI standards

### Course Outcomes:

Upon completion of the course, it is expected that student will be able to:

1. Identify information security goals and acquire fundamental knowledge on the concepts of finite fields and number theory
2. Compare and apply different encryption and decryption techniques to solve problems related to confidentiality and authentication
3. Apply the knowledge of cryptographic checksums and evaluate the performance of different message digest algorithms for verifying the integrity of varying message sizes.
4. Apply different digital signature algorithms to achieve authentication and create secure applications
5. Apply network security basics, analyze different attacks on networks and evaluate the performance of security protocols like SSL, IPSec, and PGP

### UNIT- I:

**Introduction to Security:** Security Attacks, Security Services, Security Mechanisms, Fundamental Security Design Principles, Attack Surfaces and Attack Trees, a Model for Network Security

**Mathematics of Cryptography:** Algebraic Structures (Groups, Rings, Fields and Galois Fields), Divisibility and the Division Algorithm, The Euclidean Algorithm, Modular Arithmetic, Prime Numbers, Fermat's and Euler's Theorems, Testing for Primality, The Chinese Remainder Theorem, Discrete Logarithms

### UNIT- II:

**Classical Encryption Techniques:** Symmetric Cipher Model, Substitution Techniques, Transposition Techniques, Rotor Machines, Steganography

**Block Ciphers:** Traditional Block Cipher Structure, The Data Encryption Standard, The Strength of DES, Block Cipher Design Principles, Advanced Encryption Standard, AES Structure, AES Transformation Functions, AES Key Expansion, Multiple Encryption and Triple DES, Block Cipher Modes of Operation



**UNIT- III:**

**Public-Key Cryptography:** Principles of Public-Key Cryptosystems, The RSA Algorithm, Diffie-Hellman Key Exchange, Elgamal Cryptographic System, Elliptic Curve Cryptography

**Cryptographic Hash Functions:** Applications of Cryptographic Hash Functions, Requirements and Security, Secure Hash Algorithm (SHA)

**Message Authentication Codes:** Requirements for Message Authentication Codes, HMAC, CMAC

**UNIT- IV:**

**Digital Signatures:** Digital Signatures, Elgamal Digital Signature Scheme, Schnorr Digital Signature Scheme, NIST Digital Signature Algorithm, Elliptic Curve Digital Signature Algorithm

**Key Management and Distribution:** Symmetric Key Distribution Using Symmetric Encryption, Symmetric Key Distribution Using Asymmetric Encryption, Distribution of Public Keys, X.509 Certificates, Public-Key Infrastructure

**User Authentication:** Remote User-Authentication Principles, Remote User-Authentication Using Symmetric Encryption, Kerberos, Remote User-Authentication Using Asymmetric Encryption:

**UNIT -V:**

**Transport-Level Security:** Web Security Considerations, Transport Layer Security, Secure Shell (SSH)

**Electronic Mail Security:** S/MIME, Pretty Good Privacy

**IP Security:** IP Security Overview, Encapsulating Security Payload, Combining Security Associations, Internet Key Exchange

**Text Book:**

- i. Cryptography and Network Security, William Stallings, 8th Edition, Pearson Education

**Reference Books:**

- i. Cryptography, Network Security and Cyber Laws, Bernard L. Menezes, Ravinder Kumar, Cengage Learning.
- ii. Cryptography and Network Security, Behrouz A Forouzan, Debdeep Mukhopadhyaya, 3rd Edition, Mc-GrawHill.
- iii. Network Security Illustrated, Jason Albanese, Wes Sonnenreich, McGrawHill.

**E-Resources:**

- i. <https://nptel.ac.in/courses/106/105/106105031/> lecture by Dr.Debdeep Mukhopadhyay IIT Kharagpur [VideoLecture]
- ii. <https://nptel.ac.in/courses/106/105/106105162/>lecture by Dr. Sourav Mukhopadhyay IIT Kharagpur [VideoLecture]
- iii. <https://www.mitel.com/articles/web-communication-cryptography-and-network-securityweb> articles by Mitel Power Connections



| III Year II Semester | CLOUD COMPUTING<br>(PROFESSIONAL ELECTIVE-II) | L | T | P | C |
|----------------------|-----------------------------------------------|---|---|---|---|
|                      |                                               | 3 | 0 | 0 | 3 |

### Course Objective:

1. To understand the concept of cloud computing.
2. To appreciate the evolution of cloud from the existing technologies.
3. To appreciate the emergence of cloud as the next generation computing paradigm.

### Course Outcomes:

Upon completion of the course, it is expected that student will be able to:

1. Articulate the main concepts, key technologies, strengths and limitations of cloud computing.
2. Learn the key and enabling technologies that help in the development of cloud.
3. Develop the ability to understand and use the architecture of compute and storage cloud, service and delivery models.
4. Explain the core issues of cloud computing such as resource management and security.
5. Evaluate and choose the appropriate technologies, algorithms and approaches for implementation and use of cloud.

### UNIT-I:

**Introduction:** Introduction to Cloud Computing, Definition of Cloud, Evolution of Cloud Computing, Underlying Principles of Parallel and Distributed Computing, Cloud Characteristics, Elasticity in Cloud – On-Demand Provisioning.

### UNIT-II:

**Cloud Enabling Technologies:** Service Oriented Architecture, REST and Systems of Systems, Web Services, Publish-Subscribe Model, Basics of Virtualization, Types of Virtualization, Implementation Levels of Virtualization, Virtualization Structures, Tools and Mechanisms, Virtualization of CPU, Memory, I/O Devices, Virtualization Support and Disaster Recovery.

### UNIT-III:

**Cloud Architecture, Services And Storage:** Layered Cloud Architecture Design, NIST Cloud Computing Reference Architecture, Public, Private and Hybrid Clouds, IaaS, PaaS, SaaS, Architectural Design Challenges, Cloud Storage, Storage-as-a-Service, Advantages of Cloud Storage, Cloud Storage Providers, S3.

### UNIT-IV:

**Resource Management And Security In Cloud:** Inter Cloud Resource Management, Resource Provisioning and Resource Provisioning Methods, Global Exchange of Cloud Resources, Security Overview, Cloud Security Challenges, Software-as-a-Service Security, Security Governance, Virtual Machine Security, IAM, Security Standards.

### UNIT-V:



**Cloud Technologies And Advancements:** Hadoop, MapReduce, Virtual Box, Google App Engine, Programming Environment for Google App Engine, Open Stack, Federation in the Cloud, Four Levels of Federation, Federated Services and Applications, Future of Federation.

**Text Book:**

- i. Distributed and Cloud Computing, From Parallel Processing to the Internet of Things, Kai Hwang, Geoffrey C. Fox, Jack G. Dongarra, Morgan Kaufmann Publishers.
- ii. Cloud Computing: Implementation, Management and Security, Rittinghouse, John W., and James F. Ransome, CRC Press.

**Reference Books:**

- i. Mastering Cloud Computing, Rajkumar Buyya, Christian Vecchiola, S. ThamaraiSelvi, Tata McGraw Hill.
- ii. Cloud Computing - A Practical Approach, Toby Velte, Anthony Velte, Robert Elsenpeter, Tata McGrawHill.
- iii. Cloud Application Architectures: Building Applications and Infrastructure in the Cloud: Transactional Systems for EC2 and Beyond (Theory in Practice), George Reese, O'Reilly.



| III Year II Semester | CYBER SECURITY<br>(PROFESSIONAL ELECTIVE-II) | L | T | P | C |
|----------------------|----------------------------------------------|---|---|---|---|
|                      |                                              | 3 | 0 | 0 | 3 |

### Course Objectives:

The aim of the course is to

1. identify security risks and take preventive steps
2. understand the forensics fundamentals
3. understand the preservation of digital evidence

### Course Outcomes:

At the end of the course, student will be able to

1. Identify various types and classifications of cybercrimes
2. Explain various cyberattack tools and methods
3. Explain the process of cybercrime investigation
4. Demonstrate an understanding of computer forensics principles and investigation procedures
5. Analyze the provisions and challenges of the Indian IT Act

**UNIT I: Introduction to Cybercrime:** Introduction, Cyber crime: Definition and Origins of the Word, Cybercrime and Information Security, Cybercriminals, Classifications of Cybercrime, Cyber stalking, Cyber cafe and Cyber crimes, Botnets. Attack Vector, Proliferation of Mobile and Wireless Devices, Security Challenges Posed by Mobile Devices, Attacks on Mobile/Cell Phones, Network and Computer Attacks.

**UNIT II: Tools and Methods :** Proxy Servers and Anonymizers, Phishing, Password Cracking, Key loggers and Spywares, Virus and Worms, Trojan Horses and Backdoors, Steganography, Sniffers, Spoofing, Session Hijacking Buffer over flow, DoS and DDoS Attacks, SQL Injection, Buffer Overflow, Attacks on Wireless Networks, Identity Theft (ID Theft), Foot Printing and Social Engineering, Port Scanning, Enumeration.

**UNIT III: Cyber Crime Investigation:** Introduction, Investigation Tools, eDiscovery, Digital Evidence Collection, Evidence Preservation, E-Mail Investigation, E-Mail Tracking, IP Tracking, E-Mail Recovery, Hands on Case Studies. Encryption and Decryption Methods, Search and Seizure of Computers, Recovering Deleted Evidences, Password Cracking.

**UNIT IV: Computer Forensics and Investigations:** Understanding Computer Forensics, Preparing for Computer Investigations. Current Computer Forensics Tools: Evaluating Computer Forensics Tools, Computer Forensics Software Tools, Computer Forensics Hardware Tools, Validating and Testing Forensics Software, Face, Iris and Fingerprint Recognition, Audio Video Analysis, Windows System Forensics, Linux System Forensics, Graphics and Network Forensics, E-mail Investigations, Cell Phone and Mobile Device Forensics.



**UNIT V: Cyber Crime Legal Perspectives:** Introduction, Cyber crime and the Legal Land scape around the World, The Indian ITAct, Challenges to Indian Law and Cybercrime Scenario in India, Consequences of Not Addressing the Weakness in Information Technology Act, Digital Signatures and the Indian ITAct, Amendments to the Indian ITAct, Cybercrime and Punishment, Cyber law, Technology and Students: Indian Scenario.

**Text Books:**

- i. Sunit Belapure Nina Godbole “Cyber Security: Understanding Cyber Crimes, Computer Forensics and Legal Perspectives”, WILEY, 2011.
- ii. Nelson Phillips and Enfinger Steuart, “Computer Forensics and Investigations”, Cengage Learning, New Delhi, 2009.

**Reference Books:**

- i. Michael T. Simpson, Kent Backman and James E. Corley, “Hands on Ethical Hacking and Network Defence”, Cengage, 2019.
- ii. Computer Forensics, Computer Crime Investigation by John R. Vacca, Firewall Media, New Delhi.
- iii. Alfred Basta, Nadine Basta, Mary Brown and Ravinder Kumar “Cyber Security and Cyber Laws”, Cengage, 2018.

**E-Resources:**

- i. CERT-In Guidelines- <http://www.cert-in.org.in/>
- ii. <https://www.coursera.org/learn/introduction-cybersecurity-cyber-attacks> [ Online Course]
- iii. <https://computersecurity.stanford.edu/free-online-videos> [ Free Online Videos]
- iv. NickolaiZeldovich. 6.858 Computer Systems Security. Fall 2014. Massachusetts Institute of Technology: MIT OpenCourseWare, <https://ocw.mit.edu> License:Creative CommonsBY-NC-SA.



| III Year II Semester | DEVOPS<br>(PROFESSIONAL ELECTIVE-II) | L | T | P | C |
|----------------------|--------------------------------------|---|---|---|---|
|                      |                                      | 3 | 0 | 0 | 3 |

**Course Objectives:**

1. To understand the DevOps Concepts and DevOps Tools
2. To implement automated system update and DevOps lifecycle
3. Providing the perfect security for the entire infrastructure

**Course Outcomes:**

At the end of the course, student will be able to

1. Enumerate the principles of continuous development and deployment, automation of configuration management, inter-team collaboration, and IT service agility
2. Describe DevOps & DevSecOps methodologies and their key concepts
3. Illustrate the types of version control systems, continuous integration tools, continuous monitoring tools, and cloud models
4. Set up complete private infrastructure using version control systems and CI/CD tools
5. Know about DevOps maturity model.

**UNIT- I:**

Phases of Software Development life cycle. Values and principles of agile software development.

**UNIT –II:**

Fundamentals of DevOps: Architecture, Deployments, Orchestration, Need, Instance of applications, DevOps delivery pipeline, DevOps eco system.

**UNIT –III:**

DevOps adoption in projects: Technology aspects, Agiling capabilities, Tool stack implementation, People aspect, processes.

**UNIT –IV:**

CI/CD: Introduction to Continuous Integration, Continuous Delivery and Deployment , Benefits of CI/CD, Metrics to track CICD practices

**UNIT -V:**

Devops Maturity Model: Key factors of DevOps maturity model, stages of Devops maturity model, DevOps maturity Assessment

**Text Books:**

- i. The DevOps Handbook: How to Create World-Class Agility, Reliability, and Security in Technology Organizations is considered the DevOps bible. It is written by Gene Kim, Jez Humble, Patrick Debois, and John Willis
- ii. Continuous Delivery: Reliable Software Releases Through Build, Test, and Deployment Automation, is by Jez Humble and David Farley
- iii. Effective DevOps: Building A Culture of Collaboration, Affinity, and Tooling at Scale by Jennifer Davis & Ryn Daniels.



**References/Web Links:**

- i. <https://www.udacity.com/course/intro-to-devops--ud611> - Good online course with sample exercises.
- ii. <http://www.edureka.co/devops> - Online Training covering high level process and tools. (Needs Registration)
- iii. [https://www.edx.org/course?search\\_query=devops](https://www.edx.org/course?search_query=devops) – Has no. of courses from MS and Redhat.
- iv. <https://www.codementor.io/devops/tutorial> - Basic Tutorial on DevOps.
- v. <https://mva.microsoft.com/training-topics/devops#!lang=1033> – Lists no. of courses related to DevOps and various tools, methods used.
- vi. <http://devops.com/> - A good blog, has lots of contents.
- vii. <https://dzone.com/devops-tutorials-tools-news> - Lots of links and tutorials



| III Year II Semester | NoSQL DATABASES<br>(PROFESSIONAL ELECTIVE-II) | L | T | P | C |
|----------------------|-----------------------------------------------|---|---|---|---|
|                      |                                               | 3 | 0 | 0 | 3 |

### Course Objectives:

The objective of this course is to

1. Distinguish the different types of NoSQL databases.
2. Explore the emergence, requirements and benefits of a NoSQL database.
3. Understand the basic architecture and data models of a NoSQL database (key-value stores, document databases, column-family stores, graph databases).

### Course Outcomes:

On completion of this course, the students will be able to

1. Differentiate between various non-relational (NoSQL) database.
2. Create Document oriented NoSQL databases using MongoDB.
3. Create Column- oriented NoSQL databases using Apache HBASE.
4. Create NoSQL Key/Value databases using Riak.
5. Create Graph NoSQL databases using Neo4.

### UNIT-I:

#### Introduction and Basic Concepts:

Overview, and History of NoSQL Databases, Definition of the Four Types of NoSQL Database, The Value of Relational Databases, Impedance Mismatch, Application and Integration Databases, Attack of the Clusters, The Emergence of NoSQL, Key Points, Comparison of relational databases to new NoSQL stores, MongoDB, Cassandra, HBASE, Neo4j use and deployment, Application, RDBMS approach, Challenges NoSQL approach, Key-Value and Document Data Models, Column-Family Stores, Aggregate-Oriented Databases.

### UNIT-II:

#### NoSQL Key/Value databases using MongoDB:

**Document** Databases, What Is a Document Database? Features, Suitable Use Cases, Event Logging, Content Management Systems, Blogging Platforms, Web Analytics or Real-Time Analytics, E-Commerce Applications, When Not to Use, Complex Transactions Spanning Different Operations, Queries against Varying Aggregate Structure.

### UNIT-III:

#### Column- oriented NoSQL databases using Apache HBASE:

Column-oriented NoSQL databases using Apache Cassandra, Architecture of HBASE, What Is a Column-Family Data Store? Features, Consistency, Transactions, Availability, Query Features,



Scaling, Suitable Use Cases, Event Logging, Content Management Systems, Blogging Platforms, Counters, Expiring Usage, When Not to Use.

#### **UNIT-IV:**

##### **NoSQL Key/Value databases using Riak:**

Key-Value Databases, What Is a Key-Value Store, Key-Value Store Features, Consistency, Transactions, Query Features, Structure of Data, Scaling, Suitable Use Cases, Storing Session Information, User Profiles, Preferences, Shopping Cart Data, When Not to Use, Relationships among Data, Multioperation Transactions, Query by Data, Operations by Sets.

#### **UNIT-V:**

##### **Graph NoSQL databases using Neo4:**

NoSQL database development tools and programming languages, Graph Databases, What Is a Graph Database? Features, Consistency, Transactions, Availability, Query Features, Scaling, Suitable Use Cases, Connected Data, Routing, Dispatch, and Location-Based Services, Recommendation Engines, When Not to Use.

#### **Text Books:**

- i. NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence, Pramod J. Sadalage, Martin Fowler, Pearson Education, 2013.
- ii. Shashank Tiwari. Professional NoSQL. John Wiley and Sons. ISBN:978-0-470-94224-6.

#### **Reference Books:**

- i. A Guide to Modern Databases and the NoSQL Movement Edition, Redmond, E. & Wilson
- ii. Redmond, E. & Wilson, J. (2012). Seven Databases in Seven Weeks: A Guide to Modern Databases and the NoSQL Movement (1<sup>st</sup> Ed.). Raleigh, NC: The Pragmatic Programmers, LLC.
- iii. Dan Sullivan. NoSQL for Mere Mortals. Addison-Wesley Professional. 2015.
- iv. Guy Harrison. Next-Generation Databases. Apress. 2016.



|                      |                                                            |   |   |   |   |
|----------------------|------------------------------------------------------------|---|---|---|---|
| III Year II Semester | SOFTWARE PROJECT MANAGEMENT<br>(PROFESSIONAL ELECTIVE-III) | L | T | P | C |
|                      |                                                            | 3 | 0 | 0 | 3 |

### Course Objectives:

The objectives of this course is to acquire knowledge on the

- 1.To study how to plan and manage projects at each stage of the software development life cycle(SDLC)
- 2.To train software project managers and other individuals involved in software project planning and tracking and oversight in the implementation of the software project management process.
- 3.To understand successful software projects that support organization's strategic goals.

### Course Outcomes:

The students should be able to:

- 1.Estimate overall cost of a software project.
- 2.Explain software development process.
- 3.Distinguish workflows of process.
- 4.Design project organization structure & analyze quality.
- 5.Estimate effort and schedule needed for project.

### UNIT-I:

**Conventional Software Management:** The Waterfall Model, Conventional Software Management Performance.

**Evolution Of Software Economics:** Software Economics, Pragmatic Software Cost Estimation.

**Improving Software Economics:** Reducing Software Product Size, Improving Software Processes, Improving Team Effectiveness, Improving Automation through Software Economics.

### UNIT-II:

**The Old Way and the New:** The Principles of Conventional Software Engineering, The Principles of Modern Software Management, Transitioning to an Iterative Process.

**Life Cycle Phases:** Engineering and Production Stages, Inception Phase, Elaboration Phase, Construction Phase, Transition Phase.

### UNIT-III:

**Model Based Software Architectures:** A Management Perspective, A Technical Perspective.

**Workflows of the Process:** Software Process Workflows, Iteration Workflows.

**Iterative Process Planning:** Work Breakdown Structures, Planning Guidelines, The Cost and Schedule Estimating Process, The Iteration Planning Process.



**UNIT-IV:**

**Project Organization and Responsibilities:** Line-Of-Business Organizations, Project Organizations, Evolution of Organizations.

**Project Control and Process Instrumentation:** The Seven Core Metrics, Management Indicators, Quality Indicators Modern Project Profiles. The COCOMO Cost Estimation Model: COCOMO.

**UNIT-V:**

**Effort Estimation and Scheduling:** Effort Estimation, Scheduling.

**Quality Planning:** Quality Concepts, Quantitative Quality Management Planning. RISK MANAGEMENT: Risk Assessment, Risk Control.

**Textbooks:**

- i. Walker Royce, “Software Project Management – A UnifiedFramework”, 1stEdition, Pearson Education,2002.
- ii. PankajJalote, “Software Project Management in Practice”, 1stEdition, Pearson Education, 2005.
- iii. Software Project Management, Bob Hughes & Mike Cotterell, TATAMcgraw-Hill.

**References:**

- i. Bob Hughes, “Mike Cotterell, Rajib Mall, Software Project Management”, 5thEdition, McGraw-Hill Higher Education,2011.
- ii. Joel Henry, “Software Project Management”, 1st Edition, Pearson Education,2006.
- iii. Norman E. Fenton,Shari Lawrence Pfleeger, “Software Metrics: A Rigorous and Practical Approach “, 1st Edition, PWS Publishing Company,1997



| III Year II Semester | HUMAN COMPUTER INTERACTION<br>(PROFESSIONAL ELECTIVE-III) | L | T | P | C |
|----------------------|-----------------------------------------------------------|---|---|---|---|
|                      |                                                           | 3 | 0 | 0 | 3 |

**Course Objectives:**

1. Describe typical human–computer interaction (HCI) models, styles, and various historic HCI paradigms.
2. Apply an interactive design process and universal design principles to designing HCI systems.
3. Describe and use HCI design principles, standards and guidelines.

**Course Outcomes:**

1. Explain the capabilities of both humans and computers from the viewpoint of human information processing.
2. Describe typical human–computer interaction (HCI) models, styles, and various historic HCI paradigms.
3. Apply an interactive design process and universal design principles to designing HCI systems.
4. Describe and use HCI design principles, standards and guidelines.
5. Analyze and identify user models, user support, socio-organizational issues, and stakeholder requirements of HCI systems.

**UNIT I:**

**Introduction:** Usability of Interactive Systems- introduction, usability goals and measures, usability motivations, universal usability, goals for our profession  
**Managing Design Processes:** Introduction, Organizational design to support usability, Four pillars of design, development methodologies, Ethnographic observation, Participatory design, Scenario Development, Social impact statement for early design review, legal issues, Usability Testing and Laboratories

**UNIT II:**

Menu Selection, Form Fill-In and Dialog Boxes: Introduction, Task- Related Menu Organization, Single menus, Combinations of Multiple Menus, Content Organization, Fast Movement Through Menus, Data entry with Menus: Form Fill-in, dialog Boxes, and alternatives, Audio Menus and menus for Small Displays

**UNIT III:**

**Command and Natural Languages:** Introduction, Command organization Functionality, Strategies and Structure, Naming and Abbreviations, Natural Language in Computing  
**Interaction Devices:** Introduction, Keyboards and Keypads, Pointing Devices, Speech and Auditory Interfaces, Displays- Small and large.

**UNIT IV:**



**Quality of Service:** Introduction, Models of Response-Time impacts, Expectations and attitudes, User Productivity, Variability in Response Time, Frustrating Experiences Balancing Function and Fashion: Introduction, Error Messages, No anthropomorphic Design, Display Design, Web Page Design, Window Design, Color.

#### **UNIT V:**

##### **User Documentation and Online Help:**

Introduction, Online Vs Paper Documentation, Reading from paper Vs from Displays, Shaping the content of the Documentation, Accessing the Documentation, Online tutorials and animated documentation, Online communities for User Assistance, The Development Process.

**Information Search:** Introduction, Searching in Textual Documents and Database Querying, Multimedia Document Searches, Advanced Filtering and Searching Interfaces Information Visualization: Introduction, Data Type by Task Taxonomy, Challenges for Information Visualization

##### **Text Books:**

- i. Designing the User Interface, Strategies for Effective Human Computer Interaction, 5ed, Ben Shneiderman, Catherine Plaisant, Maxine Cohen, Steven M Jacobs, Pearson
- ii. The Essential guide to user interface design, 2/e, Wilbert O Galitz, Wiley DreamaTech.

##### **Reference Books**

- i. Human Computer, Interaction Dan R.Olsan, Cengage ,2010.
- ii. Designing the user interface. 4/e, Ben Shneidermann , PEA.
- iii. User Interface Design, Soren Lauesen , PEA.
- iv. Interaction Design PRECE, ROGERS, SHARPS, Wiley.



|                      |                                                   |   |   |   |   |
|----------------------|---------------------------------------------------|---|---|---|---|
| III Year II Semester | BIG DATA ANALYTICS<br>(Professional Elective-III) | L | T | P | C |
|                      |                                                   | 3 | 0 | 0 | 3 |

### Course Objectives:

The objectives of this course is to acquire knowledge on the

1. Necessity of Big data analysis and challenges in Big data analysis
2. Descriptive, Predictive, Real time analysis of bigdata
3. Programming tools PIG & HIVE in Hadoop echo system

### Course Outcomes:

The students should be able to:

1. Understand and Illustrate characteristics of big data and big data challenges in different domains including social media, transportation, finance and medicine
2. Demonstrate stream processing on real time applications
3. Do Big data processing using Map reduce on Hadoop
4. Do Big data processing using PIG scripts and HiveQLqueries
5. Understand Predictive analysis of bigdata.

**UNIT-I: Introduction:** Introduction to big data: Introduction to Big Data platform, Challenges of conventional systems, Intelligent data analysis, Nature of data, Analytic processes and tools, Analysis vs Reporting.

**UNIT - II: Stream Processing:** Mining data streams: Introduction to Streams Concepts, Stream Data Model and Architecture, Stream Computing, Sampling Data in a Stream, Filtering Streams, Counting Distinct Elements in a Stream, Estimating Moments, Counting Oneness in a Window, Decaying Window, Real time Analytics Platform (RTAP) Applications, Case Studies - Real Time Sentiment Analysis - Stock Market Predictions.

**UNIT - III: Introduction to Hadoop:** Hadoop: History of Hadoop, the Hadoop Distributed File System, Components of Hadoop Analysing the Data with Hadoop, Scaling Out, Hadoop Streaming, Design of HDFS, Java interfaces to HDFS Basics, Developing a Map Reduce Application, How Map Reduce Works, Anatomy of a Map Reduce Job run, Failures, Job Scheduling, Shuffle and Sort, Task execution, Map Reduce Types and Formats, Map Reduce Features Hadoop environment.

**UNIT - IV: Frameworks and Applications:** Frameworks: Applications on Big Data Using Pig and Hive, Data processing operators in Pig, Hive services, HiveQL, Querying Data in Hive, fundamentals of HBase and ZooKeeper.

**UNIT - V: Predictive Analytics and Visualizations:** Predictive Analytics, Simple linear regression, Multiple linear regression, Interpretation of regression coefficients, Visualizations, Visual data analysis techniques, interaction techniques, Systems and application



**JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY GURJADA  
VIZIANAGARAM**

**VIZIANAGARAM – 535 003, Andhra Pradesh, India**

**B.Tech CSE (R23-COURSE STRUCTURE & SYLLABUS)**

**Text Books:**

- i. Tom White, “Hadoop: The Definitive Guide”, Third Edition, O’reilly Media, Fourth Edition, 2015.
- ii. Chris Eaton, Dirk DeRoos, Tom Deutsch, George Lapis, Paul Zikopoulos, “Understanding Big Data: Analytics for Enterprise Class Hadoop and Streaming Data”, McGrawHill Publishing, 2012.
- iii. Anand Rajaraman and Jeffrey David Ullman, “Mining of Massive Datasets”, CUP, 2012

**Reference Books:**

- i. Bill Franks, “Taming the Big Data Tidal Wave: Finding Opportunities in Huge Data Streams with Advanced Analytics”, John Wiley & sons, 2012.
- ii. Paul Zikopoulos, Dirk deRoos, Krishnan Parasuraman, Thomas Deutsch, James Giles, David Corrigan, “Harness the Power of Big Data: The IBM Big Data Platform”, Tata McGraw Hill Publications, 2012.
- iii. Arshdeep Bahga and Vijay Madisetti, “Big Data Science & Analytics: A Hands On Approach”, VPT, 2016.
- iv. Bart Baesens, “Analytics in a Big Data World: The Essential Guide to Data Science and its Applications (WILEY Big Data Series)”, John Wiley & Sons, 2014.

**E-resources:**

- i. <https://www.edx.org/course/big-data-fundamentals>
- ii. <https://hadoop.apache.org/>
- iii. <https://pig.apache.org/>
- iv. <https://hive.apache.org/>



| III Year II Semester | DISTRIBUTED SYSTEM<br>(PROFESSIONAL ELECTIVE-III) | L | T | P | C |
|----------------------|---------------------------------------------------|---|---|---|---|
|                      |                                                   | 3 | 0 | 0 | 3 |

### Course Objectives:

1. To understand the foundations of distributed systems.
2. To learn issues related to clock Synchronization, the need for global state and remote invocation in distributed systems.
3. To learn distributed mutual exclusion and deadlock detection algorithms.

### Course Outcomes:

At the end of the course, the students will be able to:

1. Understand the foundations and issues of distributed systems.
2. Illustrate the various synchronization issues, global state and remote invocation for distributed systems.
3. Develop the Mutual Exclusion and Deadlock detection algorithms in distributed systems.
4. Apply the features of peer-to-peer, distributed shared memory systems and security.
5. Analyze the distributed transactions, agreement protocols and fault tolerance mechanisms in distributed systems.

### UNIT- I:

**Characterization of Distributed Systems:** Introduction, Examples of Distributed Systems, Resource Sharing and the Web, Challenges, Relation to Computer system Components, Motivation, Relation to Parallel Systems, Message-Passing systems versus Shared Memory systems, Primitives for Distributed Communication, Synchronous versus Asynchronous executions, Design issues and Challenges. **A model of Distributed Computations:** A distributed program, A model of distributed executions, Models of communication networks, Global state, Cuts, Past and future cones of an event, Models of Process Communications. **Logical Time:** A framework for a system of logical clocks, Scalar time, Vector time, Physical clock synchronization:NTP.

### UNIT –II:

**Message Ordering and Group Communication:** Message ordering paradigms, Asynchronous execution with synchronous communication, Synchronous program order on an asynchronous system, Group communication, Causal order (CO), Total order.

**Global state and Snapshot Recording Algorithms:** Introduction, System model and definitions, Snapshot algorithms for FIFO channels. **Remote Invocation:** Introduction, Design Issues for RMI, Implementation of RMI, Distributed Garbage Collection, Remote Procedure Call, Events and Notifications, Case Study: JAVARMI.

### UNIT- III:

**Distributed Mutual Exclusion Algorithms:** Introduction, Preliminaries, Lamport's algorithm, Ricart-Agrawala algorithm, Maekawa's algorithm, Suzuki-Kasami's broadcast algorithm. **Deadlock Detection in Distributed Systems:** Introduction, System model, Preliminaries, Models of deadlocks, Knapp's Classification, Algorithms for the Single Resource Model, the AND model and the ORmodel.

### UNIT -IV:



**Peer-to-Peer Computing and Overlay Graphs:** Introduction, Data indexing and overlays, Chord distributed hash table, Content addressable networks, Tapestry. **Distributed Shared Memory:** Abstraction and advantages, Memory consistency models, Shared Memory Mutual Exclusion.

**Security:** Introduction, Overview of Security Techniques, Cryptographic Algorithms, Digital Signatures, Cryptography Pragmatics.

#### UNIT –V:

**Distributed Transactions:** Introduction, Flat and Nested Distributed Transactions, Atomic commit protocols, Concurrency control in distributed transactions. **Check Pointing and Rollback Recovery:** Introduction, Background and definitions, Issues in Failure recovery, Checkpoint-based recovery, Log-based rollback recovery, coordinated check pointing algorithm, Algorithms for asynchronous and synchronous check pointing and recovery. **Consensus and Agreement Algorithms:** Problem definition, Overview of results, Agreement in a Failure-Free system (synchronous or asynchronous).

#### Text Books:

- i. Distributed computing: Principles, algorithms, and systems, Ajay D Kshemkalyani and Mukesh Singhal, Cambridge University Press, 2011.
- ii. Distributed Systems Concepts and Design, George Coulouris, Jean Dollimore and Tim Kindberg, 5<sup>th</sup> Edition, Pearson Education, 2012.

#### Reference Books:

- i. Distributed Operating Systems: Concepts and Design, Pradeep K Sinha, Prentice Hall of India, 2007.
- ii. Advanced concepts in operating systems. Mukesh Singhal and Niranjana G. Shivaratri, McGraw-Hill, 1994.
- iii. Distributed Systems: Principles and Paradigms, Tanenbaum A.S., Van Steen M., Pearson Education, 2007.

#### E-Resources:

- i. <https://nptel.ac.in/courses/106/106/106106168/>



| III Year II Semester | WEB SERVICES<br>(OPEN ELECTIVE-II) | L | T | P | C |
|----------------------|------------------------------------|---|---|---|---|
|                      |                                    | 3 | 0 | 0 | 3 |

**Course Objective:**

1. To Understand XML concepts
2. To Understand paradigms needed for testing Web Services
3. To explore different Test Strategies for SOA-based applications

**Course Outcomes:** At the end of the Course, the Student will be able to:

1. Describe and explain the relationship among HTML, XHTML, CSS, JavaScript, XML and other web technologies.
2. Create and publish advanced HTML pages with the help of frames, scripting languages, and CSS.
3. Understand and use JavaScript variables, control structures, functions, arrays, and objects. Understand and develop XML Technologies such as XML Schemas, XSLT.
4. Understand and develop Server-Side Programming using Servlets and JSP's.
5. Develop web pages using AJAX and PHP

**UNIT-I:**

**INTRODUCTION TO HTML5:** New HTML5 Form input Types, Introduction to Cascading Style Sheets: Part 1: Inline Styles, Embedded Style Sheets, Conflicting Styles, and Linking External Style Sheets (Text Book: 1)

**JAVA SCRIPT:** Introduction to scripting, Control Structures-I, Control Structures-II, Functions, Arrays, Objects. (Text Book: 1).

**UNIT-II :**

**XML:** Introduction, XML Basics, Structuring Data, XML Namespaces, Document Type Definitions (DTDs), W3C XML Schema Documents, XML Vocabularies, Extensible Style sheet Language and XSL Transformations, Document Object Model (DOM): Objects and Collections (Text Book : 1).

**UNIT-III:**

**JDBC AND SERVLETS: DATABASE ACCESS:** Overview of JDBC, JDBC Drivers,

Connecting to a Database, the Statement Interfaces, Result Sets, Using Metadata (Text Book: 3) **SERVLETS:** The Life Cycle of a Servlet, Using Tomcat for Servlet Development, A Simple Servlet, The Servlet API, The javax.servlet Package , Reading Servlet Parameters, The javax.servlet.http Package, Handling HTTP Requests and Responses, Cookies, Session Tracking. (Text Book : 2).

**UNIT-IV: JSP:**

JSP Overview, How JSP Works , A Basic Example, JSP Syntax and Semantics: The JSP Development Model, Components of a JSP Page: Directives, Comments, Expressions, Scriptlets, Declarations, implicit objects, Standard Actions, Tag Extensions, A Complete Example (Text Book: 3).

Expressions, Scriptlets, Expression and Scriptlet Handling by the JSP Container, Implicit Objects and the JSP Environment, Initialization Parameters, Request Dispatching: Anatomy of Request Processing, include Directive, The Action, Forwarding Requests, RequestDispatcher Object (Text Book:3).



**JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY GURJADA  
VIZIANAGARAM**

**VIZIANAGARAM – 535 003, Andhra Pradesh, India**

**B.Tech CSE (R23-COURSE STRUCTURE & SYLLABUS)**

**UNIT-V: PHP:**

Introduction, Simple PHP Program, Converting Between Data Types, Arithmetic Operators, Initializing and Manipulating Arrays, String Comparisons, String Processing with Regular Expressions, Form Processing and Business Logic, Reading from a Database (Text Book: 1)

**AJAX:** Traditional Web Applications vs. Ajax Applications, Rich Internet Applications (RIAs) with Ajax, History of Ajax, Ajax Example Using the XML, HttpRequest Object, Using XML and the DOM. (Reference Book: 6)

**Text Books:**

- i. Dietel and Dietel : “Internet and World Wide Web – How to Program”, 5th Edition, PHI/Pearson Education,2011
- ii. Herbert Schildt, “The complete Reference Java 2”, 9th Edition, TMH,2014.
- iii. Phil Hanna: “The Complete Reference JSP”, 2nd Edition, TMH,2008.

**References:**

- i. Hans Bergsten : “Java Server Pages”, 3rdEdition, O’Reillypublication,2008.
- ii. Raj Kamal, “Internet & Web technologies”, 8th Edition, Tata McGraw-Hill,2007.
- iii. Chris Bates, “Web Programming, building internet applications”, 2ndEdition, WILEY, Dreamtech,2008.
- iv. Xavier. C, “Web technology and design”, 1stEdition, New Age International,2011.
- v. Marty Hall and Larry Brown, “Core servlets and java Server pages volume 1: core technologies”, 2nd Edition, Pearson Education,2007.
- vi. Thomas A Powel, “The Complete Reference: AJAX”, 1st Edition, Tata McGraw Hill,2008.

**Web References:**

- i. [www.w3schools.com](http://www.w3schools.com)
- ii. [www.tutorialspoint.com](http://www.tutorialspoint.com)



| III Year II Semester | COMPUTER NETWORKS<br>(OPEN ELECTIVE-II) | L | T | P | C |
|----------------------|-----------------------------------------|---|---|---|---|
|                      |                                         | 3 | 0 | 0 | 3 |

**Course Objectives:**

1. To introduce the fundamental various types of computer networks.
2. To understand state-of-the-art in network protocols, architectures, and applications.
3. To explore the various layers of OSI Model.

**Course Outcomes:**

The students can

1. Understand OSI and TCP/IP reference models with an emphasis to Physical Layer, Data Link Layer and Network Layer.
2. Analyze the issues related to data link, medium access and transport layers by using channel allocation and connection management schemes. Analyze MAC layer protocols and LAN technologies.
3. Solve problems related to Flow control, Error control, Congestion control and Network Routing.
4. Design and compute subnet masks and addresses for networking requirements.
5. Understand how internetworks,

**UNIT-I:**

**Introduction:** Network Hardware and software Reference models- The OSI Reference Model- the TCP/IP Reference Model - A Comparison of the OSI and TCP/IP Reference Models, Examples of Networks: Novell Networks, Arpanet, Internet, Network Topologies WAN, LAN, MAN.

**Physical Layer:** Guided Transmission Media, Digital Modulation and Multiplexing: frequency division multiplexing, wavelength division multiplexing, synchronous time division multiplexing, statistical time division multiplexing.

**UNIT-II:**

**The Data Link Layer** - Design Issues, Services Provided to the Network Layer – Framing – Error Control – Flow Control, Error Detection and Correction – Error-Correcting Codes – Error Detecting Codes, Elementary Data Link Protocols, Sliding Window Protocols.

**Channel allocation methods:** TDM, FDM, ALOHA, Carrier sense Multiple access protocols, Collision Free protocols – IEEE standard 802 for LANS – Ethernet, Token Bus, Token ring, Bridges and IEEE 802.11 and 802.16. Data link layer switching, virtual LANs.

**UNIT-III:**

**Network layer Routing Algorithms:** Design Issues, Routing Algorithms-Shortest path, Flooding, Flow based Distance vector, Link state, Hierarchical, Broadcast routing, Congestion Control algorithms-General principles of congestion control, Congestion prevention policies, Choke packets, Load shedding, and Jitter Control.

**Internet Working:** Tunnelling, internetworking, Fragmentation, Network layer in the internet

– IP protocols, IP address, Subnets, Internet control protocols, OSPF, BGP, Internet multicasting, Mobile IP, IPV6.



#### UNIT IV:

**The Transport Layer:** Elements of transport protocols – addressing, establishing a connection, releasing connection, flow control and buffering and crash recovery, End to end protocols: UDP, Real Time Transport Protocol.

**The Internet Transport Protocol: TCP-** reliable Byte Stream (TCP) end to end format, segment format, connection establishment and termination, sliding window revisited, adaptive retransmission, TCP extension, Remote Procedure Call.

#### UNIT – V:

**Application Layer:** WWW and HTTP: Architecture- Client (Browser), Server, Uniform Resource Locator HTTP: HTTP Transaction, HTTP Operational Model and Client/Server Communication, HTTP Generic Message Format, HTTP Request Message Format, HTTP Response Message Format.

**The Domain Name System:** The DNS Name Space, Resource Records, Name Servers, Electronic Mail: Architecture and Services, The User Agent, Message Formats, Message Transfer, Final Delivery.

#### Text Books:

- i. Data Communications and Networks – Behrouz A. Forouzan, Third EditionTMH.
- ii. Computer Networks, 5ed, David Patterson,Elsevier
- iii. Computer Networks: Andrew S Tanenbaum, 4th Edition. PearsonEducation/PHI
- iv. Computer Networks, Mayank Dave, CENGAGE

#### References:

- i. Tanenbaum and David J Wetherall, Computer Networks, 5th Edition, Pearson Edu, 2010
- ii. Computer Networks: A Top Down Approach, Behrouz A. Forouzan, FirouzMosharraf, McGraw Hill Education
- iii. An Engineering Approach to Computer Networks-S.Keshav, 2nd Edition, PearsonEducationUnderstanding communications and Networks, 3rd Edition, W.A. Shay, ThomsonThe TCP/IP Guide, by Charles M. Kozierok, Free online Resource,<http://www.tcpipguide.com/free/index.htm>.



**JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY GURJADA  
VIZIANAGARAM**

**VIZIANAGARAM – 535 003, Andhra Pradesh, India**

**B.Tech CSE (R23-COURSE STRUCTURE & SYLLABUS)**

| III Year II Semester | OBJECT ORIENTED PROGRAMMING<br>THROUGH JAVA<br>(OPEN ELECTIVE-II) | L | T | P | C |
|----------------------|-------------------------------------------------------------------|---|---|---|---|
|                      |                                                                   | 3 | 0 | 0 | 3 |

**Course Objectives:**

1. The objective of the course is to teach the basic concepts and techniques which form the object oriented programming paradigm
2. To understand skills in writing programs using exception handling techniques and multithreading
3. Well equipped with Java SDK environment to create, debug and run simple Java programs

**Course Outcomes:**

By the end of this course the student will be able to:

1. Understand the structure and model of the Java programming language
2. Explain the concept of class and objects with access control to represent real world entities
3. Demonstrate the implementation of inheritance and polymorphism.
4. Illustrate different techniques on creating and accessing packages (fully qualified name and import statements)
5. Understand the impact of exception handling to avoid abnormal termination of program using checked and unchecked exceptions

**Unit – I**

**Introduction to Java:** Evolution of Java, Java Buzzwords, The Java Virtual Machine, An overview of Java- Simple Java Program, Naming Conventions in Java, Data types, Variables, Expressions, Automatic type Conversion, Operators, Control Statements , Arrays, Strings

**Unit – II**

**Classes & Objects:** Class fundamentals, Declaring Objects, Initializing the instance variables, Access Control, Constructors, Methods in Java, Overloading Methods and constructors, Static Methods, Recursion, final keyword, this keyword, garbage collection, finalize() method

**Unit – III**

**Inheritance:** Inheritance Basics, Types of Inheritance, use of ‘super’, ‘final’ keywords in inheritance.

**Polymorphism:** Method Overriding, Dynamic Method Dispatch, Abstract Classes.

**Interfaces:** Def, Multiple Inheritance using Interface, Abstract Classes vs. Interfaces

**Unit – IV**

**Packages:** Packages, Different Types of Packages, Access Protection, Importing Packages.

**Exception Handling:** Exception-handling basics, throw Clause, throws Clause. Types: Built-in Exception, User Defined Exception.

**Unit – V**

**Threads:** Java Thread Model, Main Thread, Creating a Thread and Running it, terminating the Thread, Creating Multiple Threads, Thread Synchronization, and Thread Priorities.



**JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY GURJADA  
VIZIANAGARAM**

**VIZIANAGARAM – 535 003, Andhra Pradesh, India**  
**B.Tech CSE (R23-COURSE STRUCTURE & SYLLABUS)**

**Text Books:**

1. Herbert Schildt, “Java The complete reference”, 9thEdition, McGrawHill, 2017.
2. Timothy Budd, “An introduction to object-oriented programming”, 3rdEdition, Pearson Education, 2009.

**Reference Books**

- 1 E.Balaguruswamy, “Programming with Java A Primer”, 5th Edition, TataMcGraw-Hill, 2017.
2. Y. Daniel Liang, “Introduction to Java programming”, 9thEdition, Pearson education, 2012

**Reference Link**

1. [http://en.wikibooks.org/wiki/Java\\_Programming](http://en.wikibooks.org/wiki/Java_Programming) - Java Learning WikiBook
2. <http://www.javabeginner.com> - Java Beginner Tutorial



| III Year II Semester | FUNDAMENTALS OF CLOUD<br>COMPUTING<br>(OPEN ELECTIVE-II) | L | T | P | C |
|----------------------|----------------------------------------------------------|---|---|---|---|
|                      |                                                          | 3 | 0 | 0 | 3 |

**Course Objective:**

1. To provide students with the fundamentals and essentials of Cloud Computing.
2. To provide students a sound foundation of the Cloud Computing so that they are able to start using and adopting Cloud Computing services and tools in their real life scenarios.
3. To enable students exploring some important cloud computing driven commercial systems and applications.

**Course Outcomes**

Upon completion of the course, it is expected that student will be able to:

1. Understand and analyze different computing paradigms
2. Understand the basics of cloud computing and different cloud deployment models.
3. Understand different cloud implementation and management strategies.
4. Understand and evaluate different cloud service models.
5. Identify, analyze and use different cloud services/applications/tools available from key cloud providers.

**UNIT-I:**

**Computing Paradigms:** High-Performance Computing, Parallel Computing, Distributed Computing, Cluster Computing, Grid Computing, Cloud Computing, Bio computing, Mobile Computing, Quantum Computing, Optical Computing, Nano computing..

**UNIT-II:**

**Cloud Computing Fundamentals:** Motivation for Cloud Computing, The Need for Cloud Computing, Defining Cloud Computing, Definition of Cloud Computing, Cloud Computing is a Service, Cloud Computing is a Platform, Principles of Cloud computing, Five Essential Characteristics, Four Cloud Deployment Models

**UNIT-III:**

**Cloud Computing Architecture and Management:** Cloud architecture, Layer, Anatomy of the Cloud, Network Connectivity in Cloud Computing, Applications on the Cloud, Managing the Cloud, Managing the Cloud Infrastructure, Managing the Cloud Application, Migrating Application to Cloud, Phases of Cloud Migration Approaches for Cloud Migration.

**UNIT-IV:**

**Cloud Service Models:** Infrastructure as a Service, Characteristics of IaaS. Suitability of IaaS, Pros and Cons of IaaS, Summary of IaaS Providers, Platform as a Service, Characteristics of PaaS, Suitability of PaaS, Pros and Cons of PaaS, Summary of PaaS Providers, Software as a Service, Characteristics of SaaS, Suitability of SaaS, Pros and Cons of SaaS, Summary of SaaS Providers, Other Cloud Service Models.

**UNIT-V:**

**Cloud Providers and Applications:** EMC, EMC IT, Captiva Cloud Toolkit, Google Cloud Platform, Cloud



**JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY GURJADA  
VIZIANAGARAM**

**VIZIANAGARAM – 535 003, Andhra Pradesh, India**

**B.Tech CSE (R23-COURSE STRUCTURE & SYLLABUS)**

Storage, Google Cloud Connect, Google Cloud Print, Google App Engine, Amazon Web Services, Amazon Elastic Compute Cloud, Amazon Simple Storage Service, Amazon Simple Queue service, Microsoft, Windows Azure, Microsoft Assessment and Planning Toolkit, SharePoint, IBM, Cloud Models, IBM Smart Cloud, SAP Labs, SAP HANA Cloud Platform, Virtualization Services Provided by SAP, Sales force, Sales Cloud, Service Cloud: Knowledge as a Service, Rack space, VMware, Manjra soft, Aneka Platform.

**Text Book:**

- i. Essentials of Cloud Computing, K. Chandrasekhran, CRCpress.

**Reference Books:**

- i. Cloud Computing: Principles and Paradigms, Rajkumar Buyya, James Broberg and Andrzej M. Goscinski, Wiley.
- ii. Distributed and Cloud Computing, Kai Hwang, Geoffery C. Fox, Jack J. Dongarra, Elsevier.
- iii. Cloud Security and Privacy: An Enterprise Perspective on Risks and Compliance, Tim Mather, SubraKumaraswamy, ShahedLatif, O'Reilly.



| III Year II Semester | FUNDAMENTALS OF INFORMATION<br>SECURITY<br>(OPEN ELECTIVE-II) | L | T | P | C |
|----------------------|---------------------------------------------------------------|---|---|---|---|
|                      |                                                               | 3 | 0 | 0 | 3 |

### Course Objective:

1. Formulate information security objectives of privacy (confidentiality), data integrity, authentication, and non-repudiation.
2. Describe the fundamental axioms and concepts in abstract algebra and number theory that form the foundation of network and computer security solutions.
3. Design and implement cryptosystems based on the design principles of symmetric-key and asymmetric-key algorithms.

### Course Outcomes

Upon completion of the course, it is expected that student will be able to:

1. Understand the basics and need for information security
2. Identify, analyze, and evaluate infrastructure and network vulnerabilities.
3. Understand and analyze different access control and authentication methods.
4. Identify and assess current and anticipated security risks and vulnerabilities with vulnerability assessment and auditing methods.
5. Learn the fundamentals of cryptography and how cryptography serves as the central language of information security..

### UNIT-I:

**Introduction to Security:** Challenges of Securing Information, Definition of Information Security, Attackers, Attacks and Defenses.

**Systems Threats and Risks:** Software-Based Attacks, Hardware-Based Attacks, Attacks on Virtualized Systems, Hardening the Operating System, Preventing Attacks that Target the Web Browser, Hardening Web Servers, Protecting Systems from Communications-Based Attacks, Applying Software Security Applications.

### UNIT-II:

**Network Vulnerabilities and Attacks:** Network Vulnerabilities, Categories of Attacks, Methods of Network Attacks.

**Network Defences:** Crafting a Secure Network, Applying Network Security Devices, Host and Network Intrusion Prevention Systems (HIPS/NIPS), Protocol Analyzers, Internet Content Filters, Integrated Network Security Hardware.

### UNIT-III:

**Access Control:** Access Control Models and Practices, Logical Access Control Methods, Physical Access Control.

**Authentication:** Definition of Authentication, Authentication Credentials, Extended Authentication Protocols, Remote Authentication and Security.



**JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY GURJADA  
VIZIANAGARAM**

**VIZIANAGARAM – 535 003, Andhra Pradesh, India**  
**B.Tech CSE (R23-COURSE STRUCTURE & SYLLABUS)**

**UNIT-IV:**

**Vulnerability Assessment:** Risk Management, Assessment, and Mitigation, Identifying Vulnerabilities.

**Security Audit:** Privilege Auditing, Usage Auditing, Monitoring Methodologies and Tools.

**UNIT-V:**

**Cryptography:** Introduction to Cryptography, Cryptographic Algorithms, Using Cryptography on Files and Disks, Digital Certificates, Public Key Infrastructure, Key Management.

**Text Book:**

- i. Security+ Guide to Network Security Fundamentals, Third Edition, Mark Ciampa, Cengage Learning.

**Reference Books:**

- i. Principles of Information Security, Michael E. Whitman and Herbert J. Mattord, Cengage Learning.
- ii. Information Security: The Complete Reference, Rhodes-Ousley, Mark, Second Edition, McGraw-Hill.
- iii. Information Security: Principles and Practices, Mark S. Merkow, Jim Breithaupt, 2nd Edition, Pearson Education



| III Year II Semester | AI TOOLS LAB | L | T | P | C   |
|----------------------|--------------|---|---|---|-----|
|                      |              | 0 | 0 | 3 | 1.5 |

### Course Objectives:

This course is introduced to

1. Learn the fundamentals of most widely used Python packages NumPy, Pandas and Matplotlib, and then apply them to Data Analysis and Data Visualization projects.
2. To introduce the fundamental techniques and principles of Neural Networks
3. Teach students the leading trends and systems in natural language processing

**Course Outcomes:** Upon the successful completion of this course, students will be able to

1. Apply the tools of AI in the field of Engineering.
2. Identify the deep learning algorithms which are more appropriate for various types of learning tasks in various domains.
3. design and implement solutions to classification, regression, and clustering problems
4. Implement deep learning algorithms and solve real-world problems
5. Understand machine learning techniques used in NLP, including hidden Markov models and probabilistic context-free grammars, clustering and unsupervised methods

### List of Experiments:

1. **Numpy:** Illustrate the concepts multi-dimensional arrays and matrices, along with a large library of high-level mathematical functions to operate on these arrays using numpy
2. **Pandas:** Visualize New York Motor Vehicle Crash Data Using Python, Pandas, and Matplotlib.

### Datasets Details:

<https://data.ny.gov/Transportation/Motor-Vehicle-Crashes-Case-Information-Three-Year-/e8ky-4vqe>

<https://data.ny.gov/Transportation/Motor-Vehicle-Crashes-Individual-Information-Three/ir4y-sesj>

<https://data.ny.gov/Transportation/Motor-Vehicle-Crashes-Violation-Information-Three-/abfj-y7uq>

<https://data.ny.gov/Transportation/Motor-Vehicle-Crashes-Vehicle-Information-Three-Ye/xe9x-a24f>

3. **Tensor-Flow:** Learn simple data curation by creating a pickle with formatted datasets for training, development and testing in Tensor Flow and develop visualizations in tensorboard.
4. Create convolutional neural networks in TensorFlow.
5. **Image recognition** (or image classification) : identifying images and categorizing them in one of several predefined distinct classes using neural network models.
6. **OpenCV:** Develop an online writing Whiteboard with minimal features for online classes
7. **Keras:** Recognize handwritten digits from MNIST using Keras
8. **Scikit-learn :** Write a Python program using Scikit-learn to split the iris dataset into 70% train data and 30% test data. Out of total 150 records, the training set will contain 120 records and the test set contains 30 of those records. Print both datasets
9. Design a perceptron classifier to classify handwritten numerical digits (0-9). Implement using Scikit



**JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY GURJADA  
VIZIANAGARAM**

**VIZIANAGARAM – 535 003, Andhra Pradesh, India**

**B.Tech CSE (R23-COURSE STRUCTURE & SYLLABUS)**

or Weka.

10. **NLP:** Program to illustrate the concepts sentence segmentation, word tokenization, stemming and lemmatization, Hidden markov model(HMM) for Parts of speech (PoS)tag.

**References:**

- i. Machine Learning: The art and Science of algorithms that make sense of data, Peter Flach, Cambridge University Press, 2012
- ii. Tom M. Mitchell, Machine Learning, India Edition 2013, McGraw Hill Education
- iii. Chris Albon : Machine Learning with Python Cookbook , O'Reilly Media, Inc. 2018

**Web Resources:**

- i. [https://scikit-learn.org/stable/\\_downloads/scikit-learn-docs.pdf](https://scikit-learn.org/stable/_downloads/scikit-learn-docs.pdf)
- ii. [docs.python.org > library](https://docs.python.org/3/library/)
- iii. <https://opencv.org/>
- iv. <https://matplotlib.org/>



**JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY GURJADA  
VIZIANAGARAM**

**VIZIANAGARAM – 535 003, Andhra Pradesh, India**

**B.Tech CSE (R23-COURSE STRUCTURE & SYLLABUS)**

| III Year II Semester | COMPUTER NETWORKS LAB | L | T | P | C   |
|----------------------|-----------------------|---|---|---|-----|
|                      |                       | 0 | 0 | 3 | 1.5 |

**Course Objectives:**

1. Understand and apply different network commands
2. Analyze different networking functions and features for implementing optimal solutions Apply different networking concepts for implementing network solution
3. Implement different network protocols

**Course Outcomes:**

1. Apply the basics of Physical layer in real time applications
2. Apply data link layer concepts, design issues, and protocols
3. Apply Network layer routing protocols and IP addressing
4. Implement the functions of Application layer and Presentation layer paradigms and Protocols

**Experiments:**

- 1) Implement the data link layer framing methods such as character stuffing and bit stuffing.
- 2) Write a C program to develop a DNS client server to resolve the given hostname.
- 3) Implement on a data set of characters the three CRC polynomials – CRC-12, CRC-16 and CRC-CCIP.
- 4) Implement Dijkstra's algorithm to compute the shortest path in a graph.
- 5) Take an example subnet graph with weights indicating delay between nodes. Now obtain Routing table at each node using distance vector routing algorithm
- 6) Take an example subnet of hosts. Obtain broadcast tree for it.
- 7) Write a client-server application for chat using UDP
- 8) Implement programs using raw sockets (like packet capturing and filtering)
- 9) Write a C program to perform sliding window protocol.
- 10) Get the MAC or Physical address of the system using Address Resolution Protocol.
- 11) Simulate the Implementing Routing Protocols using border gateway protocol (BGP)
- 12) Simulate the OPEN SHORTEST PATH FIRST routing protocol based on the cost assigned to the path.
- 13) Install Wireshark Tool on PC and use into:
  - a) Capture network traffic
  - b) Determine default gateway address of your network
  - c) Examine frame format and contents of Ethernet frames
  - d) Filter and examine only ICMP traffic
  - e) Run various network services like ping, ssh, dns ..etc and examine the traffic captured by Wire shark
- 14) Simulate a three nodes point-to-point network with duplex links between them. Set the queue size vary the bandwidth and find the number of packets dropped.



**JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY GURJADA  
VIZIANAGARAM**

**VIZIANAGARAM – 535 003, Andhra Pradesh, India**

**B.Tech CSE (R23-COURSE STRUCTURE & SYLLABUS)**

- 15) Simulate a four node point-to-point network, and connect the links as follows: n0-n2, n1- n2 and n2-n3. Apply TCP agent between n0-n3 and UDP between n1-n3. Apply relevant applications over TCP and UDP agents changing the parameter and determine the number of packets by TCP/UDP.
- 16) Simulate the transmission of ping messaged over a network topology consisting of 6nodes and find the number of packets dropped due to congestion.
- 17) Simulate an Ethernet LAN using N-nodes (6-10), change error rate and data rate and compare the throughput.
- 18) Simulate an Ethernet LAN using N nodes and set multiple traffic nodes and plot congestion window for different source/destination.

\* ns2/ns3/CISCO Packet Tracet/OPNET/any other network simulator may be used for simulation experiments.



| III Year II Semester | SOFT SKILLS<br>(SKILL ENHANCEMENT COURSE) | L | T | P | C |
|----------------------|-------------------------------------------|---|---|---|---|
|                      |                                           | 0 | 1 | 2 | 2 |

**Course Objectives:**

1. To equip the students with the skills to effectively communicate in English
2. To train the students in interview skills, group discussions and presentation skills
3. To motivate the students to develop confidence

**Course Outcomes:** Upon the successful completion of this course, students will be able to

1. Apply effective communication techniques including verbal expression and non-verbal cues
2. Apply self-management skills including stress, anger, and time management
3. Demonstrate proficiency in standard operational communication methods
4. Develop job-oriented skills by effectively participating in group discussions
5. Understand the importance and types of interpersonal relationships

**UNIT – I**

**Analytical Thinking & Listening Skills:** Self-Introduction, Shaping Young Minds - A Talk by Azim Premji (Listening Activity), Self – Analysis, Developing Positive Attitude, Perception.

**Communication Skills:** Verbal Communication; Non Verbal Communication (Body Language)

**UNIT – II**

**Self-Management Skills:** Anger Management, Stress Management, Time Management, Six Thinking Hats, Team Building, Leadership Qualities

**Etiquette:** Social Etiquette, Business Etiquette, Telephone Etiquette, Dining Etiquette

**UNIT – III**

**Standard Operation Methods :** Basic Grammars, Tenses, Prepositions, Pronunciation, Letter Writing; Note Making, Note Taking, Minutes Preparation, Email & Letter Writing

**UNIT-IV**

**Job-Oriented Skills:** Group Discussion, Mock Group Discussions, Resume Preparation, Interview Skills, Mock Interviews

**UNIT-V**

**Interpersonal relationships:** Introduction, Importance, Types, Uses, Factors affecting interpersonal relationships, Accommodating different styles, Consequences of interpersonal relationships

**Text books:**

- i. Barun K. Mitra, Personality Development and Soft Skills, Oxford University Press, 2011.
- ii. S.P. Dhanavel, English and Soft Skills, Orient Blackswan, 2010.

**Reference books:**

- i. R.S. Aggarwal, A Modern Approach to Verbal & Non-Verbal Reasoning, S.Chand & Company Ltd., 2018.
- ii. Raman, Meenakshi & Sharma, Sangeeta, Technical Communication Principles and Practice, Oxford University Press, 2011.

**E-resources:**



1. [https://swayam-plus.swayam2.ac.in/courses/course-details?id=P\\_CAMBR\\_01](https://swayam-plus.swayam2.ac.in/courses/course-details?id=P_CAMBR_01)

| III Year II Semester | TECHNICAL PAPER WRITING & IPR | L | T | P | C |
|----------------------|-------------------------------|---|---|---|---|
|                      |                               | 2 | 0 | 0 | - |

**Course Objective :**

1. To develop the ability to write clear, concise, and well-structured technical documents, including reports, summaries, and proposals, suitable for academic and professional contexts.
2. To enhance understanding of grammar, syntax, and technical vocabulary necessary for effective communication in technical writing.
3. To introduce students to the process of planning, drafting, revising, and presenting technical papers, using tools like MS Word for formatting, referencing, and collaboration.

**Course Outcomes:** Upon the successful completion of this course, students will be able to

1. Demonstrate the ability to write effective technical reports by constructing clear technical sentences,
2. Apply effective drafting techniques in technical report writing
3. Demonstrate proficiency in proofreading and summarizing technical content
4. Demonstrate advanced proficiency in using Microsoft Word for technical documentation
5. Describe the process of patenting from innovation to development

**Unit I:**

**Introduction:** An introduction to writing technical reports, technical sentences formation, using transitions to join sentences, Using tenses for technical writing.

**Planning and Structuring:** Planning the report, identifying reader(s), Voice, Formatting and structuring the report, Sections of a technical report, Minutes of meeting writing.

**Unit II:**

**Drafting report and design issues:** The use of drafts, Illustrations and graphics.

**Final edits:** Grammar, spelling, readability and writing in plain English: Writing in plain English, Jargon and final layout issues, Spelling, punctuation and Grammar, Padding, Paragraphs, Ambiguity.

**Unit III:**

**Proofreading and summaries:** Proofreading, summaries, Activities on summaries. **Presenting final reports:** Printed presentation, Verbal presentation skills, Introduction to proposals and practice.

**Unit IV: Using word processor:**

Adding a Table of Contents, Updating the Table of Contents, Deleting the Table of Contents, Adding an Index, Creating an Outline, Adding Comments, Tracking Changes, Viewing Changes, Additions, and Comments, Accepting and Rejecting Changes , Working with Footnotes and Endnotes, Inserting citations and Bibliography, Comparing Documents, Combining Documents, Mark documents final and make them read only., Password protect Microsoft Word documents., Using Macros,



**JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY GURJADA  
VIZIANAGARAM**

**VIZIANAGARAM – 535 003, Andhra Pradesh, India**

**B.Tech CSE (R23-COURSE STRUCTURE & SYLLABUS)**

**Unit V:**

**Nature of Intellectual Property:** Patents, Designs, Trade and Copyright. Process of **Patenting and Development:** technological research, innovation, patenting, development. International Scenario: International cooperation on Intellectual Property

**Text Books:**

- i. KompalBansal&ParshitBansal, “Fundamentals of IPR for Beginner’s”, 1<sup>st</sup> Ed., BS Publications, 2016.
- ii. William S. Pfeiffer and Kaye A. Adkins, “Technical Communication: A Practical Approach”, Pearson.
- iii. Ramappa,T., “Intellectual Property Rights Under WTO”, 2<sup>nd</sup> Ed., S Chand, 2015.

**Reference Books:**

- i. Adrian Wallwork , English for Writing Research Papers, Springer New York Dordrecht Heidelberg London, 2011.
- ii. Day R, How to Write and Publish a Scientific Paper, Cambridge University Press(2006)

**E-resources:**

- i. <https://www.udemy.com/course/reportwriting/>
- ii. <https://www.udemy.com/course/professional-business-english-and-technical-report-writing/>
- iii. <https://www.udemy.com/course/betterbusinesswriting/>