

B. Tech. (Computer Science and Engineering)
Semester III (Second Year)

Code		Category	Course Title	Hours/Week			Total	Credits	Marks		
Department	University			L	T	P			CIE	SEE	Total
MA6301C	7-6-3-01R25	MA-BS	Probability and Statistics	3	0	0	3	3	40	60	100
CS6302C	7-6-3-02R25	CS-PC	Database Management Systems	3	0	0	3	3	40	60	100
CS6303C	7-6-3-03R25	CS-PC	Design and Analysis of Algorithms	3	0	0	3	3	40	60	100
CS6304C	7-6-3-04R25	CS-PC	Discrete Mathematical Structures	3	0	0	3	3	40	60	100
CS6305C	7-6-3-05R25	CS-PC	Digital Electronics and Logic Design	3	0	0	3	3	40	60	100
CS6306L	7-6-3-R25P1	CS-PC	Database Management Systems Lab	0	0	3	3	1.5	40	60	100
CS6307L	7-6-3-R25P2	CS-PC	Design and Analysis of Algorithms Lab	0	0	3	3	1.5	40	60	100
CS6308S	7-6-3-R25S1	CS-SC	Basic Python Programming	1	0	2	3	2	100	0	100
CS6309M	7-0-3-R25M1	CE-MC	Environmental Science	2	0	0	2	0	100	0	100
TOTAL				18	0	8	26	20	480	420	900

All interested students are advised to **Register for Minors Degree or Honors Degree** at the end of this semester by submitting filled in registration form available from the Head of the Department /Chair person, Board of Studies (UG) of the parent department.



Chairperson
(BoS UG)


 DEAN
 FACULTY OF ENGINEERING
 S. V. U. COLLEGE OF ENGINEERING
 TIRUPATI-517 502

PROBABILITY AND STATISTICS**Credits: 3****Assessment: 40+60**

Semester	L-T-P	Course Code	Branches
III	3-0-0	7-6-3-01R25	CS

Course Educational Objectives(CEOs):

1. This course aims at providing the student with the knowledge on probability and random variables.
2. To acquaint the students with different types of binomial, Poisson, normal and other distributions.
3. To provide suitable and effective method for estimation of parameters and testing of hypothesis for large samples.
4. To acquaint the students with different types of small sample tests
5. To provide knowledge on different of types of control charts. .

UNIT I: Descriptive statistics, Statistics Introduction, Population vs Sample, Collection of data, primary and secondary data, Measures of Central tendency, Measures of Variability (spread or variance) Skewness, Kurtosis.

UNIT II: Probability: probability axioms, addition law and multiplicative law of probability, conditional probability, Baye's theorem, Random variables (discrete and continuous), probability density functions, properties, mathematical expectation.

UNIT III: Probability distributions: Binomial, Poisson and Normal-their properties (Chebyshevs inequality). Approximation of the binomial distribution to normal distribution.

UNIT IV: Estimation and Testing of hypothesis, large sample tests Estimation-parameters, statistics, sampling distribution, point estimation, Formulation of null hypothesis, alternative hypothesis, the critical and acceptance regions, level of significance, two types of errors and power of the test. Large Sample Tests: Test for single proportion, difference of proportions, test for single mean and difference of means. Confidence interval for parameters in one sample and two sample problems

UNIT V: Small sample tests Student-distribution (test for single mean, two means and paired t-test), testing of equality of variances (F-test), χ^2 - test for goodness of fit, χ^2 - test for independence of attributes. Quality Control: Concept of quality of a manufactured product- Causes of variation-Principles of Shewart control charts- X-chart, R-chart, p-chart, np-chart and C-chart.



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

Textbooks:

- 1) Millerand Freunds, Probability and Statistics for Engineers, 7/e, Pearson, 2008.
- 2) S.C. Gupta and V.K. Kapoor, Fundamentals of Mathematical Statistics, 11/e, Sultan Chand & Sons Publications, 2012.

Reference Books:

1. S.Ross, A First Course in Probability, Pearson Education India, 2002.
2. W.Feller, An Introduction to Probability Theory and its Applications, 1/e, Wiley, 1968.
3. B.V.Ramana, Higher Engineering Mathematics, McGrawHill Education

Course Outcomes: At the end of the course, the student will be able to

CO1: Analyze the data quantitatively or categorically using various statistical elementary tools.

CO2: Design mathematical models involving probability, random variables and the critical Thinking in the theory of probability and its applications in real life problems.

CO3: Apply the theoretical probability distributions like binomial, Poisson, and Normal in the Relevant application areas.

CO4: Analyze to test various hypotheses included in theory and types of errors for large samples. CO5: Apply the different testing tools like t-test, F-test, chi-square test to analyze the relevant

Real life problems

Mapping of Course Outcomes with Program Outcomes:

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	2	3	3	3	1	2	1	1	2	1	3	1	3
CO2	3	2	3	3	3	1	2	1	1	2	1	3	1	3
CO3	3	2	2	3	1	1	1	1	1	2	1	3	1	3
CO4	3	2	2	3	1	1	1	1	1	2	1	3	1	3
CO5	3	3	3	3	2	1	1	1	1	2	1	3	1	3

1 – Slightly; 2 - Moderately; 3 – Substantially


Chairperson
(BoS UG)


DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

DATABASE MANAGEMENT SYSTEMS**Credits: 3****Assessment: 40+60**

Semester	L-T-P	Course Code	Branches
III	3-0-0	7-6-3-02R25	CS

Prerequisites: NIL**Course Educational Objectives(CEOs):**

1. To learn data models, conceptualize and depict a database system using ER diagram
2. To understand the internal storage structures in a physical DB design
3. To demonstrate the fundamental concepts, operation and function of different components of database systems.
4. To describe the roles of transaction processing and concurrency control in a modern DBMS.

UNIT-I

Introduction: Managing Data, File Systems versus a DBMS, Advantages of a DBMS, Storing data in a DBMS, Queries in a DBMS, Transactions, Structure of a DBMS.

Introduction to Data base design: ER diagrams, Beyond ER Design, Entities, Attributes and Entity sets, Relationships and Relationship sets, Additional features of ER Model, Conceptual Design with the ER Model, Conceptual Design for Large enterprises.

Relational Model: Introduction to the Relational Model, Integrity Constraints over Relations, Enforcing Integrity constraints, Querying relational data, Logical data base Design, Introduction to Views Destroying/ altering Tables and Views.

UNIT-II

Relational Algebra and Calculus: Relational Algebra , Relational calculus, Expressive Power of Algebra and calculus.

SQL: Form of Basic SQL Query, UNION, INTERSECT and EXCEPT, Nested Queries, Aggregate Operators, NULL values , Complex Integrity Constraints in SQL, Triggers and Active Databases, Designing Active Databases

UNIT-III

Schema Refinement and Normal Forms: Introduction, Functional Dependencies, Reasoning about FDS, Normal Forms - FIRST, SECOND, THIRD Normal forms, BCNF, Properties of Decompositions, Normalization, Schema Refinement in Data base Design, Multi valued Dependencies , FOURTH Normal Form, Join Dependencies, FIFTH Normal form, Inclusion Dependencies.

Database Application Development: Accessing Databases from Applications, Introduction to JDBC, JDBC Classes and Interfaces, SQLJ, Stored Procedures.



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

UNIT-IV

Overview of Transaction Management: ACID Properties, Transactions and Schedules, Concurrent Execution of Transactions, Lock-based Concurrency Control, Performance Locking, Transaction Support in SQL, Introduction to Crash Recovery.

Concurrency Control: 2PL, Serializability and Recoverability, Introduction to Lock Management, Lock Conversions, Dealing with Deadlocks, Specialized Locking Techniques, Concurrency Control without Locking

Crash Recovery: Introduction to ARIES, Log, Recovery related Structures, Write-Ahead Log Protocols, Checkpointing, Recovering from a System Crash, Media Recovery, Interaction with Concurrency Control

UNIT-V

Overview of Storage and Indexing: Data on External Storage, File Organization and Indexing, Index data Structures, Comparison of File Organizations, Indexes and Performance Tuning.

Indexing and Hashing: Intuitions for tree indexes, Indexed Sequential Access Method, B+ Trees: A Dynamic Index Structure, Search, Insert, Delete, Duplicates, B+ Trees in Practice, Static Hashing, Extendable Hashing, Linear Hashing, Extendable vs. Linear Hashing.

Parallel and Distributed Databases: Introduction, Architectures for Parallel Databases, Parallel Query Evaluation, Parallelizing Individual Operations, Introduction to Distributed Databases, Distributed DBMS Architectures, Storing Data in Distributed DBMS, Distributed Catalog Management, Distributed Query Processing

Text Books:

1. Raghu Ramakrishnan and Johannes Gehrke, Database Management Systems, Third Edition, McGraw-Hill, 2014.
2. C. J. Date, A. Kannan and S. Swamynathan, An Introduction to Database Systems, 8th edition, Pearson Education, 2006.

Reference Books:

1. Silberschatz A, Korth H F, and Sudarshan S, Database System Concepts, 6th edition, McGraw-Hill, 2011.
2. Ramez Elmasri and Shamkant B. Navathe, Fundamentals of Database Systems, Fourth Edition, Pearson/Addison Wesley, 2007.
3. J D Ullman, H. Garcia-Molina and J. Widom, Database Systems: The Complete Book, Prentice-Hall, 2009.
4. Jeffrey A. Hoffer, Ramesh Venkataraman, Heikki Topi, Modern Database Management, 12th edition, Pearson, 2015.


Chairperson
(BoS UG)


DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

Course Outcomes (COs):

Upon successful completion of this course, the student should be able to

1. Explain database concepts, data models, ER modeling, and relational database design principles.
2. Apply relational algebra, relational calculus, and SQL queries to retrieve and manipulate data efficiently.
3. Design normalized database schemas using functional dependencies and normal forms, and develop database applications using JDBC and stored procedures.
4. Analyze transaction management, concurrency control, and crash recovery mechanisms in DBMS.
5. Evaluate storage structures, indexing, hashing techniques, and parallel/distributed database architectures for performance optimization.

Mapping of Course Outcomes with Program Outcomes:														
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	3	2	-	-	-	-	-	-	-	-	2	2	1
CO2	3	3	2	-	2	-	-	-	-	-	-	2	3	2
CO3	3	3	3	2	2	-	-	-	1	-	-	2	3	3
CO4	3	3	2	2	-	-	-	-	-	-	-	2	2	2
CO5	3	3	2	2	2	-	-	-	-	-	-	3	3	3

1 – Slightly; 2 - Moderately; 3 – Substantially


Chairperson
(BoS UG)


DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

DESIGN AND ANALYSIS OF ALGORITHMS**Credits: 3****Assessment: 40+60**

Semester	L-T-P	Course Code	Branches
III	3-0-0	7-6-3-03R25	CS

Prerequisites: Mathematics, C Programming, Data Structures.**Course Educational Objectives(CEOs):**

1. To understand how to design an algorithm for the given problem.
2. To analyze the complexity of an algorithm in terms of time and space.
3. To get better insight on different strategies of algorithm design.

UNIT-I

Introduction: What is an Algorithm?, Algorithm Specification, Performance Analysis - Space Complexity, Time Complexity, Amortized Complexity, Asymptotic Notation (O , Ω , Θ), Practical Complexities, Performance Measurement, Randomized Algorithms: An Informal Description, Identifying the Repeated Element, Primality Testing, Advantages and Disadvantages.

Sets and Disjoint Set Union: Introduction, Union and Find Operations.

UNIT-II

Divide-and-Conquer: General Method, Defective Chess Board, Binary Search, Finding the Maximum and Minimum, Merge Sort, Quicksort, Selection, Strassen's Matrix Multiplication, Convex Hull.

UNIT-III

The Greedy Method: The General Method, Container Loading, Knapsack Problem, Tree Vertex Splitting, Job Sequencing with Deadlines, Minimum-Cost Spanning Trees, Optimal Storage on Tapes, Optimal Merge Patterns, Single-Source Shortest Paths.

Basic Traversal and Search Techniques: Techniques for Binary Trees, Techniques for Graphs, Connected Components and Spanning Trees, Biconnected Components and DFS.

UNIT-IV

Dynamic Programming: The General Method, Multistage Graphs, All Pairs Shortest Paths, Single-Source Shortest Paths: General Weights, Optimal Binary Search Trees, String Editing, 0/1-Knapsack, Reliability Design, The Traveling Salesperson Problem, Flow Shop Scheduling.

Backtracking: The General Method, The 8-Queens Problem, Sum of Subsets, Graph Coloring, Hamiltonian Cycles, Knapsack Problem.



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

UNIT-V

Branch-and-Bound: The Method, 0/1 Knapsack Problem, Traveling Salesperson, Efficiency Considerations.

NP-Hard and NP-Complete Problems: Basic Concepts, Cook's Theorem, NP-Hard Graph Problems, NP-Hard Scheduling Problems.

PRAM Algorithms: Introduction, Computational Model, Fundamental Techniques and Algorithms, Selection.

Text Books:

1. Ellis Horowitz, Sartaj Sahni, and Sanguthevar Rajasekaran, Fundamentals of Computer Algorithms, 2nd edition, Universities Press, 2008.
2. Cormen T H, Leiserson C E, Rivest R L, and Stein C, Introduction to Algorithms, 3rd edition, Prentice-Hall of India, 2009.

Reference Books:

3. Levitin A, Introduction to the Design and Analysis of Algorithms, 3rd edition, Pearson Education, 2012.
4. Goodrich M T, Tamassia R, Algorithm Design, Wiley, 2008.
5. Skiena S S, The Algorithm Design Manual, 2nd edition, Springer, 2012.
6. Heineman G T, Pollice G, Selkow S, Algorithms in a Nutshell, 2nd edition, O'Reilly, 2016.
7. Dave P H, and Dave H B, Design and Analysis of Algorithms, 2nd edition, Pearson Education, 2008.

Course Outcomes(COs):

Upon successful completion of this course, the student should be able to

1. Develop Systematically an Algorithm for Solving a Problem
2. Analyze The Time and Space Complexity of The Given Algorithm
3. Identify Algorithm Design Methodology to Solve Problems
4. To Get Better Insight on Different Strategies of Algorithm Design
5. Distinguish Between P And NP Classes Of Problems

Mapping of Course Outcomes with Program Outcomes:														
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	3	3	3	-	-	-	-	-	-	1	1	-	3
CO2	3	3	-	3	3	-	2	-	-	-	-	1	2	3
CO3	3	-	2	-	3	2	-	1	-	-	1	1	2	3
CO4	3	3	-	3	3	2	2	1	-	-	-	1	2	-
CO5	-	-	3	-	-	2	-	1	-	-	-	-	2	-

1-Slightly; 2-Moderately; 3-Substantially


Chairperson
(BoS UG)


DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

DISCRETE MATHEMATICAL STRUCTURES**Credits: 3****Assessment: 40+60**

Semester	L-T-P	Course Code	Branches
III	3-0-0	7-6-3-04R25	CS

Prerequisites: Mathematics**Course Educational Objectives(CEOs):**

This course is designed to

1. Use mathematical reasoning and logic to read, comprehend, and construct mathematical arguments and apply theorem-proving techniques.
2. Familiarize students with the fundamental concepts of sets, functions, relations, and algebraic structures required for discrete mathematical analysis.
3. Present counting techniques, recursion, recurrence relations, and number theory concepts, and demonstrate their applications in computational problems.
4. Introduce fundamental concepts of graph theory, including graph representations, connectivity, and classical graph problems.
5. Familiarize students with advanced graph concepts and algorithms such as planar graphs, graph coloring, spanning trees, minimum spanning trees, and network flow problems

UNIT I

Logic, Propositional Equivalences, Predicates and Quantifiers, Nested Quantifiers, Rules of Inference, Introduction to Proofs, Methods of Proof and Strategy.

UNIT II

Sets, Set Operations, Functions, Sequences and Summations, Introduction to Semigroups, Groups, Subgroups, Normal subgroups.

Relations and their properties, n-ary relations and their applications, Representing relations, Closures of relations, Equivalence relations, Partial orderings, Lattices.

UNIT III

Counting: Basics of Counting, Pigeonhole principle, Permutations and Combinations, Binomial Coefficients, Generalized Permutations and Combinations, Generating Permutations and Combinations.

Advanced Counting Techniques: Recurrence Relations, Solving Recurrence Relations, Divide-and-Conquer Algorithms and Recurrence Relations, Generating Functions, Inclusion-and-Exclusion and its Applications.

Number Theory and its Applications.

UNIT IV

Introduction to graphs, Graph terminology, Applications of some special graphs, Representation of graphs, Graph isomorphism.

Connectivity: Connectedness in undirected and directed graphs, Paths and Isomorphism,

Construction of reliable communication networks, Euler path, Hamilton path, Chinese postman problem, Shortest path problems, Traveling salesman problem.


Chairperson
(BoS UG)

 DEAN
 FACULTY OF ENGINEERING
 S. V. U. COLLEGE OF ENGINEERING
 TIRUPATI-517 502

UNIT V

Planar graphs, Kuratowski's theorem, Graph coloring and applications.

Introduction to trees, Application of trees, Spanning trees, Applications of backtracking,

Minimum spanning trees, Flows, Cuts, Max-flow Min-cut problem.

Text Books:

1. Kenneth H Rosen, Discrete Mathematics and its Applications, 6th edition, McGraw-Hill Companies.
2. Mott J L, Kandel A, and Baker T P, Discrete Mathematics for Computer Scientists and Mathematicians, 2nd edition, PHI, 2004.

Reference Books:

3. Malik D S, Sen M K, Discrete Mathematical Structures: Theory and Applications, Thomson Course Technology, 2004.
4. Mott J L, Kandel A, and Baker T P, Discrete Mathematics for Computer Scientists and Mathematicians, 2nd edition, PHI, 2004.
5. Kolman B, Busby R C, Ross S C, and Rehman N, Discrete Mathematical Structures, 5th edition, Pearson Education, 2006.
6. Lipschutz S, Lipson M, Discrete Mathematics, 2nd edition, TMH, 2006.

Course Outcomes (COs):

On successful completion of this course the students will be able to:

1. Verify the correctness of mathematical arguments using propositional and predicate logic, and apply appropriate proof techniques.
2. Apply set theory, relations, functions, and algebraic structures to solve discrete mathematical problems.
3. Solve problems involving counting principles, recurrence relations, generating functions, and number theory.
4. Construct and analyze graph models to solve problems related to connectivity, paths, and optimization.
5. Design and develop efficient solutions using advanced graph algorithms for real-world and network-based applications.

Mapping of Course Outcomes with Program Outcomes:														
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	3	1	–	–	–	–	–	–	–	–	2	2	1
CO2	3	2	2	–	–	–	–	–	–	–	–	1	3	2
CO3	3	3	2	1	–	–	–	–	–	–	–	2	3	2
CO4	3	3	3	2	1	–	–	–	1	–	–	2	3	3
CO5	3	3	3	2	2	–	–	–	1	1	–	2	3	3

1 – Slightly; 2 - Moderately; 3 – Substantially


Chairperson
(BoS UG)


DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

DIGITAL ELECTRONICS AND LOGIC DESIGN**Credits: 3****Assessment: 40+60**

Semester	L-T-P	Course Code	Branches
III	3-0-0	7-6-3-05R25	CS

Prerequisites: Electronics Engineering.**Course Educational Objectives(CEOs):**

1. To understand the essential knowledge on the fundamental of digital circuits
2. To understand the overview on the design principles of digital computing systems

UNIT I

Number Representation, Signed and Unsigned, Code Conversion, Review of Boolean Algebra and DeMorgan's Theorem, Sum-of-Product and Product-of-Sum forms, Canonical forms, Karnaugh maps up to 6 variables.

UNIT II

MSI devices like Comparators, Multiplexers, Encoder, Decoder, Driver & Multiplexed Display, Half and Full Adders, Subtractors, Serial and Parallel Adders, BCD Adder, Barrel shifter and ALU

UNIT III

Sequential Logic Design: Building blocks like S-R, JK and Master-Slave JK FF, Edge triggered FF, Ripple and Synchronous counters, Shift registers, Finite state machines, Design of synchronous FSM, Algorithmic State Machines charts. Designing synchronous circuits like Pulse train generator, Pseudo Random Binary Sequence generator, Clock generation

UNIT IV

Logic Families and Semiconductor Memories: TTL NAND gate, Specifications, Noise margin, Propagation delay, fan-in, fan-out, Tristate TTL, ECL, CMOS families and their interfacing, Memory elements, Concept of Programmable logic devices like FPGA. Logic implementation using Programmable Devices.

UNIT V

VLSI Design flow: Design entry: Schematic, FSM & HDL, different modeling styles in VHDL, Data types and objects, Data-flow, Behavioral and Structural Modeling, Synthesis and Simulation VHDL constructs and codes for combinational and sequential circuits.


Chairperson
(BoS UG)


DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

Text/Reference Books:

1. R.P. Jain, "Modern digital Electronics", Tata McGraw Hill, 4th edition, 2010.
2. Douglas Perry, "VHDL: Programming by Example", Tata McGraw Hill, 4th edition.
3. Brown S, and Vranesic Z, Fundamentals of Digital Logic with VHDL Design, 3rd edition, McGraw Hill, 2012.
4. Kinney L L, and Roth Jr. C H, Fundamentals of Logic Design, 7th edition, Cengage Learning, 2015.

Course outcomes(COs):

At the end of this course students will demonstrate the ability to

1. The design principles of digital computing systems
2. Design and analyze combinational logic circuits
3. Design and analyze synchronous sequential logic circuits
4. Design and implement complicated digital systems using Verilog
5. Design a VLSI circuit for an application

Mapping of Course Outcomes with Program Outcomes:														
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	2	3	3	3	-	-	-	-	-	1	2	2	3
CO2	3	3	3	1	1	-	-	-	-	-	1	3	3	3
CO3	3	2	3	2	1	-	-	-	1	-	1	2	2	3
CO4	3	2	2	3	1	-	-	-	1	-	1	3	3	3
CO5	3	3	3	3	2	-	-	-	1	1	1	3	2	3

1-Slightly; 2-Moderately; 3-Substantially


Chairperson
(BoS UG)


DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

DATABASE MANAGEMENT SYSTEMS LABORATORY**Credits: 1.5****Assessment: 40+60**

Semester	L-T-P	Course Code	Branches
III	0-0-3	7-6-3-R25P1	CS

At least 10 Experiments should be given covering the topics of the course Database Management Systems.

Course Objectives: This laboratory manual contains structured SQL-based experiments covering fundamental to advanced database concepts including schema design, queries, normalization, transactions, and performance optimization.

LIST OF EXPERIMENTS

1. Create a database and tables using SQL DDL statements.
2. Create tables with constraints: PRIMARY KEY, FOREIGN KEY, UNIQUE, NOT NULL, CHECK.
3. Alter table structures (ADD, MODIFY, DROP columns).
4. Drop, truncate, and rename tables.
5. Basic SQL queries using SELECT, WHERE, ORDER BY, DISTINCT.
6. Use operators: arithmetic, comparison, logical, BETWEEN, LIKE, IN.
7. Implement joins: INNER, LEFT, RIGHT, SELF.
8. Perform set operations: UNION, INTERSECT, EXCEPT.
9. Write nested and correlated queries.
10. Use aggregate functions with GROUP BY and HAVING.
11. Handle NULL values using IS NULL, COALESCE, IFNULL.
12. Create, query, and drop views.
13. Normalize relations up to 3NF and implement using SQL.
14. Decompose relations into BCNF and implement schema.
15. Create stored procedures for database operations.
16. Create triggers for automatic updates.
17. Demonstrate transaction control: COMMIT, ROLLBACK, SAVEPOINT.
18. Simulate concurrency and isolation levels.
19. Create indexes and compare performance.
20. Analyze execution plans using EXPLAIN.


Chairperson
(BoS UG)


DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

EXPERIMENT RECORD FORMAT

Each experiment should be written in the following format:

1. Aim
2. Objective
3. Theory
4. Procedure
5. SQL Queries
6. Output
7. Result

COURSE OUTCOMES(COs):

1. Understand relational database design principles.
2. Write SQL queries for data retrieval and manipulation.
3. Apply normalization techniques.
4. Implement transactions and concurrency control.
5. Optimize queries using indexes and analyze query execution performance.

1-Slightly; 2-Moderately; 3-Substantially

Mapping of Course Outcomes with Program Outcomes:														
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	2	3	—	2	—	—	—	—	—	—	1	3	2
CO2	3	3	2	—	3	—	—	—	—	—	—	1	3	2
CO3	3	3	2	2	—	—	—	—	—	—	—	1	2	2
CO4	3	2	3	—	3	—	—	—	1	—	—	2	3	3
CO5	3	3	2	2	2	—	—	—	—	—	—	3	3	3


Chairperson
(BoS UG)


DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

DESIGN AND ANALYSIS OF ALGORITHMS LABORATORY**Credits: 1.5****Assessment: 40+60**

Semester	L-T-P	Course Code	Branches
III	0-0-3	7-6-3-R25P2	CS

At least 10 Experiments should be given covering the topics of the course, Design and Analysis of Algorithms.

Experiment 1: Implementation and Analysis of Algorithm Performance

Objective: Write algorithms and analyze their time and space complexity using asymptotic notations (O , Ω , Θ).

Experiment 2: Amortized Analysis and Randomized Algorithms

Objective: Implement a randomized algorithm for identifying a repeated element and perform amortized complexity analysis.

Experiment 3: Disjoint Set Union (Union-Find)

Objective: Implement Union and Find operations using disjoint set data structure with path compression.

Experiment 4: Binary Search using Divide and Conquer

Objective: Implement binary search and analyze its time complexity.

Experiment 5: Merge Sort and Quick Sort

Objective: Implement merge sort and quicksort and compare their performance.

Experiment 6: Strassen's Matrix Multiplication

Objective: Implement Strassen's algorithm and compare with normal matrix multiplication.

Experiment 7: Convex Hull Problem

Objective: Implement convex hull using divide-and-conquer or Graham's scan method.

Experiment 8: Greedy Method – Knapsack Problem

Objective: Solve fractional knapsack problem using greedy strategy.

Experiment 9: Minimum Cost Spanning Tree

Objective: Implement Kruskal's or Prim's algorithm for minimum spanning tree.

Experiment 10: Job Sequencing with Deadlines

Objective: Implement job sequencing algorithm to maximize profit.


Chairperson
(BoS UG)


DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

Experiment 11: Single Source Shortest Path

Objective: Implement Dijkstra's algorithm and analyze performance.

Experiment 12: Graph Traversal Techniques

Objective: Implement BFS and DFS for connected components and spanning trees.

Experiment 13: All Pairs Shortest Paths

Objective: Implement Floyd-Warshall algorithm using dynamic programming.

Experiment 14: 0/1 Knapsack using Dynamic Programming

Objective: Solve 0/1 knapsack problem using dynamic programming approach.

Experiment 15: Traveling Salesperson Problem using Dynamic Programming

Objective: Solve TSP using dynamic programming method.

Experiment 16: Backtracking – 8 Queens Problem

Objective: Implement 8-Queens problem using backtracking.

Experiment 17: Graph Coloring using Backtracking

Objective: Solve graph coloring problem using backtracking method.

Experiment 18: Branch and Bound – 0/1 Knapsack

Objective: Solve 0/1 knapsack problem using branch and bound technique.

Experiment 19: NP-Complete Problems Study

Objective: Study Cook's Theorem and implement a basic NP-hard graph problem.

Experiment 20: PRAM Algorithms

Objective: Implement basic PRAM model algorithms and parallel selection algorithm.

Course Outcomes(COs):

Upon successful completion of this course, the student should be able to

1. Develop Systematically an Algorithm for Solving a Problem
2. Analyze The Time and Space Complexity of The Given Algorithm
3. Identify Algorithm Design Methodology to Solve Problems
4. To Get Better Insight on Different Strategies of Algorithm Design
5. Distinguish Between P And NP Classes Of Problems


Chairperson
(BoS UG)


DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

Mapping of Course Outcomes with Program Outcomes:														
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
C01	3	3	3	3	-	-	-	-	-	-	1	1	-	3
C02	3	3	-	3	3	-	2	-	-	-	-	1	2	3
C03	3	-	2	-	3	2	-	1	-	-	1	1	2	3
C04	3	3	-	3	3	2	2	1	-	-	-	1	2	-
C05	-	-	3		-	2	-	1	-	-	-	-	2	-

1-Slightly; 2-Moderately; 3-Substantially.


Chairperson
(BoS UG)


DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

BASIC PYTHON PROGRAMMING**Credits: 2****Assessment: 100+0**

Semester	L-T-P	Course Code	Branches
III	1-0-2	7-6-3-R25S1	CS

Prerequisites: NIL**Course Educational Objectives(CEOs):**

The course is designed to:

1. Python syntax and semantics and be fluent in the use of Python flow control and functions.
2. The concepts of Object-Oriented Programming as used in Python.
3. Various problems solving approaches of computer science in various Domains.
4. Various data structures like lists and dictionaries using python.
5. Introduce Python third- Party Tools for various domains.

UNIT I

Introduction to Python Programming: Features and History of Python, The Future of Python, Writing and Executing First Python Program, Literal Constants, Variables and Identifiers, Data Types, Input Operation, Comments, Reserved Words, Indentation, Operators and Expressions, Expressions in Python, Operations on Strings, Other Data Types, Type Conversion.

Decision Control Statements: Introduction to Decision Control Statements, Selection/ Conditional Branching Statements, Basic Loop Structures/Iterative Statements, Nested Loops, The break, continue and pass Statement, The else Statement used with Loops.

UNIT II

Functions and Modules: Introduction, Function Definition, Function Call, Variable Scope and Lifetime, The return statement, More on Defining Functions, Lambda Functions or Anonymous Functions, Documentation Strings, Good Programming Practices, Recursive Functions, Modules, Packages in Python, Standard Library modules, Globals(), Locals(), and Reload(), Function Redefinition.

Python Strings Revisited: Introduction, Concatenating, Appending, and Multiplying Strings, Strings are Immutable, String Formatting Operator, Built-in String Methods and Functions, Slice Operation, ord() and chr() Functions, in and not in operators, Comparing Strings, Iterating String, The String Module, Regular Expressions, Metacharacters in Regular Expression.

UNIT III

File Handling: Introduction, File Path, Types of Files, Opening and Closing Files, Reading and Writing Files, File Positions, Renaming and Deleting Files, Directory Methods.

Data Structures: Sequence, Lists, Functional Programming, Tuple, Sets, Dictionaries.

Classes and Objects: Introduction, Defining Classes, Creating Objects, Data Abstraction, Class Method and self Argument, The _init_() Method, Class Variables and Object



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

Variables, The `_del_()` Method, Other Special Methods, Public and Private Data Members, Private Methods, Calling a Class Method from Another Class Method, Built-in Functions to Check, Get, Set, and Delete Class Attributes, Built-in Class Attributes, Garbage Collection, Class Methods, Static Methods.

UNIT IV

Inheritance: Introduction, Inheriting Classes in Python, Types of Inheritance, Composition or Containership or Complex Objects, Abstract Classes and Interfaces, Metaclass.

Operator Overloading: Introduction, Implementing Operator Overloading, Reverse Adding, Overriding `_getitem_()` and `_setitem_()` Methods, Overriding the `in` Operator, Overloading Miscellaneous Functions, Overriding the `_call_()` Method.

Error and Exception Handling: Introduction to Errors and Exceptions, Handling Exceptions, Multiple Except Blocks, Multiple Exceptions in a Single Block, Except Block Without Exception, The else Clause, Raising Exceptions, Instantiating Exceptions, Handling Exceptions in Invoked Functions, Built-in and User-defined Exceptions, The finally Block, Pre-defined Clean-up Action, Re-raising Exception, Assertions in Python, Multi-threading.

UNIT V

Survey of The Most Common 3rd Party Packages: Requests, Numpy/Scipy, Matplotlib/Pyplot, Pandas, Pillow, Flask/Django/Twisted, Pep8, Scikit-Learn/Nltk, Stanford-Corenlp, Bcrypt, Beautiful Soup, and More.

GUI Design with Tkinter: Button, Canvas, Check Button, Entry, Frame, Label, List Box, Menu, Menu Button, Message, Radio Button, Scale, Scrollbar, Text Graphics with Turtle: Motion Control, Pen, Colour, Fill, Multiple Turtles, Reset and Clear.

Text Books:

1. Reema Thareja, Python Programming using problem solving approach, First Edition, Oxford University Press, 2017.
2. Mark Lutz, Learning Python, Fifth Edition, O’Rielly, 2016.

Reference Books:

3. Mark Lutz, Programming Python, Fourth Edition, O’Reilly, 2010.
4. John V.Gutttag, Introduction to Computation and Programming Using Python with Application to Understanding, PHI.
5. Allen Downey, Think Python: How to think like a Computer Scientist, Green Tea Press.
6. Paul Barry, Head First Python: A Brain-Friendly Guide, Second Edition, O’Reilly.
7. The Python Standard Library, Python 3.6.5 documentation (Web Resource)
<https://docs.python.org/3/library/>.


Chairperson
(BoS UG)


DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

Course Outcomes (COs):

Having successfully completed this course the students will be able to:

1. Understand the structure, syntax, and semantics of the Python language.
2. Interpret the concepts of Object-Oriented Programming as used in Python.
3. Demonstrate proficiency in handling Strings and File Systems.
4. Demonstrate proficiency in Error and Exception Handling
5. Implement desktop/Web-based applications using the Python programming language.

Mapping of Course Outcomes with Program Outcomes:														
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	2	3	3	3	1	2	1	1	2	1	3	2	3
CO2	3	2	3	3	3	1	2	1	1	2	1	3	2	3
CO3	3	2	2	3	1	1	1	1	1	2	1	3	2	3
CO4	3	2	2	3	1	1	1	1	1	2	1	3	2	3
CO5	3	3	3	3	2	1	1	1	1	2	1	3	3	3

1-Slightly; 2-Moderately; 3-Substantially


Chairperson
(BoS UG)


DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

ENVIRONMENTAL SCIENCE**Credits: 0****Assessment: 100+0**

Semester	L-T-P	Course Code	Branches
III	2-0-0	7-0-3-R25M1	CS

Course Educational Objectives(CEOs):

1. Introduce the scope and importance of environmental resources and their sources.
2. Impart knowledge on Ecosystem and Biodiversity.
3. Impart knowledge on causes and consequences of Environmental pollution its Global effects.
4. Familiarize with environmental issues and their management.
5. Outline different social issues and the guidelines formulated for protection of environment.

UNIT I**Environmental Studies and Natural Resources**

Definition, Scope and importance of Environment, Environmental studies, Need for public awareness

Components of Environment- Atmosphere, Hydrosphere, Lithosphere.

Renewable and Non Renewable Resources and associated problems

Water resources: Use and over utilization of surface and ground water, floods, drought, conflicts over water, dams benefits and problems.

Forest resources: Use and over exploitation, deforestation, case studies. Timber extraction, mining, dams and their effects on forests and tribal people.

Land resources: Land as a resource, land degradation, Man induced landslides, soil erosion and desertification.

Mineral resources: Use and over exploitation, Environmental effects of extracting and using mineral resources, case studies.

Food resources: World food problems, changes caused agriculture and overgrazing, effects of modern agriculture, fertilizer – pesticide problems, water logging, salinity, Case studies.

Energy resources: Growing energy needs, renewable and non renewable energy sources, use of alternate energy sources. Case studies.

Role of an individual in conservation of natural resources.

UNIT II**Ecosystem and Biodiversity**

Ecosystem - Concept of an ecosystem.

Structure and functions of an ecosystem.

Producers, consumers and decomposers.

Energy flow in the ecosystem.



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

Ecological succession.

Food chains, food webs and ecological pyramids.

Introduction, types, characteristic features, structure and function of the following ecosystem.

(a) Forest ecosystem. (b) Grassland ecosystem

(c) Desert ecosystem. (d) Aquatic ecosystem (ponds, streams, lakes, rivers, oceans, estuaries)

Biodiversity and its conservation:

Definition, genetic species and ecosystem diversity.

Biogeographically classification of India.

Value of biodiversity: consumptive use, productive use, social, ethical, aesthetic and option values.

Biodiversity at global, National and local levels.

India as a mega-diversity nation.

Hot-spots of biodiversity.

Threats to biodiversity: habitat loss, poaching of wildlife, man – wildlife conflicts.

Conservation of biodiversity: in-situ and ex-situ conservation of biodiversity.

UNIT – III

Environmental pollution and Global Effects

Definition, Causes, Effects, and control measures of (a) Air pollution (b) Water pollution (c)

Soil pollution (d) Marine pollution (e) Noise pollution (f) Thermal pollution (g) Nuclear hazards

Solid waste Management: Causes, effects and control measures of urban and industrial wastes.

Role of an individual in prevention of pollution.

Pollution case studies.

Disaster management: Floods, earthquakes, cyclone, landslides, Tsunami.

Climate change-Global warming, Acid rain, Ozone depletion.

UNIT – IV

Environment Issues and Management

Environment and Human health – Epidemic diseases, HIV/AIDS, Aviation Flue, Water Borne Diseases.

Environmental Impact Assessment, Sustainable Development, Clean Production and Clean Development Mechanisms

Environment Legislation: Environmental Protection Act, Water Act, Air Act, Wild Life Protection Act, Forest Conservation Act, Public Liability & Insurance Act, Issues involved in Enforcement of Environmental legislation.

UNIT – V

Social Issues and the Environment

Population growth, Population Explosion, Population Control, Women and Child welfare.

Urbanization, Industrialization, Development projects, Resettlement and Rehabilitation of people – Problems concerned, Case studies.


Chairperson
(BoS UG)


DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

Consumerism and Waste Products Conservation, Public Awareness, Water Conservation, Rain water harvesting, watershed management, Wasteland reclamation, Human Rights, Value education, Environmental ethics- Issues and possible solution.

Role of information Technology in Environment and Human Health.

Text Books / Reference Books :

1. AnubhaKaushik& C P Kaushik, Environmental studies, New age International Publishers, 2008
2. Benny Joseph, Environmental studies, Tata McGraw-Hill Publishers, 2005
3. M Chandra Sekhar, Environmental Science, Hi-Tech Publishers, 2004
4. Keerthinarayana and Daniel Yesudian, Principles of Environmental Sciences and Engineering , Hi-Tech Publishers, 2005
5. AmalK.Datta, Introduction to Environmental Science and Engineering, Oxford & IBH Publishing Co.Pvt.Ltd, 2000
6. SanthoshkumarGarg,RajeshawriGarg and RajniGarg, Ecological and Environmental studies, Khanna publishers, 2006
7. Gilbert M, Introduction to Environmental Engineering and Science, Masters Publication by Prentice –Hall of India Private Ltd., 1991
8. William P Cunningham and Mary Ann Cunningham, Principles of Environmental Science, Tata McGraw Hill Publishing Co.Ltd, 2002

Course Outcomes(Cos):

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

1. Acquire knowledge in diverse components of environment and natural resources.
2. Understand ecosystem and biodiversity and its conservation methods.
3. Identify and resolve the issues related to sources of different types of pollution and its global effects
4. Provide solutions to individuals, industries and government for sustainable development of natural resources
5. Apply environmental ethics in resolving social issues for protection of diversified ecosystems.

PO& PSO /CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	-	-	-	-	-	3	-	3	2	1	2	3	3	2
CO2	-	-	-	-	-	-	-	3	2	1	2	3	3	2
CO3	-	-	-	-	-	-	-	3	2	-	-	-	3	2
CO4	-	-	-	-	-	-	-	3	2	-	-	-	3	1
CO5	-	-	-	-	-	-	-	3	2	-	-	-	3	1

1 - Slightly; 2 - Moderately; 3 – Substantially


Chairperson
(BoS UG)

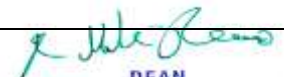

DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

B. Tech. (Computer Science and Engineering)
Semester IV (Second Year)

Code		Category	Course Title	Hours/Week			Total	Credits	Marks		
Department	University			L	T	P			CIE	SEE	Total
CS6401C	7-6-4-01R25	MA-BS	Computer Oriented Numerical Methods	3	0	0	3	3	40	60	100
CS6402C	7-6-4-02R25	CS-PC	Computer Organization	3	0	0	3	3	40	60	100
CS6403C	7-6-4-03R25	CS-PC	Object Oriented Programming Through Java	3	0	0	3	3	40	60	100
CS6404C	7-0-4-R25M2	CS-BS	Elements of Quantum Computing	2	0	0	2	2	100	0	100
CS6405C	7-0-3-01R25	MS-HS	Managerial Economics and Financial Accountancy	3	0	0	3	3	40	60	100
CS6406L	7-6-4-R25P1	CS-PC	Statistical and Numerical Methods Lab	0	0	3	3	1.5	40	60	100
CS6407L	7-6-4-R25P2	CS-PC	Assembly Language Programming and VHDL Lab	0	0	3	3	1.5	40	60	100
CS6408L	7-6-4-R25P3	CS-PC	Object Oriented Programming Through Java Lab	0	0	3	3	1.5	40	60	100
CS6409S	7-6-4-R25S1	CS-SC	Basic Web Designing	1	0	2	3	2	100	0	100
CS6410M	7-0-4-R25M1	CS-MC	Constitution of India	2	0	0	2	0	100	0	100
CS6411N		CS-NG	NSS/NSO/NCCAactivities	0	0	0	0	0	0	0	0
TOTAL				17	0	11	28	20.5	580	420	1000
All students shall mandatorily register for NCC/NSO/NSS and complete 60 hours of activities beginning from the 1 st semester but to be completed before 4 th semester in one of these three options to earn Non-graded (NG) Credits as core requirement. Grade shall be awarded as Satisfactory or Unsatisfactory by the concerned HoD on the basis of participation, attendance, performance and behaviour. If a student gets an unsatisfactory Grade, he/she shall repeat the above activity in the subsequent years, in order to complete the degree requirements.											
All interested students shall register for Minors degree or Honors degree just before or at the time of commencement of this semester by submitting filled in registration form available from the Head of the Department /Chair person, Board of Studies (UG) of the parent department.											
Students shall undergo Community Service Project (CSP) for a duration of six to eight (6-8) weeks during the summer vacation by completing 45 – 60 hours of work with at least 75% attendance, to be evaluated and reflected in 5 th semester.											



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

COMPUTER ORIENTED NUMERICAL METHODS**Credits: 3****Assessment: 40+ 60**

Semester	L-T-P	Course Code	Branches
IV	3-0-0	7-6-4-01R25	CS

Course Educational Objectives(CEOs):

1. This course aims at providing the student with the knowledge on Various numerical methods for Solving transcendental equations,.
2. To provide knowledge on fitting the different curves numerically and correlation- regression analysis
3. To acquaint the students with different types of numerical methods to interpolate the polynomials.
4. Demonstrate understanding and ability for numerical derivatives and integrals.
5. To provide suitable and effective Numerical Methods for differential equations

UNIT I

Errors in Numerical Calculations: Truncation and Round-off errors, Effect of errors in data; closed form solution versus Iterative methods. Roots of Nonlinear Equations: Bisection, False position and, Newton-Raphson methods. Solution of Linear Equations –L.U Decomposition and Relaxation methods; Convergence of iteration methods.

UNIT II

Correlation: curve fitting by method of least squares - Linear, Quadratic and Exponential fitting – Correlation – Rank, Correlation - Regression analysis.

UNIT III

Interpolation - Lagrange polynomials, Newton's forward, backward and divided difference Interpolation formula, Cubic splines and two dimensional interpolations.

UNIT IV

Numerical Differentiation - Differentiating continuous and tabulated functions, Difference tables and Richardson extrapolation. Numerical integration - Trapezoidal, Simpson's 1/3 and Simpson's 3/8 Rules.

UNIT V

Numerical Solution of Ordinary Differential Equations - Taylor's Series, Euler's, Runge- Kutta methods.



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

Text Books:

1. Schilling R J, and Harries S L, Applied Numerical Methods for Engineers Using MATLAB and C,
2. Chapra S C, Applied Numerical Methods with MATLAB for Engineers and Scientists, 2nd edition, Tata McGraw-Hill, 2007.
3. Gerald C F, and Wheatley P O, Applied Numerical Analysis, 6th edition, Pearson Education Asia, 2002.
4. Niyogi P, Numerical Analysis and Algorithms, Tata McGraw Hill, 2003.
Heath M T, Scientific Computing: An Introductory Survey, McGraw-Hill, 1997. Thomson Brooks/Cole, 2006.
5. B.S . Grewal, Higher Engineering Mathematics.
6. T. K. V. Iyengar et al, Probability and Statistics 2nd Edition, S. Chand & Company Ltd, 2010

Course Outcomes (COs):

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

1. To apply numerical methods to solve the Solution of Linear and Non-Linear Algebraic Equations and curve fitting problems, error analysis.
2. Interpret the association of characteristics and through correlation and regression tools.
3. Derive interpolating polynomials using interpolation formulae by different methods.
4. Test and evaluate the accuracy of Differentiation and integrals numerically.
5. Using numerical methods we find solution of Ordinary differential equations.

Mapping of Course Outcomes with Program Outcomes:														
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	2	3	3	2	1	1	1	1	2	1	3	1	3
CO2	3	2	3	3	2	1	1	1	1	2	1	3	1	3
CO3	3	3	3	3	2	2	2	1	1	2	1	3	1	3
CO4	3	3	3	3	2	2	2	1	1	2	1	3	1	3
CO5	3	2	2	3	2	1	2	1	1	2	1	3	1	3

1-Slightly; 2-Moderately; 3-Substantially



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

COMPUTER ORGANIZATION**Credits: 3****Assessment: 40+ 60**

Semester	L-T-P	Course Code	Branches
IV	3-0-0	7-6-4-02R25	CS

Prerequisites: Digital Electronics and Logic Design.**Course Educational Objectives (CEOs):**

This course is designed to

1. Introduce the basic structure, organization, and performance characteristics of digital computers, including instruction formats, addressing modes, and instruction sequencing.
2. Familiarize students with the design of processing units, including arithmetic operations, hardwired control, and microprogrammed control organization.
3. Develop understanding of the hierarchical memory system, including cache memory, virtual memory, and performance considerations.
4. Introduce methods of input–output organization and interfacing, including interrupts, direct memory access, and communication with peripheral devices.
5. Familiarize students with advanced processor organization, including pipelining, instruction-level parallelism, and modern processor families.

UNIT I

Structure of Computers: Introduction, Performance, Memory addressing and Operations, Instructions and Instruction sequencing, Addressing modes, Basic I/O operations, Push down stacks, Subroutines, Encoding of machine instructions, Brief description and functional classification of IA-32 Pentium instruction set

UNIT II

Basic Processing Unit: Fundamental concepts, Single and Multiple bus organization, Hardwired control, Multi-programmed control – Micro-instructions, Micro program sequencing, Wide-branch addressing, Micro instructions with next-address field, Prefetching micro-instructions.

Arithmetic: Multiplication – Booth algorithm; Integer division, Floating-Point Addition and Subtraction.

UNIT III

The memory System: Basic concepts, RAM and ROM Memories and their internal organization, Cache Memories - Mapping functions, Replacement algorithms; Performance Considerations, Virtual Memories, Secondary Storage.

UNIT IV

Input/ Output Organization: Accessing I/O devices; Interrupts –Enabling and disabling, Handling multiple devices; Direct Memory Access - Bus Arbitration; Buses – Synchronous and Asynchronous; Interface circuits – Parallel port, Serial port



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

UNIT V

Pipe lining: Basic concepts, Data hazards, Instruction hazards, Influence on instruction sets, Data path and control considerations, Super scalar operation.

Processor Families: The ARM family, The Motorola 680x0 and Cold fire families, The IA-32 family.

Text Books

1. Hamacher C, Vranesic Z, and Zaky S, Computer Organization, 5th edition, McGraw-Hill.

Reference Books

2. Heuring V P, and Jordan H F, Computer systems Design and Architecture Addison-Wesley.
3. Carpinelli J D, Computer System Organization and Architecture. Addison-Wesley 2001.
4. Mano M M, Computer system Architecture, 4th edition, Pearson Education Asian 2002.

Course Outcomes(COs):

On successful completion of this course the students will be able to:

1. Identify and analyze the structure and functional units of a digital computer, and evaluate instruction execution and addressing mechanisms.
2. Design and analyze processing unit organization, including arithmetic operations and control unit implementation.
3. Evaluate the memory hierarchy and storage organization, and analyze their impact on system performance and cost.
4. Analyze and select appropriate input–output organization and interfacing techniques for communication with peripheral devices.
5. Analyze pipelined and parallel processor architectures and evaluate their role in high-performance computing systems.

Mapping of Course Outcomes with Program Outcomes:															
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	
CO1	3	3	1	–	–	–	–	–	–	–	–	2	2	1	
CO2	3	3	3	2	1	–	–	–	–	–	–	2	3	3	
CO3	3	3	2	2	1	–	–	–	–	–	–	2	3	2	
CO4	3	3	2	1	2	–	–	–	–	–	–	2	3	2	
CO5	3	3	3	2	1	–	–	–	–	–	–	2	3	3	

1-Slightly; 2-Moderately; 3-Substantially



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

OBJECT ORIENTED PROGRAMMING THROUGH JAVA**Credits: 3****Assessment: 40+ 60**

Semester	L-T-P	Course Code	Branches
IV	3-0-0	7-6-4-03R25	CS

Prerequisites: Basic Programming Knowledge.**Course Educational Objectives(CEOs):**

1. To understand object oriented concepts and problem solving techniques
2. To obtain knowledge about the principles of inheritance and polymorphism
3. To implement the concept of packages, interfaces, exception handling and concurrency mechanism.
4. To design the GUIs using applets and swing controls.
5. To understand the Java Database Connectivity Architecture

UNIT - I

Introduction: Introduction to Object Oriented Programming, The History and Evolution of Java, Introduction to Classes, Objects, Methods, Constructors, this keyword, Garbage Collection, Data Types, Variables, Type Conversion and Casting, Arrays, Operators, Control Statements, Method Overloading, Constructor Overloading, Parameter Passing, Recursion, String Class and String handling methods.

UNIT - II

Inheritance: Basics, Using Super, Creating Multilevel hierarchy, Method overriding, Dynamic Method Dispatch, Using Abstract classes, Using final with inheritance, Object class, Packages: Basics, finding packages and CLASSPATH, Access Protection, Importing packages. Interfaces: Definition, Implementing Interfaces, Extending Interfaces, Nested Interfaces, Applying Interfaces, Variables in Interfaces.

UNIT - III

Exception handling - Fundamentals, Exception types, Uncaught exceptions, using try and catch, multiple catch clauses, nested try statements, throw, throws and finally, built-in exceptions, creating own exception subclasses. Stream based I/O (java.io) – The Stream classes-Byte streams and Character streams, Reading console Input and Writing Console Output, File class, Reading and Writing Files, Random access file operations, The Console class, Serialization, Enumerations, Autoboxing, Generics.



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

UNIT - IV

Multithreading: The Java thread model, Creating threads, Thread priorities, Synchronizing threads, Interthread communication. The Collections Framework (java.util): Collections overview, Collection Interfaces, The Collection classes, Array List, Linked List, Hash Set, Tree Set, Priority Queue, Array Deque. Hashtable, Properties, Stack, Vector, String Tokenizer, Bit Set, Date, Calendar, Random, Formatter, Scanner.

UNIT – V

Applet: Basics, Architecture, Applet Skeleton, requesting repainting, using the status window, passing parameters to applets

GUI Programming with Swings – The origin and design philosophy of swing, components and containers, layout managers, event handling, using a push button, jtextfield, jlabel and image icon, the swing buttons, jtext field, jscrollpane, jlist, jcombobox, trees, jtable, An overview of jmenubar, jmenu and jmenuItem, creating a main menu, showmessagedialog, showconfirmdialog, showinputdialog, showoptiondialog, jdialog, create a modeless dialog.

Accessing Databases with JDBC: Types of Drivers, JDBC Architecture, JDBC classes and Interfaces, Basic steps in developing JDBC applications, Creating a new database and table with JDBC.

Textbooks:

1. Java The complete reference, 9th edition, Herbert Schildt, McGraw Hill Education (India)
2. Java How to Program, 10th Edition, Paul Dietel, Harvey Dietel, Pearson Education.

Reference Books:

1. Understanding Object-Oriented Programming with Java, updated edition, T. Budd, Pearson Education.
2. Core Java Volume – 1 Fundamentals, Cay S. Horstmann, Pearson Education.
3. Java Programming for core and advanced learners, Sagayaraj, Dennis, Karthik and Gajalakshmi, University Press
4. Introduction to Java programming, Y. Daniel Liang, Pearson Education.
5. Object Oriented Programming through Java, P. Radha Krishna, University Press.
6. Programming in Java, S. Malhotra, S. Chaudhary, 2nd edition, Oxford Univ. Press.
7. Java Programming and Object-oriented Application Development, R.A. ohnson, Cengage
8. Learning. Online Learning Resources: https://www.w3schools.com/java/java_oop.asp
<http://peterindia.net/JavaFiles.html>



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

Course Outcomes (COs):

After completion of the course, students will be able to

1. Solve real-world problems using OOP techniques.
2. Apply code re-usability through inheritance, packages and interfaces
3. Solve problems using java collection framework and I/O classes and
4. Develop applications by using parallel streams for better performance.
5. Develop applets for web applications and Build GUI and handle events generated by user interactions. Use the JDBC API to access the database.

Mapping of Course Outcomes with Program Outcomes:														
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	2	3	3	3	1	2	1	1	2	1	3	2	3
CO2	2	3	3	3	3	1	2	1	1	2	1	3	2	3
CO3	1	2	3	3	1	1	1	1	1	2	1	3	1	3
CO4	3	2	3	3	1	1	1	1	1	2	1	3	2	3
CO5	1	2	3	3	3	1	1	1	1	2	1	3	1	3

1- Slightly; 2 - Moderately; 3- Substantially



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

ELEMENTS OF QUANTUM COMPUTING**Credits: 2****Assessment: 100+0**

Semester	L-T-P	Course Code	Branches
IV	2-0-0	7-0-4-R25M2	CS

Prerequisites: Physics**Course Educational Objectives(CEOs):**

1. To introduce the basic principles of quantum mechanics relevant to computing.
2. To differentiate between classical and quantum computation models.
3. To understand Qubits, superposition, entanglement, and quantum gates.
4. To study basic quantum algorithms and their real-world applications.

UNIT-I:

Evolution of computation: classical to quantum paradigm, Limitations of classical computation, Basic principles of quantum mechanics relevant to computation.

Introduction to Quantum Computing: Bits Vs Qubits, Classical Vs Quantum logical operations

UNIT-II

Quantum states and qubits, Superposition principle and quantum measurement. Quantum entanglement and its importance.

Quantum gates: Pauli, Hadamard, Phase, CNOT, Toffoli. Bloch sphere representation of qubit states.

UNIT-III

Qubit: Physical implementations of Qubit, Qubit as a quantum unit of information. The Bloch sphere
Quantum Circuits: single qubit gates, multiple qubit gates, designing the quantum circuits. Bell states.

UNIT-IV

Quantum Algorithms: Classical computation on quantum computers. Relationship between quantum and classical complexity classes. Deutsch's algorithm, Deutsch's-Jozsa algorithm, Shor's factorization algorithm, Grover's search algorithm.

UNIT-V

Noise and error correction: Graph states and codes, Quantum error correction, fault-tolerant computation. Quantum Information and Cryptography: Comparison between classical and quantum information theory. Quantum Cryptography, Quantum teleportation



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

TEXTBOOKS:

1. Nielsen M.A., Quantum Computation and Quantum Information, Cambridge

REFERENCEBOOKS:

1. Quantum Computing for Computer Scientists by Noson S. Yanofsky and Mirco A. Mannucci.
2. Benenti G., Casati G. and Strini G., Principles of Quantum Computation and Information, Vol. I: Basic Concepts, Vol II
3. Basic Tools and Special Topics, World Scientific. Pittenger A. O., An Introduction to Quantum Computing Algorithms

Course Outcomes (COs):

1. Understand basic principles of quantum mechanics relevant to computation.
2. Understand basics of quantum states, qbit and gates.
3. Understand physical implementation of Qubit
4. Understand Quantum algorithms and their implementation
5. Understand The Impact of Quantum Computing on Cryptography

Mapping of Course Outcomes with Program Outcomes:														
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	2	3	3	2	-	-	-	1	-	1	1	3	3
CO2	3	2	3	3	2	-	-	-	1	1	1	2	3	3
CO3	3	3	3	3	3	-	-	-	-	-	1	3	3	3
CO4	3	3	3	3	3	-	-	-	-	1	1	3	3	3
CO5	3	3	3	3	3	-	-	-	1	-	1	3	3	3

1 – Slightly; 2 - Moderately; 3 – Substantially



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

MANAGERIAL ECONOMICS AND FINANCIAL ACCOUNTANCY**Credits: 3****Assessment: 40+ 60**

Semester	L-T-P	Course Code	Branches
IV	3-0-0	7-0-3-01R25	CS

COURSE DESCRIPTION

The objective of the Managerial Economics and Financial Accountancy is to equip students with the tools to make sound business and organisational decisions by analysing economic principles and financial data.

Course Educational Objectives(CEOs):**Managerial Economics objectives:**

1. Decision-making: To enable managers make optimal decisions to maximize a firm's profitability and success by integrating economic theory with business practices.
2. Analysis: To help in the analysis of business situations by providing insights into demand, supply, production functions and costs.
3. Problem-solving: To provide the tools to find optimal solutions for business problems such as those related to pricing, production, capital management, and profit planning.
4. Economic Concepts: To develop understanding of concepts like consumer behaviour, production and cost analysis and market structures.

Financial Accountancy objectives

5. Financial Analysis: To introduce students to financial accounting and analysis techniques, including the use of ratios like liquidity, activity, and profitability.
6. Historical Reporting: To understand how financial accounting records and reports on past performance through final accounts (income statement, balance sheet).
7. Financial Health Assessment: To provide a method for analysing and interpreting financial statements to understand a company's financial position and performance.
8. Informed Decisions: While primarily historical, the objective is to provide the information necessary for management and external stakeholders like investors to make informed decisions, even if it's for future forecasting.

Unit -I

Managerial Economics: Definition, Characteristics and Scope – Basic economic tools in Managerial Economics. Managerial Economics and its relation with other subjects – Demand and Supply: Concept and Analysis – Time value of money, Cash flow and Time Diagrams – Choosing among alternative investment proposals: Methods of Economic analysis (Pay Back, ARR, NPV, IRR and B/C ratio)



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

Unit -II

Depreciation and methods of calculating depreciation (Straight line, Sum of the years digit method, Declining balance method, Annuity method, Sinking fund method) – National income accounting, Methods of estimation, Various concepts of National Income, Significance of National income Estimation and limitations.

Unit -III

Types of Industrial Organization: Characteristics of Industrial organization, Features and evaluation of Sole Proprietorship, Partnership, Joint Stock Company, State/Public Enterprises and their types – The Business cycles and their relevance to decision making – Inflation: Definition, Process of inflation and measure of controlling inflation – New Economic Policy 1991 (Industrial Policy, Trade Policy, Fiscal Policy). It's Impact on Industry.

Unit -IV

Accounting Principles, procedure, Double entry system, Journal, Ledger, Trial balance, Cashbook – Preparation of Trading account and Profit and Loss account, Balance sheet.

Unit –V

Cost Accounting: Introduction, Classification of costs, Methods of costing, Techniques of costing – Cost sheet and preparation of cost sheet – Break-even Analysis: Meaning, its application and Limitation.

Text Books

1. Gupta, G.S. (2018), Managerial Economics (2e), McGraw Hill Education (India) Pvt. Ltd., Chennai.
2. Dewett. K.K. (2006), Modern Economic Theory, Sultan Chand and Co., New Delhi.
3. Agarwal A.N. (2019), Indian Economy, New Age International Pvt. Ltd., New Delhi.
4. Jain and Narang (2011), Accounting Part-I, Kalyani Publishers, New Delhi.
5. Arora, M.N. (2012), Cost Accounting: Principles and Practice (12e), Vikas Publication, New Delhi.

Reference Books

1. Henry Malcom Steiner (1996), Engineering Economics Principles (2e), McGraw Hill Education.
2. Leland Blank and Anthony Tarquin (2012), Engineering Economy (7e), McGraw Hill, New York.



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

Course Outcomes (COs):

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

- CO1** Understand the concepts of financial management and smart investment.
- CO2** Identify various cost elements of the product and its effect on decision making.
- CO3** Understand Economic and regulatory environment of the business and its impact on enterprises.
- CO4** Prepare the Accounting records and interpret the data for Managerial Decisions.
- CO5** Identify various cost elements of the product and its effect on financial decision making.

Mapping of Course outcomes with program outcomes:

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	2	1	2	1	2	1	1	2	1	3
CO2	3	3	2	2	1	2	2	1	1	2	1	3
CO3	3	2	1	1	1	1	1	1	1	2	1	3
CO4	3	2	2	1	1	1	1	1	1	2	1	3
CO5	3	3	3	2	2	1	1	1	1	2	1	3

1 – Slightly; 2 - Moderately; 3 – Substantially



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

STATISTICAL AND NUMERICAL METHODS LABORATORY**Credits: 1.5****Assessment: 40+ 60**

Semester	L-T-P	Course Code	Branches
IV	0-0-3	7-6-4-R25P1	CS

- At least 10 Experiments should be given covering the topics of the course Statistical and Numerical Methods (Using C / Python / MATLAB / R).
- Recommended Programming Language**
Best choice for these experiments → **Python**
Reason: Easy syntax, Built-in math libraries, Graph plotting support, Fast implementation.
- Evaluation Suggestion for Faculty**
Each lab record should include: Algorithm, Flowchart, Program, Output, Error analysis, Result comparison.

STATISTICAL METHODS EXPERIMENTS:**Experiment 1 — Data Collection & Classification (Unit I)****Objective:** Understand types of data and data organization.**Task:** Collect primary data from 20 students (height/marks/age). Obtain secondary data from a website or report. Classify both into grouped frequency tables. Compare primary vs secondary data characteristics.**Experiment 2 — Measures of Central Tendency (Unit I)****Objective:** Compute mean, median, and mode.**Task:** Given raw dataset, compute: Arithmetic Mean, Median, Mode, Repeat for grouped data. Compare results and interpret differences.**Experiment 3 — Measures of Variability, Skewness & Kurtosis (Unit I)****Objective:** Analyze dispersion and distribution shape.**Task:** Compute: Range, Variance, Standard deviation, Coefficient of variation, Calculate skewness and kurtosis. Interpret whether data is symmetric or skewed.

Chairperson
(BoS UG)

 DEAN
 FACULTY OF ENGINEERING
 S. V. U. COLLEGE OF ENGINEERING
 TIRUPATI-517 502

Experiment 4 — Probability Laws (Unit II)

Objective: Verify probability axioms and rules.

Task: Simulate coin tosses or dice throws.

Verify: Addition rule, Multiplication rule, Conditional probability, Solve numerical problems using Bayes' theorem.

Experiment 5 — Random Variables & Expectation (Unit II)

Objective: Understand discrete and continuous variables.

Task: Define a discrete random variable for dice experiment.

Compute: Probability distribution, Expected value, Variance, Plot probability distribution graph.

Experiment 6 — Probability Distributions (Unit III)

Objective: Apply standard distributions.

Task: Solve problems involving: Binomial distribution, Poisson distribution, Normal distribution. Verify properties (mean and variance). Demonstrate approximation of binomial to normal.

Experiment 7 — Chebyshev's Inequality Verification (Unit III)

Objective: Test theoretical inequality.

Task: Take dataset of 50 values. Compute mean and standard deviation. Verify Chebyshev's theorem for: $k = 2$, $k = 3$. Compare theoretical vs actual percentages.

Experiment 8 — Large Sample Hypothesis Testing (Unit IV)

Objective: Perform hypothesis tests.

Task: Conduct tests for: Single proportion, Difference of proportions, Single mean, Difference of means. Identify: Null hypothesis, Critical region, Decision.

Experiment 9 — Confidence Intervals (Unit IV)

Objective: Estimate population parameters.

Task: Calculate confidence intervals for: Mean (one sample), Difference of means (two sample), Proportion, Compare interval widths for different confidence levels.



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

Experiment 10 — Small Sample Tests & Quality Control (Unit V)

Objective: Perform statistical testing and process control.

Task: Conduct: t-test (single, two sample, paired), F-test for variances, χ^2 goodness of fit, χ^2 independence test,

Construct control charts: X-chart, R-chart, p-chart, Interpret whether process is in control.

NUMERICAL METHODS EXPERIMENTS :

Experiment 1 — Error Analysis

Aim: Study truncation and round-off errors.

Task: Write a program to compute value of a function (e.g., $\sin x$ or e^x) using series expansion. Compare approximate vs actual value.

Calculate: Absolute error, Relative error, Compare closed-form vs iterative computation.

Experiment 2 — Bisection Method

Aim: Find root of nonlinear equation.

Task: Input function $f(x)$ and interval $[a, b]$.

Implement bisection algorithm. Print iterations and root approximation. Verify convergence.

Experiment 3 — Regula-Falsi Method

Aim: Solve nonlinear equation using false position method.

Task: Implement algorithm. Compare number of iterations with bisection method. Display convergence table.

Experiment 4 — Newton–Raphson Method

Aim: Find roots using derivative-based iteration.

Task: Accept function and derivative. Perform iterations until tolerance reached. Compare speed with previous methods.



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

Experiment 5 — LU Decomposition**Aim:** Solve system of linear equations.**Task:** Input coefficient matrix A and vector B. Decompose $A = LU$. Solve using forward and backward substitution.**Experiment 6 — Iterative Methods & Convergence****Aim:** Solve linear system using iterative techniques.**Task:** Implement Jacobi or Gauss-Seidel method. Check convergence condition. Compare results with direct method.**Experiment 7 — Curve Fitting & Correlation****Aim:** Fit curve using least squares.**Task:** Fit: Linear curve, Quadratic curve, Exponential curve, Calculate correlation coefficient. Plot original vs fitted curve.**Experiment 8 — Interpolation Methods****Aim:** Estimate unknown values from known data.**Task:** Implement: Lagrange interpolation, Newton forward method, Divided difference method, Compare accuracy of each.**Experiment 9 — Numerical Differentiation & Integration****Aim:** Approximate derivative and integral numerically.**Task:** Differentiate tabulated function.

Apply: Trapezoidal rule, Simpson's 1/3 rule, Simpson's 3/8 rule, Compare with exact value.

Experiment 10 — ODE Solution Methods**Aim:** Solve differential equation numerically.**Task:** Solve $dy/dx = f(x,y)$ using: Euler method, Runge-Kutta method (RK4), Taylor series method, Compare accuracy of each.**Course Outcomes (COs):**

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

1. Analyze the data quantitatively or categorically using various statistical elementary tools.
2. Design mathematical models involving probability, random variables and the critical Thinking in the theory of probability and its applications in real life problems.


Chairperson
(BoS UG)

 DEAN
 FACULTY OF ENGINEERING
 S. V. U. COLLEGE OF ENGINEERING
 TIRUPATI-517 502

3. Apply the theoretical probability distributions like binomial, Poisson, and Normal in the Relevant application areas.
4. Analyze to test various hypotheses included in theory and types of errors for large samples.
5. Apply the different testing tools like t-test, F-test, chi-square test to analyze the relevant Real life problems

Numerical Methods

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

- 1) To apply numerical methods to solve the Solution of Linear and Non-Linear Algebraic Equations and curve fitting problems, error analysis.
- 2) Interpret the association of characteristics and through correlation and regression tools.
- 3) Derive interpolating polynomials using interpolation formulae by different methods.
- 4) Test and evaluate the accuracy of Differentiation and integrals numerically.
- 5) Using numerical methods we find solution of Ordinary differential equations.

Mapping of Course Outcomes with Program Outcomes:														
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	2	3	3	2	1	1	1	1	2	1	3	1	3
CO2	3	2	3	3	2	1	1	1	1	2	1	3	1	3
CO3	3	3	3	3	2	2	2	1	1	2	1	3	1	3
CO4	3	3	3	3	2	2	2	1	1	2	1	3	1	3
CO5	3	2	2	3	2	1	2	1	1	2	1	3	1	3

1-Slightly; 2-Moderately; 3-Substantially



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

ASSEMBLY LANGUAGE PROGRAMMING AND VHDL LABORATORY**Credits: 1.5****Assessment: 40+ 60**

Semester	L-T-P	Course Code	Branches
IV	0-0-3	7-6-4-R25P2	CS

At least 10 Experiments should be given covering the topics of the course Assembly Language Programming and VHDL

Assembly Language Programming**Experiment 1:** Study of Instruction Set and Addressing Modes

Objective: Write assembly programs demonstrating data transfer, arithmetic, logical instructions and various addressing modes.

Experiment 2: Arithmetic Operations

Objective: Develop assembly programs for addition, subtraction, multiplication and division of 8-bit and 16-bit numbers.

Experiment 3: Sorting of Numbers

Objective: Write an assembly program to sort a list of numbers in ascending and descending order.

Experiment 4: Searching Techniques

Objective: Implement linear search and binary search in assembly language.

Experiment 5: String Manipulation

Objective: Write assembly programs for string length calculation, string reversal and string comparison.

Experiment 6: Stack Operations and Subroutines

Objective: Implement stack operations and demonstrate procedure calls using CALL and RET instructions.



Chairperson
(BoS UG)


 DEAN
 FACULTY OF ENGINEERING
 S. V. U. COLLEGE OF ENGINEERING
 TIRUPATI-517 502

Experiment 7: Input/Output and Data Transfer

Objective: Develop assembly programs to interface with I/O devices and perform data transfer operations.

Experiment 8: Interrupt Handling

Objective: Write an assembly program to demonstrate software interrupts and basic interrupt service routines.

Experiment 9: Memory Organization and Block Transfer

Objective: Implement block data transfer between memory locations using assembly instructions.

Experiment 10: Basic Microprocessor Based Application

Objective: Develop a small application such as traffic light controller or simple calculator using assembly language.

VHDL LABORATORY

Experiment 1: Number Representation

Representation of signed and unsigned numbers, 1's complement and 2's complement representation.

Experiment 2: Code Conversion

Binary, BCD, Gray and Excess-3 code conversions.

Experiment 3: Boolean Algebra and Canonical Forms

Verification of DeMorgan's Theorem, SOP and POS forms, Canonical forms of Boolean expressions.

Experiment 4: Karnaugh Map Simplification

Minimization of Boolean expressions using K-Maps (up to 6 variables).

Unit II

Experiment 5: MSI Devices – Combinational Circuits

Implementation of Comparators, Multiplexers, Encoders, Decoders, Driver and Multiplexed Display.



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

Experiment 6: Arithmetic Circuits and ALU

Half Adder, Full Adder, Subtractor, Serial and Parallel Adders, BCD Adder, Barrel Shifter and ALU.

Unit III

Experiment 7: Flip-Flops and Counters

S-R, JK, Master-Slave JK and Edge Triggered Flip-Flops, Ripple and Synchronous Counters.

Experiment 8: Shift Registers and FSM Design

Shift Registers, Finite State Machine design using ASM charts, Pulse Train Generator and PRBS Generator.

Unit IV

Experiment 9: Logic Families and Memory Devices

TTL, CMOS, ECL characteristics, Propagation delay, fan-in, fan-out, noise margin, Memory elements and FPGA.

Unit V

Experiment 10: VHDL Modeling and Simulation

Data-flow, Behavioral and Structural modeling, Synthesis and simulation of combinational and sequential circuits.

Course outcomes(COs):

At the end of this course students will demonstrate the ability to

1. The design principles of digital computing systems
2. Design and analyze combinational logic circuits
3. Design and analyze synchronous sequential logic circuits
4. Design and implement complicated digital systems using Verilog
5. Design a VLSI circuit for an application



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

Mapping of Course Outcomes with Program Outcomes:														
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	2	1	3	3	1	2	1	1	2	1	2	2	3
CO2	1	3	2	1	1	1	2	1	1	2	1	3	1	3
CO3	1	2	3	2	1			1	1	2	1	2	2	2
CO4	3	2	2	3	1	1	1			2	1	3	1	2
CO5	3	3	3	3	2	1	1			2	1	3	2	3

1-Slightly; 2-Moderately; 3-Substantially



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

OBJECT ORIENTED PROGRAMMING THROUGH JAVA LABORATORY**Credits: 1.5****Assessment: 40+ 60**

Semester	L-T-P	Course Code	Branches
IV	0-0-3	7-6-4-R25P3	CS

At least 10 Experiments should be given covering the topics of the course Object Oriented Programming through Java.

Unit I

Experiment 1: Introduction to Java Program Structure - Basics of Java, data types, variables, operators and control statements.

Experiment 2: Classes, Objects and Constructors - Creation of classes and objects, constructors, this keyword and garbage collection.

Experiment 3: Method Overloading and Recursion - Implementation of method overloading, constructor overloading and recursion.

Experiment 4: Arrays and String Handling - One-dimensional and multi-dimensional arrays, String class and string handling methods.

Unit II

Experiment 5: Inheritance and Method Overriding - Single and multilevel inheritance, super keyword and dynamic method dispatch.

Experiment 6: Abstract Classes and final Keyword - Implementation of abstract classes and usage of final with inheritance.

Experiment 7: Packages and Access Protection - Creating packages, importing packages, CLASSPATH and access specifiers.

Experiment 8: Interfaces - Defining, implementing and extending interfaces, nested interfaces and variables in interfaces.

Unit III

Experiment 9: Exception Handling - Using try, catch, multiple catch, finally, throw and throws. Creating user-defined exceptions.

Experiment 10: File Handling and Stream I/O - Byte streams, character streams, File class, reading and writing files.



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

Experiment 11: Serialization, Enumerations and Generics - Object serialization, enum types, autoboxing and generics.

Experiment 12: Console and Random Access Files - Using Console class and performing random access file operations.

Unit IV

Experiment 13: Multithreading - Creating threads, thread priorities, synchronization and interthread communication.

Experiment 14: Collection Framework – List and Set - Implementation of ArrayList, LinkedList, HashSet and TreeSet.

Experiment 15: Collection Framework – Queue and Map - Using PriorityQueue, ArrayDeque, Hashtable and Properties.

Experiment 16: Utility Classes - Using Stack, Vector, StringTokenizer, BitSet, Date, Calendar, Random, Formatter and Scanner.

Unit V

Experiment 17: Applet Programming - Creating applet skeleton, repainting, passing parameters and using status window.

Experiment 18: Swing Components and Event Handling - Using JButton, JTextField, JLabel, ImageIcon and layout managers.

Experiment 19: Advanced Swing Components - Using JScrollPane, JList, JComboBox, JTable, JMenuBar and dialog boxes.

Experiment 20: JDBC Programming - JDBC architecture, types of drivers, connecting to database, creating table and performing basic operations.

Course Outcomes (COs):

After completion of the course, students will be able to

1. Solve real-world problems using OOP techniques.
2. Apply code re-usability through inheritance, packages and interfaces
3. Solve problems using java collection framework and I/O classes and
4. Develop applications by using parallel streams for better performance.
5. Develop applets for web applications and Build GUI and handle events generated by user interactions. Use the JDBC API to access the database.



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

Mapping of Course Outcomes with Program Outcomes:														
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	2	3	3	3	1	2	1	1	2	1	3	2	3
CO2	2	3	3	3	3	1	2	1	1	2	1	3	2	3
CO3	1	2	3	3	1	1	1	1	1	2	1	3	1	3
CO4	3	2	3	3	1	1	1	1	1	2	1	3	2	3
CO5	1	2	3	3	3	1	1	1	1	2	1	3	1	3

1-Slightly; 2-Moderately; 3-Substantially



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

BASIC WEB DESIGNING (Skill Oriented Course)**Credits: 2****Assessment: 100+0**

Semester	L-T-P	Course Code	Branches
IV	1-0-2	7-6-4-R25S1	CS

Prerequisites: NIL**Course Educational Objectives(CEOs):**

The objectives of this course is to acquire knowledge on the

1. Web related terminology and how does a website work.
2. Web standards and W3C elements
3. Responsive Web Designing
4. Client-side Scripting Languages (Front End)
5. Domains and Hosting

UNIT I**Introduction to Web and Web Design Principles:**

Brief History of Internet, What is World Wide Web, Why create a web site, Web Standards, Web pages, Website, Web browsers and Web servers and Web protocols. Basic principles involved in developing a web site, Planning process, Five Golden rules of web designing, Designing navigation bar, Page design, Home Page Layout, Design concept.

UNIT II**Introduction to HTML:**

What is HTML, HTML Documents, Basic structure of an HTML document, Creating an HTML document, Mark up Tags, Heading-Paragraphs, Line Breaks, HTML Tags. Introduction to elements of HTML, Working with Text, Working with Lists, Tables and Frames, Working with Hyperlinks, Images and Multimedia, Working with Forms and controls.

UNIT III**Introduction to Cascading Style Sheets:**

Concept of CSS, Creating Style Sheet, CSS Properties, CSS Styling (Background, Text Format, Controlling Fonts), Working with block elements and objects, Working with Lists and Tables, CSS Id and Class, Box Model (Introduction, Border properties, Padding Properties, Margin properties), CSS Advanced (Grouping, Dimension, Display, Positioning, Floating, Align, Pseudo class, Navigation Bar, Image Sprites, Attribute selector), CSS Color, Creating page Layout and Site Designs.



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

UNIT IV**Introduction to Java Script:**

What is Java Script? Basics of Java Script: Variables, functions, and Operators, select HTML elements with Java Script, Java Script Events and Event Handlers, Regular expressions and pattern matching in Java Script. Form validation using Java Script.

UNIT V**Introduction to Web Publishing or Hosting:**

Creating the Web Site, Saving the site, Working on the web site, Creating web site structure, Creating Titles for web pages, Themes-Publishing web sites. Case study: Web publishing and hosting using Heroku cloud platform (<https://www.heroku.com/>).

Text Books:

1. Kogent Learning Solutions Inc., HTML 5 in simple steps, Dreamtech Press.
2. A beginner's guide to HTML, NCSA, 14th May 2003.
3. Murray, Tom/Lynchburg, Creating a Web Page and Web Site, College, 2002.

Reference Books:

4. Web Designing and Architecture-Educational Technology Centre, University of Buffalo.
Steven M Schafer, HTML, XHTML, CSS and JavaScript, Wiley India.
5. Ian Pouncey, Richard York, Beginning CSS: Cascading Style Sheets for Web Design, Wiley India.
6. Kogent Learning, Web Technologies: HTML, JavaScript, Wiley India.

Course Outcomes(COs):

Having successfully completed this course the students will be able to:

1. Understand the web related terminology and how does a website work.
2. Describe and explain the relationship among HTML, XHTML, CSS, JavaScript, XML and other web technologies.
3. Create and publish advanced web pages with the help of HTML frames, scripting languages, and CSS.
4. Design forms for thick clients using JavaScript with interactive responsiveness and validations.
5. Design, host and publish websites in various domains.



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

Mapping of Course Outcomes with Program Outcomes:														
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	2	3	3	3	1	2	1	1	2	1	3	1	3
CO2	3	2	3	3	3	1	2	1	1	2	1	3	1	3
CO3	3	2	2	3	1	1	1	1	1	2	1	3	1	3
CO4	3	2	2	3	1	1	1	1	1	2	1	3	1	3
CO5	3	3	3	3	2	1	1	1	1	2	1	3	1	3

1 – Slightly; 2 - Moderately; 3 – Substantially



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

CONSTITUTION OF INDIA**Credits: 1.5****Assessment: 40+ 60**

Semester	L-T-P	Course Code	Branches
IV	0-0-3	7-0-4-R25M1	CS

Course Educational Objectives(CEOs):

Students will learn:

1. To Understand the premises informing the twin themes of liberty and freedom from a civil rights perspective.
2. To address the growth of Indian opinion regarding modern Indian intellectuals' constitutional role and entitlement to civil and economic rights as well as the emergence of nationhood in the early years of Indian nationalism.
3. To address the role of socialism in India after the commencement of the Bolshevik Revolution in 1917 and its impact on the initial drafting of the Indian Constitution.

Unit-I

History of Making of the Indian Constitution: History ,Drafting Committee, (Composition & Working) Philosophy of the Indian Constitution: Preamble Salient Features

Unit-II

Contours of Constitutional Rights & Duties:

- Fundamental Rights
- Right to Equality
- Right to Freedom
- Right against Exploitation
- Right to Freedom of Religion
- Cultural and Educational Rights
- Right to Constitutional Remedies
- Directive Principles of State Policy
- Fundamental Duties.

Unit-III

Organs of Governance:

- Parliament
- Composition
- Qualifications and Disqualifications
- Powers and Functions
- Executive
- President



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

- Governor
- Council of Ministers
- Judiciary, Appointment and Transfer of Judges, Qualifications
- Powers and Functions

Unit-IV

Local Administration:

- District's Administration head: Role and Importance,
- Municipalities: Introduction, Mayor and role of Elected Representative, CEO of Municipal Corporation.
- Pachayati raj: Introduction, PRI: Zila Pachayat.
- Elected officials and their roles, CEO Zila Pachayat: Position and role.
- Block level: Organizational Hierarchy (Different departments),
- Village level: Role of Elected and Appointed officials,
- Importance of grass root democracy

Unit-V

Election Commission:

- Election Commission: Role and Functioning.
- Chief Election Commissioner and Election Commissioners.
- State Election Commission: Role and Functioning.
- Institute and Bodies for the welfare of SC/ST/OBC and women.

TEXT BOOKS/REFERENCES:

1. The Constitution of India, 1950 (Bare Act), Government Publication.
2. Dr. S. N. Busi, Dr. B. R. Ambedkar framing of Indian Constitution, 1st Edition, 2015.
3. M. P. Jain, Indian Constitution Law, 7th Edn., Lexis Nexis, 2014.
4. D.D. Basu, Introduction to the Constitution of India, Lexis Nexis, 2015.

Course Outcomes(COs):

Students will be able to:

1. Discuss the growth of the demand for civil rights in India for the bulk of Indians before the arrival of Gandhi in Indian politics.
2. Discuss the intellectual origins of the framework of argument that informed the conceptualization of social reforms leading to revolution in India.
3. Discuss the circumstances surrounding the foundation of the Congress Socialist Party [CSP] under the leadership of Jawaharlal Nehru and the eventual failure of the proposal of direct elections through adult suffrage in the Indian Constitution.
4. Discuss the passage of the Hindu Code Bill of 1956.


Chairperson
(BoS UG)


DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

Mapping of Course Outcomes with Program Outcomes:

PO&PSO/ CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	2	2	-	-	2	1	2	-	-	2	1	1
CO2	2	1	1	-	1	1	-	-	-	1	1	1
CO3	2	2	-	-	2	1	2	-	-	2	1	1
CO4	-	-	-	-	2	1	2		-	2	1	1

1 - Slightly; 2 - Moderately; 3 – Substantially



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

B. Tech. (Computer Science and Engineering)
Semester V (Third Year)

Code		Category	Course Title	Hours/Week			Total	Credits	Marks		
Department	University			L	T	P			CIE	SEE	Total
CS6501C	7-6-5-01R25	CS-PC	Formal Languages and Automata Theory	3	0	0	3	3	40	60	100
CS6502C	7-6-5-02R25	CS-PC	Computer Networks	3	0	0	3	3	40	60	100
CS6503C	7-6-5-03R25	CS-PC	Operating Systems	3	0	0	3	3	40	60	100
CS6504C	7-6-5-04R25	CS-PC	Software Engineering	3	0	0	3	3	40	60	100
CS6505O	7-6-5-R25O1	CS-OE	Open Elective - I (MOOCS)	0	0	0	0	3	100	0	100
CS6506L	7-6-5-R25P1	CS-PC	Computer Networks Lab	0	0	3	3	1.5	40	60	100
CS6507L	7-6-5-R25P2	CS-PC	Operating Systems Lab	0	0	3	3	1.5	40	60	100
CS6508L	7-6-5-R25P3	CS-PC	Software Engineering Lab	0	0	3	3	1.5	40	60	100
CS6509S	7-6-5-R25S1	CS-SC	Advanced Python Programming	1	0	2	3	2	100	0	100
CS6510K	7-6-5-R25MK	CS-IK	IKS in CSE (IKS – Mandatory)	2	0	0	2	0	100	0	100
CS6511I	7-0-5-R25CS	CS-SI	Community Service Project (CSP)	0	0	0	0	2	100	0	100
TOTAL				15	0	11	26	23.5	680	420	1100
The students have to choose open electives (OE-I, OE-II, OE-III, etc..) from other disciplines or departments. The student cannot opt for an open elective course offered by his own (parent) department/discipline.											
Students shall ensure that IKS MOOKS and IKS Mandatory courses are not same either in title and/or in syllabi.											



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

FORMAL LANGUAGES AND AUTOMATA THEORY**Credits: 3****Assessment: 40+ 60**

Semester	L-T-P	Course Code	Branches
V	3-0-0	7-6-5-01R25	CS

Prerequisites: Discrete Mathematical Structures.**Course Educational Objectives(CEOs):**

The course is designed to

1. Identify different formal language classes and their relationships
2. Design grammars and recognizers for different formal languages
3. Understand the logical limits to computational capacity
4. Get proper insight on un-decidable problems

UNIT I

Why study Automata Theory, Central Concepts of Automata Theory, Informal Picture of Finite Automata, Deterministic Finite Automata, Nondeterministic Finite Automata and Applications, Finite Automata with Epsilon Transitions.

UNIT II

Regular Expressions and their Applications, Finite Automata and Regular Expressions, Algebraic Laws for Regular Expressions.

Properties of Regular Languages, Equivalence and Minimization of Automata.

UNIT III

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

Definition of Pushdown Automaton, The Languages of a PDA, Equivalence of PDA's and CFG's, Deterministic Pushdown Automata.

UNIT IV

Normal forms for CFG, Pumping Lemma for Context Free Languages, Closure and Decision Properties of CFLs

Turing Machine Model, Representation of Turing Machines, Language Acceptability by TM, Design of TMs, Universal Turing Machine, Halting Problem of TM, Church-Turing Thesis.

UNIT V

A Language that is not Recursively Enumerable, An Undecidable Problem that is Recursively Enumerable, Undecidable Problems about Turing Machines, The Classes of P and NP, NP Complete Problem.



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

Text Books:

1. John E. Hopcroft, Rajeev Motwani and Jeffrey D. Ullman, Introduction to Automata Theory, Languages, and Computation, Pearson Education Asia.
2. Martin J C, Introduction to Languages and the Theory of Computation, 3rd edition, Tata McGraw-Hill, 2003.

Reference Books:

1. Krithivasan K, Introduction to Formal Languages, Automata Theory and Computation, Pearson Education, 2009.
2. Rich E, Automata, Computability, and Complexity – Theory and Applications, Pearson Education, 2012.
3. Singh A, Elements of Computation Theory, Springer, 2009.
4. Cohen D I A, Introduction to Computer Theory, 2nd edition, John Wiley, 2000.
5. Lewis H, Papadimitriou C H, Elements of the Theory of Computation, 2nd edition, Prentice Hall, 1997.

Course Outcomes(COs):

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

1. Design finite automata to accept a set of strings of a language.
2. Determine whether the given language is regular or not.
3. Design context free grammars to generate strings of context free language.
4. Determine equivalence of languages accepted by pushdown automata and languages generated by context free grammars
5. Distinguish between computability & non-computability and decidability & undecidability.

Mapping of Course Outcomes with Program Outcomes:														
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	2	3	2	3	-	-	-	-	-	1	3	3	3
CO2	3	3	2	3	3	-	-	-	-	-	2	2	2	3
CO3	3	2	3	2	3	-	-	-	1	-	2	2	2	3
CO4	3	2	2	2	3	-	-	-	1	1	1	3	3	3
CO5	3	3	3	3	2	-	-	-	1	1	1	3	3	3

1 – Slightly; 2 - Moderately; 3 – Substantially



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

COMPUTER NETWORKS**Credits: 3****Assessment: 40+ 60**

Semester	L-T-P	Course Code	Branches
V	3-0-0	7-6-5-02R25	CS

Prerequisites : Operating systems, Data structures.**Course Educational Objectives(CEOs):**

The course is designed to

1. Provide insight about fundamental concepts, basic taxonomy and terminology of Computer Networks.
2. Gain comprehensive knowledge about the principles, protocols, and significance of Layers in OSI and TCP/IP

UNIT I

Introduction to Computer Networks: Networks, Component and Categories, Topologies, Transmission Media, Reference Models: ISO/OSI Model and TCP/IP Model.

Physical Layer: Analog and Digital Signals, Periodic Analog Signals, Transmission Impairments, Data rate limits, Performance, Digital data transmission techniques, Analog data transmission techniques, Multiplexing: FDM, WDM and TDM; Spread Spectrum, Switching: Circuit and Packet.

UNIT II

Data Link Layer and Medium Access Sub Layer: Design Issues, Error Detection and Error Correction, Elementary Data Link Protocols, Sliding Window Protocols, Channel allocation problems, Multiple Access Protocols: Pure ALOHA, Slotted ALOHA, CSMA/CD, CSMA/CA; IEEE 802.3 Ethernet.

UNIT III

The Network Layer: Network layer design issues, Routing Algorithms, Congestion Control Algorithms, Quality of Service, Internetworking, The Network layer in the Internet: IPV4 Addresses, IPV6, Internet Control protocol, OSPF, BGP, IP, ICMPv4, IGMP.

UNIT IV

The Transport Layer: The Transport Service, Elements of Transport Protocols, Congestion Control, Internet Transport Protocols: UDP, TCP; Network Performance Measurement.

UNIT V

The Application Layer: Introduction, Client-Server Programming, Domain Name System (DNS), WWW and HTTP, FTP, E-mail, TELNET, Secure Shell, SNMP, IP Security Architecture, Firewalls.


Chairperson
(BoS UG)

DEAN
 FACULTY OF ENGINEERING
 S. V. U. COLLEGE OF ENGINEERING
 TIRUPATI-517 502

Text Books:

1. Computer Networks, Andrew S. Tanenbaum, Wetherall, Pearson, 5th edition, 2010.
2. Data communications and Networking, Behrouz A. Forouzan, Mc Graw Hill Education, 5th edition, 2012.

Reference Books:

3. Kurose J F, Ross K W, Computer Networking – A Top-Down Approach, 5th edition, Pearson Education, 2010.
4. Peterson L L, Davie B S, Computer Networks - A Systems Approach, 5th edition, Morgan Kaufmann, 2011.
5. Forouzan B A, Mosharraf F, Computer Networks – A Top-Down Approach, Tata McGraw-Hill, 2012.
6. Olifer N, Olifer V, Computer Networks – Principles, Technologies, and Protocols for Network Design, Wiley, 2006.

Course Outcomes(COs):

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

1. Understand and explain the fundamentals of computer networks, including network components, categories, and their applications.
2. Explain the functions and design issues of the Data Link Layer and the role of the Medium Access Sub-Layer in reliable data communication.
3. Understand and explain the design issues of the Network Layer, including service models, packet switching, and internetworking principles.
4. Describe the elements of transport protocols, including addressing, connection establishment and release, flow control, and error control.
5. Understand and apply client–server programming concepts, including request–response communication models used in networked applications.

Mapping of Course Outcomes with Program Outcomes:														
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	2	3		3	-	-	-	-	2	-	-	1	3
CO2	3	2	3	3	3	-	-	-	-	2	-	3	1	3
CO3	3	2	2	3	1	-	-	-	-	2	-	3	1	3
CO4	3	2	2	3	1	-	-	-	-	2	-	3	1	3
CO5	3	3	3	3	2	2	-	3	1	2	1	3	1	3

1 – Slightly; 2 - Moderately; 3 – Substantially



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

OPERATING SYSTEMS**Credits: 3****Assessment: 40+ 60**

Semester	L-T-P	Course Code	Branches
V	3-0-0	7-6-5-03R25	CS

Prerequisites : Data structures, Computer Organization, Digital Electronics and Logic Design.

Course Educational Objectives(CEOs):

The course is designed to provide knowledge about the services rendered by operating systems

1. present detail discussion on processes, threads and scheduling algorithms.
2. discuss various file-system design and implementation issues
3. provide good insight on various memory management techniques
4. expose the students with different techniques of handling of deadlocks
5. familiarize students the basics of Linux system and perform administrative tasks on Linux servers
6. discuss how the protection domains help to achieve security in a system

UNIT I

Operating Systems Overview: Operating system functions, Operating system structure, Operating systems operations, protection and security, Computing environments, Open-Source Operating Systems

System Structures: Operating System Services, User and Operating-System Interface, systems calls, Types of System Calls, system programs, operating system structure, operating system debugging, System Boot.

UNIT II

Process Concept: Process scheduling, Operations on processes, Interprocess communication, Communication in client server systems.

Multithreaded Programming: Multithreading models, Thread libraries, Threading issues, Examples.

Process Scheduling: Basic concepts, Scheduling criteria, Scheduling algorithms, Multiple processor scheduling, Thread scheduling, Examples.

Inter-process Communication: Race conditions, Critical Regions, Mutual exclusion with busy waiting, Sleep and wakeup, Semaphores, Mutexes, Monitors, Message passing, Barriers, Classical IPC Problems - Dining philosophers problem, Readers and writers problem.



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

UNIT III

Memory-Management Strategies: Introduction, Swapping, Contiguous memory allocation, Paging, Segmentation, Examples.

Virtual Memory Management: Introduction, Demand paging, Copy on-write, Page replacement, Frame allocation, Thrashing, Memory-mapped files, Kernel memory allocation, Examples.

UNIT IV

Deadlocks: Resources, Conditions for resource deadlocks, Ostrich algorithm, Deadlock detection and recovery, Deadlock avoidance, Deadlock prevention.

File Systems: Files, Directories, File system implementation, management and optimization.

Secondary-Storage Structure: Overview of disk structure, and attachment, Disk scheduling, RAID structure, Stable storage implementation.

UNIT V

System Protection: Goals of protection, Principles and domain of protection, Access matrix, Access control, Revocation of access rights.

System Security: Introduction, Program threats, System and network threats, Cryptography for security, User authentication, Implementing security defenses, Fire-walling to protect systems and networks, Computer security classification.

Case Studies: Linux, Microsoft Windows 7.

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.)

Reference Books:

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



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

Course Outcomes(COs):

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

1. Understand how operating system manages resources such as processors, memory and I/O.
 2. Demonstrate knowledge and understanding of how concurrency in OS is handled.
 3. Understand the techniques used to implement the process manager
 4. Implement various memory management and demand paging techniques.
5. Understand various schemes available for achieving system protection and system security

Mapping of Course Outcomes with Program Outcomes:														
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	3	2	2	3	-	-	-	2	-	-	2	3	3
CO2	3	2	3	3	3	-	-	2	2	-	-	2	3	3
CO3	2	3	2	3	2	-	-	2	2	2	2	2	3	3
CO4	3	2	2	3	3	-	-	2	-	2	2	2	3	3
CO5	3	3	3	3	2	-	-	-	-	2	1	2	3	3

1 – Slightly; 2 - Moderately; 3 – Substantially



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

SOFTWARE ENGINEERING**Credits: 3****Assessment: 40+ 60**

Semester	L-T-P	Course Code	Branches
V	3-0-0	7-6-5-04R25	CS

Prerequisites : OOPS, Data Structures, Operating Systems.**Course Educational Objectives(CEOs):**

The course is designed to understand

1. Software life cycle models.
2. Software requirements and SRS document.
3. Different software design strategies
4. Quality control and how to ensure good quality software.
5. Planning and estimation of software projects.
6. Maintenance of software and gain knowledge of the overall project activities.

UNIT I**Introduction to Software Engineering:** The Nature of Software, The Unique Nature of WebApps, Software Engineering, Software Process, Software Engineering Practice, Software Myths.**Process Models:** A Generic Process Model, Process Assessment and Improvement, Prescriptive Process Models, Specialized Process Models, The Unified Process, Personal and Team Process Models, Process Technology, Product and Process.**Agile Development:** Agility, Agility and the Cost of Change, Extreme Programming, Agile Process Models**UNIT II****Understanding Requirements:** Requirements Engineering, Eliciting Requirements, Developing Use Cases, Building the Requirements Model, Negotiating Requirements, Validating Requirements.**Requirements Modeling:** Requirements Analysis, Scenario based Modeling, Class based Modeling, Requirements Modeling Strategies, Flow Oriented Modeling, Patterns for Requirement Modeling, Requirements Modeling for WebApps**Design Concepts:** Design Process, Design Concepts, Design Model.**Architectural Design:** Software Architecture, Architectural Genres, Architectural Styles, Architectural Design, Alternative Architectural Designs, Architectural Mapping using Data flow.**UNIT III****Component Level Design:** Component, Class based Components, Conducting Component level design, Component level Design for WebApps, Designing Traditional Components, Component based Development.**User Interface Design:** The Golden Rules, User Interface Analysis and Design, Interface Analysis, Interface Design Steps, WebApp Interface Design, Design Evaluation.

Chairperson
(BoS UG)


DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

Pattern Based Design: Design Patterns, Pattern based Software Design, Architectural Patterns, Component Level Design Patterns, User Interface Design Patterns.

UNIT IV

Software Quality Concepts: Software Quality, Software Quality Dilemma, Achieving Software Quality.

Software Quality Assurance: Elements of Software Quality Assurance, SQA Goals and Metrics, Formal Approaches to SQA, Statistical SQA, Software Reliability.

Software Testing Strategies: A Strategic Approach to Software Testing, Strategic Issues, Unit Testing and Integration Testing (both Conventional and OO Software), Test Strategies for WebApps, Validation Testing, System Testing, Art of Debugging.

Testing Conventional Applications: Software Testing Fundamentals, Internal and External View of Testing, White-Box Testing, Basis Path Testing, Control Structure Testing, Black-Box Testing, Model based Testing, Testing for Specialized Environments, Patterns for Software Testing.

Computer Aided Software Engineering: CASE and its Scope, CASE Environment, CASE Support in Software Life Cycle, Characteristics of CASE Tools, Towards Second Generation CASE Tool.

UNIT V

Managing Software Projects: Project Management Concepts, Metrics in the Process and Project Domains, Software Measurement, Metrics for Software Quality, Project Planning Process, Software Scope and Feasibility, Software Project Estimation, Decomposition Techniques, Empirical Estimation Models, Estimation for OO Projects, Project Scheduling – Basic Principles, Defining a Task Set and Task Network, Scheduling, Introduction to Risk Management, Software Maintenance, Software Supportability, Software Reengineering, Reverse Engineering, Restructuring, Forward Engineering, Economics of Reengineering.

Text Books:

1. Pressman R S, Software Engineering: A Practitioner's Approach, 7th edition, McGraw-Hill, 2010.
2. Sommerville I, Software Engineering, 9th edition, Pearson Education, 2011.

Reference Books:

1. Jalote P, Software Engineering: A Precise Approach, Wiley, 2010.
2. Braude E J, Bernstein M E, Software Engineering: Modern Approaches, 2nd edition, Wiley, 2010.
3. Saleh K A, Software Engineering, J Ross Publishing, 2009.
4. Bruegge B, Dutoit A H, Object-Oriented Software Engineering Using UML, Patterns, and Java, 3rd edition, Prentice Hall, 2009.



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

5. Bennett S, McRobb S, Farmer R, Object-Oriented System Analysis and Design Using UML, 4th edition, McGraw-Hill, 2010.
6. Lethbridge T C, Laganieri R, Object-Oriented Software Engineering, 2nd edition, McGraw-Hill, 2005.

Course Outcomes(COs):

By the end of this course students will be able to

1. Define and develop a software project by applying appropriate software life cycle models and requirement engineering practices.
2. Apply software engineering principles, processes, and agile practices to develop quality software systems.
3. Model and design software systems using requirements modeling, architectural design, component design, and UI design techniques.
4. Apply software quality assurance, testing strategies, and CASE tools to ensure reliability and quality of software systems.
5. Analyze project planning, estimation, risk, maintenance, and reengineering issues involved in software project management.

Mapping of Course Outcomes with Program Outcomes:														
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	3	3	–	–	–	–	–	1	1	–	2	3	2
CO2	3	2	3	–	2	–	–	1	1	–	–	2	3	2
CO3	3	3	3	2	2	–	–	–	1	1	–	2	3	3
CO4	3	3	2	2	2	–	–	–	–	–	–	2	2	2
CO5	3	3	2	2	2	–	–	–	1	–	3	3	3	3

1 – Slightly; 2 - Moderately; 3 – Substantially



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

COMPUTER NETWORKS LABORATORY**Credits: 1.5****Assessment: 40+ 60**

Semester	L-T-P	Course Code	Branches
V	0-0-3	7-6-5-R25P1	CS

At least 10 Experiments should be given covering the topics of the course, Computer Networks.

Computer Networks Laboratory Experiments (Using C / Java / Python)**Unit I**

Experiment 1: Program to simulate different network topologies (Bus, Star, Ring) using Python.

Experiment 2: Program to calculate data rate using Nyquist and Shannon theorem using C.

Experiment 3: Program to demonstrate analog and digital signal representation using Python.

Experiment 4: Program to simulate TDM multiplexing using C.

Experiment 5: Program to simulate circuit switching and packet switching using Java.

Unit II

Experiment 6: Program to implement CRC error detection using C.

Experiment 7: Program to implement Hamming Code error correction using Python.

Experiment 8: Program to simulate Stop-and-Wait protocol using C.

Experiment 9: Program to simulate Sliding Window protocol using Java.

Experiment 10: Program to simulate CSMA/CD protocol using Python.

Unit III

Experiment 11: Program to implement Dijkstra's shortest path routing algorithm using C.

Experiment 12: Program to implement Distance Vector Routing algorithm using Python.

Experiment 13: Program to validate IPv4 address and perform subnet calculation using Java.

Experiment 14: Program to simulate simple congestion control mechanism using Python.

Unit IV

Experiment 15: Program to implement UDP client-server communication using C sockets.

Experiment 16: Program to implement TCP client-server communication using Java sockets.

Experiment 17: Program to measure network performance (RTT calculation) using Python.



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

Unit V

Experiment 18: Program to perform DNS lookup using Python.

Experiment 19: Program to implement HTTP client to fetch web page using Java.

Experiment 20: Program to demonstrate secure communication using SSL sockets in Python.

Course Outcomes(COs):

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

1. Understand and explain the fundamentals of computer networks, including network components, categories, and their applications.
2. Explain the **functions and design issues of the Data Link Layer** and the role of the **Medium Access Sub-Layer** in reliable data communication.
3. Understand and explain the design issues of the Network Layer, including service models, packet switching, and internetworking principles.
4. **Describe the elements of transport protocols**, including addressing, connection establishment and release, flow control, and error control.
5. **Understand and apply client-server programming concepts**, including request-response communication models used in networked applications.

Mapping of Course Outcomes with Program Outcomes:														
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	3	3	–	–	–	–	–	1	1	–	2	3	2
CO2	3	2	3	–	2	–	–	1	1	–	–	2	3	2
CO3	3	3	3	2	2	–	–	–	1	1	–	2	3	3
CO4	3	3	2	2	2	–	–	–	–	–	–	2	2	2
CO5	3	3	2	2	2	–	–	–	1	–	3	3	3	3

1 – Slightly; 2 - Moderately; 3 – Substantially



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

OPERATING SYSTEMS LABORATORY**Credits: 1.5****Assessment: 40+ 60**

Semester	L-T-P	Course Code	Branches
V	0-0-3	7-6-5-R25P2	CS

At least 10 Experiments should be given covering the topics of the course, Operating Systems.

Unit I

Experiment 1: Write a C program to demonstrate different types of system calls in Linux.

Experiment 2: Write a C program to implement process creation using fork() and exec() system calls.

Experiment 3: Write a Python program to simulate Operating System services and user interface.

Experiment 4: Write a Java program to demonstrate client-server communication using sockets.

Unit II

Experiment 5: Write a C program to implement CPU scheduling algorithms (FCFS and SJF).

Experiment 6: Write a C program to implement Priority Scheduling and Round Robin Scheduling.

Experiment 7: Write a C program to demonstrate Interprocess Communication using pipes.

Experiment 8: Write a C program to solve Producer-Consumer problem using Semaphores.

Experiment 9: Write a C program to solve Dining Philosophers problem.

Experiment 10: Write a Java program to create multiple threads and demonstrate synchronization.

Unit III

Experiment 11: Write a C program to implement Paging memory management technique.

Experiment 12: Write a C program to implement Page Replacement algorithms (FIFO, LRU).

Experiment 13: Write a Python program to simulate Demand Paging.

Experiment 14: Write a C program to simulate Memory Allocation strategies (First Fit, Best Fit, Worst Fit).

Unit IV

Experiment 15: Write a C program to implement Deadlock Detection algorithm.

Experiment 16: Write a C program to implement Banker's Algorithm for Deadlock Avoidance.

Experiment 17: Write a Python program to simulate Disk Scheduling algorithms (FCFS, SSTF).

Experiment 18: Write a C program to implement File handling system calls in Linux.



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

Unit V

Experiment 19: Write a Java program to implement Access Control Matrix model.

Experiment 20: Write a Python program to demonstrate basic cryptography techniques for system security.

Course Outcomes(COs):

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

1. Understand how operating system manages resources such as processors, memory and I/O.
2. Demonstrate knowledge and understanding of how concurrency in OS is handled.
3. Understand the techniques used to implement the process manager
4. Implement various memory management and demand paging techniques.
5. Understand various schemes available for achieving system protection and system security

Mapping of Course Outcomes with Program Outcomes:														
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	1	2	2	3	1	2	1	1		1	3	1	3
CO2	1	2	3	3	3	1	2	1	1		1	3	2	3
CO3	2	1	2	3	1	1	1			2	1	3	1	3
CO4	3	2	2	3	1	1	1			2	1	3	2	3
CO5	3	3	3	3	2	1	1			2	1	3	1	3

1 – Slightly; 2 - Moderately; 3 – Substantially



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

SOFTWARE ENGINEERING LABORATORY**Credits: 1.5****Assessment: 40+ 60**

Semester	L-T-P	Course Code	Branches
V	0-0-3	7-6-5-R25P3	CS

At least 10 Experiments should be given covering the topics of the course, Software Engineering.

Experiment 1: Develop a program to illustrate different Software Process Models (Waterfall, Incremental, Agile) through case study simulation.

Experiment 2: Develop a mini project demonstrating Agile methodology (Sprint planning and iteration).

Experiment 3: Develop a simple tool to track software myths and realities in project scenarios.

Experiment 4: Create a program to simulate Extreme Programming (XP) practices such as pair programming workflow.

Experiment 5: Develop a program to create and manage Software Requirement Specification (SRS) document.

Experiment 6: Develop a Use Case modeling application for a sample system.

Experiment 7: Implement a program to demonstrate scenario-based and class-based requirement modeling.

Experiment 8: Design and implement a simple architectural design for a web-based system.

Experiment 9: Develop a prototype illustrating different architectural styles (Layered, Client-Server).

Experiment 10: Develop a component-based application demonstrating modular design principles.

Experiment 11: Create a program implementing reusable class-based components.

Experiment 12: Develop a simple User Interface following UI golden rules.

Experiment 13: Design and implement design patterns (Singleton, Factory, MVC) in a sample application.

Experiment 14: Develop a WebApp interface prototype applying UI design patterns.

Experiment 15: Develop a program to calculate and analyze Software Quality Metrics.



Chairperson
(BoS UG)


 DEAN
 FACULTY OF ENGINEERING
 S. V. U. COLLEGE OF ENGINEERING
 TIRUPATI-517 502

Experiment 16: Implement Unit Testing and Integration Testing for a sample application.

Experiment 17: Develop a program to demonstrate White-box and Black-box testing techniques.

Experiment 18: Create a simple CASE tool prototype to support documentation and design.

Experiment 19: Develop a project estimation tool using empirical estimation models.

Experiment 20: Develop a project scheduling and risk management simulation tool.

Course Outcomes(COs):

1. Apply software life cycle models and requirement engineering techniques to develop Software Requirement Specification (SRS) documents.
2. Design software systems using UML diagrams such as use case, class, sequence, activity, and architecture diagrams.
3. Implement software projects by applying appropriate design patterns, coding standards, and version control practices.
4. Perform software testing using unit, integration, system, and validation testing techniques to ensure software quality.
5. CO5: Use software engineering tools and CASE tools to manage software projects, documentation, and maintenance activities.

Mapping of Course Outcomes with Program Outcomes:														
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	3	2	–	–	–	–	–	1	1	–	2	3	2
CO2	3	3	3	2	2	–	–	–	1	–	–	2	3	3
CO3	3	2	3	–	3	–	–	–	2	1	–	2	3	3
CO4	3	3	2	2	2	–	–	–	–	–	–	2	2	2
CO5	2	2	2	2	3	–	–	–	1	–	2	3	3	3

1 – Slightly; 2 - Moderately; 3 – Substantially


Chairperson
(BoS UG)


DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

ADVANCED PYTHON PROGRAMMING**Credits: 2****Assessment: 100+0**

Semester	L-T-P	Course Code	Branches
V	1-0-2	7-6-5-R25S1	CS

Prerequisites : Basic Python programming.**Course Educational Objectives(CEOs):**

The course is designed to

1. Familiarize the basics of Python Third Party Tools and usages.
2. Understand the advantage of using Python libraries for implementing Machine Learning models.
3. Understand the Python Data Structures for Full Stack Development.

The course shall cover the following topics:

- Parallel System Tools: Forking Processes, Threads, Program Exits, Interprocess Communication, The multiprocessing Module, Other Ways to Start Programs, A Portable Program-Launch Framework, Other System Tools Coverage.
- GUI Design with Tkinter: Menus, Listboxes and Scrollbars, Text, Canvas, Grids, Time Tools, Threads, and Animation, Other Widgets and Options.
- GUI Coding Techniques: GuiMixin: Common Tool Mixin Classes, GuiMaker: Automating Menus and Toolbars, ShellGui: GUIs for Command-Line Tools, GuiStreams: Redirecting Streams to Widgets, Reloading Callback Handlers Dynamically, Wrapping Up Top-Level Window Interfaces, GUIs, Threads, and Queues, More Ways to Add GUIs to Non-GUI Code, The PyDemos and PyGadgets Launchers.
- Complete GUI Programs: PyEdit: A Text Editor Program/Object, PyPhoto: An Image Viewer and Resizer, PyView: An Image and Notes Slideshow, PyDraw: Painting and Moving Graphics, PyClock: An Analog/Digital Clock Widget, PyToe: A Tic-Tac-Toe Game Widget.
- Network Scripting: Python Internet Development Options, Plumbing the Internet, Socket Programming, Handling Multiple Clients, Making Sockets Look Like Files and Streams, A Simple Python File Server.



Chairperson
(BoS UG)


 DEAN
 FACULTY OF ENGINEERING
 S. V. U. COLLEGE OF ENGINEERING
 TIRUPATI-517 502

- Client-Side Scripting: FTP: Transferring Files over the Net, Transferring Files with ftplib, Transferring Directories with ftplib, Transferring Directory Trees with ftplib, Processing Internet Email, POP: Fetching Email, SMTP: Sending Email, email: Parsing and Composing Mail Content, A Console-Based Email Client, The mailtools Utility Package, NNTP: Accessing Newsgroups, HTTP: Accessing Websites, The urllib Package Revisited, Other Client-Side Scripting Options.
- The PyMailGUI Client: Major PyMailGUI Changes, A PyMailGUI Demo, PyMailGUI Implementation, Ideas for Improvement.
- Server-Side Scripting: What's a Server-Side CGI Script?, Running Server-Side Examples, Climbing the CGI Learning Curve, Saving State Information in CGI Scripts, The Hello World Selector, Refactoring Code for Maintainability, More on HTML and URL Escapes, Transferring Files to Clients and Servers.
- The PyMailCGI Server: The PyMailCGI Website, The Root Page, Sending Mail by SMTP, Reading POP Email, Processing Fetched Mail, Utility Modules, Web Scripting Trade-Offs.
- Databases and Persistence: Persistence Options in Python, DBM Files, Pickled Objects, Shelve Files, The ZODB Object-Oriented Database, SQL Database Interfaces, ORMs: Object Relational Mappers, PyForm: A Persistent Object Viewer .
- Data Structures: Implementing Stacks, Implementing Sets, Subclassing Built-in Types, Binary Search Trees, Graph Searching, Permuting Sequences, Reversing and Sorting Sequences, PyTree: A Generic Tree Object Viewer.
- Text and Language: Strategies for Processing Text in Python, String Method Utilities, Regular Expression Pattern Matching, XML and HTML Parsing, Advanced Language Tools, Custom Language Parsers, PyCalc: A Calculator Program/Object.
- Python Data Structures for Data Science: Numpy, Pandas, Scipy, Matplotlib, Seaborn
- Python Data Structures for Full Stack Development, Python for Natural Language Processing
- Python for Machine Learning: Working with Beautiful Soup, Scikit-Learn, NLP with Python, Text mining with python.
- Python for Deep Learning: Working with Tensor flow, Keras and PyTorch
- Python for Cryptography and Network Security: Stanford-Corenlp, Bcrypt



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

Text Books:

1. Mark Lutz, Programming Python, O`Reilly, 4th Edition, 2010
2. Michael Urban and Joel Murach, Python Programming, Shroff/Murach, 2016

Reference Books:

1. Mark Lutz, Learning Python, O`Reilly, Fifth Edition, 2013.
2. Dattaraj Rao, Keras to Kubernetes: The Journey of a Machine Learning Model to Production, Wiley, 2019.
3. Wes McKinney, Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython, O`Reilly, Second Edition, 2017.
4. Aurélien Géron, Hands-On Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems, O`Reilly, First Edition, 2017.
5. Andreas C. Müller and Sarah Guido, Introduction to Machine Learning with Python: A Guide for Data Scientists, O`Reilly, First Edition, 2016.
6. Jake VanderPlas, Python Data Science Handbook: Essential Tools for Working with Data, O`Reilly, First Edition, 2016.
7. Delip Rao and Brian McMahan, Natural Language Processing with PyTorch: Build Intelligent Language Applications Using Deep Learning, O`Reilly, First Edition, 2019.
8. Miguel Grinberg, Flask Web Development: Developing Web Applications with Python, O`Reilly, Second Edition, 2018.
9. Joel Grus, Data Science from Scratch: First Principles with Python, O`Reilly, First Edition, 2015.
10. J. Burton Browning, Marty AlchinPro Python 3: Features and Tools for Professional Development, Apress, Third Edition, 2019.
11. Abhishek Nandy, Manisha Biswas, Reinforcement Learning: With Open AI, TensorFlow and Keras Using Python, Apress, 2018.
12. Navin Kumar Manaswi, Deep Learning with Applications Using Python: Chatbots and Face, Object, and Speech Recognition With TensorFlow and Keras, Apress, 2018.
13. Akshay Kulkarni, Adarsha Shivananda, Natural Language Processing Recipes: Unlocking Text Data with Machine Learning and Deep Learning using Python, Apress, 2019.
14. Sayan Mukhopadhyay, Advanced Data Analytics Using Python: With Machine Learning, Deep Learning and NLP Examples, Apress, 2018.
15. Jojo Moolayil, Learn Keras for Deep Neural Networks: A Fast-Track Approach to Modern Deep Learning with Python, Apress, 2019.
16. Manohar Swamynathan, Mastering Machine Learning with Python in Six Steps: A Practical Implementation Guide to Predictive Data Analytics Using Python, Apress, 2017.



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

21. Daniel Rubio, Beginning Django Web Application Development and Deployment with Python, Apress, 2017.
22. Fabio Nelli, Python Data Analytics: Data Analysis and Science Using Pandas, matplotlib, and the Python Programming Language, Apress, 2015.
23. Fabio Nelli, Python Data Analytics: With Pandas, NumPy, and Matplotlib, Apress, Second Edition, 2018.

Course Outcomes (COs):

After successfully completion o this course, the students will be able to:

1. Apply Python third-party tools and libraries in real-time programming environments.
2. Apply Python for cryptography and network security applications.
3. Implement Python for full-stack development and client–server applications.
4. Apply Python libraries for implementing Machine Learning and Deep Learning models.
5. Apply Python data structures and libraries for data science and analytical applications.

Mapping of Course Outcomes with Program Outcomes:														
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	2	2	–	3	–	–	–	1	1	–	2	3	2
CO2	3	3	2	–	2	2	–	1	–	–	–	2	2	2
CO3	3	2	3	–	3	–	–	–	1	2	1	2	3	3
CO4	3	3	3	2	3	–	–	–	–	–	–	3	3	3
CO5	3	3	2	2	3	–	–	–	–	–	–	3	3	3

1 – Slightly; 2 - Moderately; 3 – Substantially



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

IKS IN CSE (IKS-Mandatory)**Credits: NIL****Assessment: 100+0**

Semester	L-T-P	Course Code	Branches
V	1-0-2	7-6-5-R25MK	CS

Prerequisites : NIL**Course Educational Objectives(CEOs):**

1. To introduce students to the foundations of the Indian Knowledge System (IKS) and its philosophical underpinnings.
2. To explore the mathematical, linguistic, and logical traditions of ancient India relevant to computer science.
3. To highlight Indian contributions to algorithms, computation, and scientific reasoning.
4. To relate Indian epistemology, logic, and linguistics to modern computing paradigms such as AI, data structures, and formal languages.
5. To encourage ethical, sustainable, and human-centric approaches to technology inspired by IKS.

UNIT I: Introduction to Indian Knowledge Systems (IKS)

Meaning, scope, and sources of IKS. Overview of knowledge traditions. Branches of traditional knowledge. Relevance of IKS to modern education and science.

UNIT II: Indian Mathematics and Computation

Indian contributions to number systems, zero, infinity, and place value. Vedic mathematics and algorithmic problem-solving. Ancient algorithms for arithmetic, geometry, and astronomy (Sulba Sutras, Aryabhata, Bhaskaracharya). Binary ideas in Indian prosody (Chandas) and connection to modern digital systems.

Comparative study: Indian algorithms vs. modern computational methods.

UNIT III: Indian Logic and Knowledge Representation

principles of inference, reasoning, and knowledge validation. Logic and ontology, Parallels to propositional logic, predicate logic, and ontology in computer science.

UNIT IV: Indian Linguistics and Formal Language Theory

Rule-based system of grammar. Concept of generative grammar and recursion. Relation of Pāṇinian grammar to formal languages and compiler design..



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

UNIT V: Ethics, Sustainability, and Human-Centric Computing

Indian worldview on knowledge, ethics, and social responsibility. Technology and Dharma: balancing progress with well-being. Environmental awareness and sustainable innovation from ancient India. Integrating IKS ethics in AI, data privacy, and digital society.

Textbooks & References

1. AICTE Model Curriculum – Indian Knowledge System (Level 1), AICTE, 2023.
2. Kapur, R., Indian Knowledge System: Relevance in Modern Context.
3. Subhash Kak, The Architecture of Knowledge: Quantum Mechanics, Neuroscience, Computers and Consciousness.
4. Pāṇini's Ashtādhyāyī (translated selections).
5. Govind Sadashiv Ghurye, Indian Logic and Its Application to Modern Science.
6. Research papers on "IKS relevance to Computer Science" (VIIRJ, 2025 Special Issue).

Course Outcomes(COs):

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

1. Explain the principles and structure of Indian Knowledge Systems.
2. Identify parallels between ancient Indian logic, linguistics, and modern computational theory.
3. Analyze mathematical and algorithmic ideas found in Indian texts and traditions.
4. Evaluate the ethical and philosophical dimensions of computing using IKS perspectives.
5. Apply IKS concepts to modern computer science challenges through case studies or mini-projects.

Mapping of Course Outcomes with Program Outcomes:														
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	3	3	3	1	1	2	1	1	2	1	3	3	3
CO2	3	3	3	3	3	1	2	-	1	2	1	3	3	3
CO3	3	3	2	3	1	1	1	-	1	-	1	3	3	3
CO4	3	3	2	3	1	1	1	-	1	-	1	3	3	3
CO5	3	3	3	3	2	1	1	1	1	2	1	3	3	3

1 – Slightly; 2 - Moderately; 3 – Substantially



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

B. Tech. (Computer Science and Engineering)
Semester VI (Third Year)

Code		Category	Course Title	Hours/Week			Total	Credits	Marks		
Department	University			L	T	P			CIE	SEE	Total
CS6601C	7-6-6-01R25	CS-PC	Compiler Design	3	0	0	3	3	40	60	100
CS6602C	7-6-6-02R25	CS-PC	Cryptography and Network Security	3	0	0	3	3	40	60	100
CS6603C	7-6-6-03R25	CS-PC	Artificial Intelligence	3	0	0	3	3	40	60	100
CS6604E	7-6-6-R25E1	CS-PE	Professional Elective - I	3	0	0	3	3	40	60	100
CS6605E	7-6-6-R25E2	CS-PE	Professional Elective - II	3	0	0	3	3	40	60	100
CS6606O	7-6-6-R25O1	CS-OE	Open Elective - II (MOOCS)	0	0	0	0	3	100	0	100
CS6607L	7-6-6-R25P1	CS-PC	Cryptography and Compiler Design Lab	0	0	3	3	1.5	40	60	100
CS6608L	7-6-6-R25P2	CS-PC	Artificial Intelligence Lab	0	0	3	3	1.5	40	60	100
CS6609S	7-6-6-R25S1	CS-SC	Full Stack Development	1	0	2	3	2	100	0	100
CS6610M	7-1-5-09R25	CS-MC	Professional Ethics in Engineering	2	0	0	2	0	100	0	100
				18	0	08	26	23	580	420	1000
Students shall undergo summer Industrial Internships for a duration of six to eight (6-8) weeks during the summer vacation by completing 45 – 60 hours of work with at least 75% attendance, to be evaluated and reflected in 7 th semester.											



Chairperson
(BoS UG)


 DEAN
 FACULTY OF ENGINEERING
 S. V. U. COLLEGE OF ENGINEERING
 TIRUPATI-517 502

COMPUTER SCIENCE & ENGINEERING
PROFESSIONAL ELECTIVE COURSES

Professional Elective - I (VI Sem)	7-6-6-R25E11	Object Oriented Analysis and Design
	7-6-6-R25E12	Soft Computing
	7-6-6-R25E13	Data mining
Professional Elective – II (VI Sem)	7-6-6-R25E21	Cloud Computing
	7-6-6-R25E22	Computer Graphics
	7-6-6-R25E23	Image Processing
	7-0-6-R25CE1	Introduction to Semiconductor Manufacturing
Professional Elective – III (VII Sem)	7-6-7-R25E11	Cyber Security
	7-6-7-R25E12	Internet of Things
	7-6-7-R25E13	Quantum Technologies and Applications
Professional Elective – IV (VII Sem)	7-6-7-R25E21	Deep Learning
	7-6-7-R25E22	Block Chain Technologies
	7-6-7-R25E23	Fog and Edge Computing
Professional Elective – V (VII Sem)	7-6-7-R25E31	Big Data Analytics
	7-6-7-R25E32	Natural Language Processing
	7-6-7-R25E33	DevOps and Agile Methodologies.



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

COMPILER DESIGN**Credits: 3****Assessment: 40+ 60**

Semester	L-T-P	Course Code	Branches
VI	3-0-0	7-6-6-01R25	CS

Prerequisites : Formal Languages and Automata Theory**Course Educational Objectives(CEOs):**

The course is designed to

1. Enrich the knowledge in various phases of compiler and its use
2. Identify different methods of lexical analysis
3. Design top-down and bottom-up parsers
4. Develop syntax directed translation schemes
5. Develop algorithms to generate code for a target machine
6. Use the tools related to compiler design effectively and efficiently

UNIT I

Introduction to Assembler, Compiler and Interpreter; Elements of ALP, Single Pass and Two Pass Assemblers, Structure of a Compiler

Lexical Analysis: Role of Lexical Analyzer, Input Buffering, Specification of Tokens, Recognition of Tokens, Lexical Analyzer Generator (Lex)

UNIT II

Syntax Analysis: Introduction, Context Free Grammars, Writing a Grammar, Top-down Parsing, Bottom-up Parsing, Introduction to LR Parsing, More Powerful LR Parsers, Introduction to YAAC.

UNIT III

Syntax Directed Translation (SDT): Syntax Definitions, Evaluation Orders for SDTs, Applications of SDTs, Schemes of SDTs

Run Time Environments: Storage Organization, Stack Allocation of Space, Access to Nonlocal Data on the Stack, Heap Management.

UNIT IV

Intermediate Code Generation: Variants of Syntax Trees, Three Address Code, Types and Declarations, Translation of Expressions, Type Checking, Control Flow Statements, Back Patching.

UNIT V

Code Generation: Issues in the Design of a Code Generator, The Target Language, Addresses in the Target Code, Basic Blocks and Flow Graphs, Optimization of Basic Blocks, A Simple Code Generator, Peephole Optimization.


Chairperson
(BoS UG)

 DEAN
 FACULTY OF ENGINEERING
 S. V. U. COLLEGE OF ENGINEERING
 TIRUPATI-517 502

Text Books:

1. Alfred V. Aho, Monica S. Lam, Ravi Sethi, Jeffrey D. Ullman, Compilers Principles, Techniques and Tools, Second Edition, Pearson Education, 2014.
2. D M Dhamdhere, Systems Programming, TMH Education, 2011.

Reference Books:

1. Jean Paul Tremblay, Paul G Serenson, "The Theory and Practice of Compiler Writing", BS Publications, 2005
2. Dhamdhere, D. M., "Compiler Construction Principles and Practice", 2nd edition, Macmillan India Ltd., New Delhi, 2008

Course Outcomes(COs):

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

1. Design a compiler for a simple programming language
2. Understand phases in the design of compiler
3. Design top-down and bottom-up parsers
4. Develop syntax directed translation schemes
5. Comprehend and adapt to Lex and Yacc tools in compiler design

Mapping of Course Outcomes with Program Outcomes:														
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	3	3	2	3	-	-	-	-	1	1	3	3	3
CO2	3	3	3	2	2	-	-	-	-	1	2	2	3	3
CO3	3	2	2	3	1	-	-	-	1	2	2	2	3	2
CO4	3	2	2	2	1	-	-	-	1	2	1	3	3	2
CO5	3	3	3	3	2	-	-	-	1	2	1	3	3	3

1 – Slightly; 2 - Moderately; 3 – Substantially



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

CRYPTOGRAPHY AND NETWORK SECURITY**Credits: 3****Assessment: 40+ 60**

Semester	L-T-P	Course Code	Branches
VI	3-0-0	7-6-6-02R25	CS

Prerequisites : Data Structures, Discrete Mathematical Structures, Computer Networks.**Course Educational Objectives (CEOs):**

1. This course aims at training students to master the:
2. Understand the basic categories of threats to computers and networks
3. Discuss various cryptographic algorithms including secret key cryptography and public-key cryptography.
4. Different encryption techniques along with hash functions, MAC, digital signatures
5. Design issues and working principles of various authentication protocols and PKI standards.

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



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

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 Books:

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

Reference Books:

2. Bernard L. Menezes, Ravinder Kumar, Cryptography, Network Security and Cyber Laws, Cengage Learning.
3. Behrouz A Forouzan, Debdeep Mukhopadhyaya, Cryptography and Network Security, 3rd Edition, Mc-GrawHill.
4. Jason Albanese, Wes Sonnenreich, Network Security Illustrated, McGraw Hill.

Course Outcomes(COs):

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

1. Apply different encryption and decryption techniques to solve problems related to confidentiality and authentication.
2. Understand key management and distribution schemes and design user authentication protocols
3. Apply different digital signature algorithms to achieve authentication and create secure applications
4. Perform simple vulnerability assessments and password audits.
5. Configure simple firewall architectures

Mapping of Course Outcomes with Program Outcomes:														
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	2	3	3	3	-	-	-	-	1	2	3	2	3
CO2	2	3	2	2	1	-	-	-	-	1	-	2	2	3
CO3	3	2	2	3	1	1	2		1	2	1	2	2	3
CO4	3	2	2	3	1	-	-	-	1	2	-	3	3	3
CO5	2	2	3	3	3	1	1	2	1	2	1	3	3	3

1 – Slightly; 2 - Moderately; 3 – Substantially



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

ARTIFICIAL INTELLIGENCE**Credits: 3****Assessment: 40+ 60**

Semester	L-T-P	Course Code	Branches
VI	3-0-0	7-6-6-03R25	CS

Prerequisites : Maths, Data Structures, Formal Languages and Automata Theory..**Course Educational Objectives(CEOs):**

The course is designed to

1. Understand the basic principles of AI toward problem solving, inference, perception, knowledge representation, and learning.
2. Introduce the concepts of machine learning and neural networks.
3. Examine the applications of AI techniques in intelligent agents, artificial neural networks and other machine learning models.

UNIT I

The History of AI: What is Intelligence, Search for Mechanical Intelligence, Evolution of Artificial Intelligence (AI), Systems Approach, Overview of topics.

Uninformed Search: General state space search, General Search Paradigms Depth-First Search, Depth-Limited Search, Iterative Deepening Search, Breadth-First Search, Bi-directional Search, Uniform-Cost Search.

Informed Search: Best-First Search, N-Queens problem, A* Search, Eight Puzzle problem, Hill Climbing Search, Simulated Annealing, Tabu Search, Constraint Satisfaction, Constraint Satisfaction algorithms: Generate and Test, Backtracking, Forward Checking and Look Ahead, Min-Conflicts Search.

UNIT II

AI and Games: Two Player Games, The Minimax Algorithm, Tic-Tac-Toe problem, Minimax with Alpha-Beta Pruning, Classical Game AI, Checkers, Chess, Scrabble, Video Game AI, Movement and Path finding, Table Lookup with Offensive and Defensive Strategy, NPC Behavior, Team AI, Real-Time Strategy AI.

Knowledge Representation (KR): Types and Role of Knowledge, Semantic Nets, Frames, Propositional Logic, First Order Logic (Predicate Logic), Semantic Web, Computational Knowledge Discovery, Ontology, Common Sense.

UNIT III

Machine Learning: Machine Learning Algorithms, Supervised Learning, Decision Trees, Unsupervised Learning, Markov Models and implementation, Nearest Neighbor Classification, 1-NN and K-NN Examples.



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

Evolutionary Computation: Introduction to Evolutionary Computation, Biological Motivation, Genetic Algorithms, Genetic Programming, Evolutionary Strategies, Differential Evolution.

UNIT IV

Neural Networks I: Concept of Neural Networks, Biological Motivation, Fundamentals of Neural Networks, The Perceptron, Least-Mean-Square (LMS) Learning, Learning with Back-propagation, Probabilistic Neural Networks, Tips for Building Neural Networks. Neural Networks II: Unsupervised Learning, Hebbian Learning, Simple Competitive Learning, k-Means Clustering, Adaptive Resonance Theory, Hopfield Auto-Associative Model.

UNIT V

Robotics and AI: Introduction, Taxonomy of Robotics, Hard vs. Soft Robotics, Braitenburg Vehicles, Natural Sensing and Control, Perception with Sensors, Actuation with Effectors, Robotic Control Systems, Simple Control Architectures, Movement Planning, Distributed Robotics.

Intelligent Agents: Anatomy of an Agent, Agent Properties and AI, Hybrid Agent, Agent Architectures, Types of Architectures, Agent Languages, Agent Communication. Biologically Inspired and Hybrid Models: Cellular Automata, Artificial Immune Systems, Artificial Life, Fuzzy Logic, Evolutionary Neural Networks, Ant Colony Optimization, Affective Computing.

Text Books:

1. M Tim Jones, Artificial Intelligence - A Systems Approach, Infinity Science Press, 2008.

Reference Books:

2. Russel S, Norvig P, Artificial Intelligence: A Modern Approach, 3rd edition, Pearson Education, 2010.
3. Rich E, Knight K, Nair S B, Artificial Intelligence, 3rd edition, Tata McGraw-Hill, 2009.
4. Luger G F, Artificial Intelligence, 6th edition, Pearson Education, 2009.
5. Carter M, Minds and Computers: An Introduction to the Philosophy of Artificial Intelligence, Edinburgh University Press, 2007.
6. Coppin B, Artificial Intelligence Illuminated, Jones & Bartlett, 2004.
7. Ertel W, Introduction to Artificial Intelligence, Springer, 2011.

Course Outcomes(COs):

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

1. Understanding of artificial intelligence fundamentals and search Techniques
2. Design of Intelligent Game Strategies and Possess the skill for representing knowledge using the appropriate technique for a given problem.
3. Ability to apply Machine Learning concepts & Evolutionary Computation concepts for Problem Solving
4. Ability to apply Neural Networks for real world problems
5. Ability to understand the concepts of Intelligent Agents, Robotics & Hybrid AI



Chairperson
(BoS UG)


 DEAN
 FACULTY OF ENGINEERING
 S. V. U. COLLEGE OF ENGINEERING
 TIRUPATI-517 502

Mapping of Course Outcomes with Program Outcomes:														
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	3	2	2	1	-	-		-	-	-	-	-	-
CO2	3	3	3	2	2	-	-	1	1	1	-	-	-	-
CO3	3	3	3	3	2	-	-	1	1	-	-	1	-	-
CO4	3	2	3	3	2	-	-	1	1	-	-	1	-	-
CO5	3	3	3	2	3	2	1	1	2	2	1	2	-	-

1 – Slightly; 2 - Moderately; 3 – Substantially



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

CRYPTOGRAPHY AND COMPILER DESIGN LABORATORY**Credits: 1.5****Assessment: 40+ 60**

Semester	L-T-P	Course Code	Branches
VI	0-0-3	7-6-6-R25P1	CS

At least 10 Experiments should be given, covering the topics of the course Cryptography and Compiler Design Laboratory.

Experiment 1: Study of Security Attacks and Mechanisms

Aim: To analyze different types of security attacks, services, and mechanisms.

Description: Study various security attacks such as passive and active attacks. Understand security services, mechanisms, fundamental security design principles, attack surfaces, attack trees, and the model for network security.

Experiment 2: Mathematical Foundations of Cryptography

Aim: To implement basic mathematical concepts used in cryptography.

Description: Perform problems on algebraic structures (groups, rings, fields, Galois fields), Euclidean algorithm, modular arithmetic, prime number testing, Fermat's and Euler's theorems, Chinese Remainder Theorem, and discrete logarithms.

Experiment 3: Implementation of Classical Encryption Techniques

Aim: To implement and analyze classical substitution and transposition ciphers.

Description: Implement Caesar cipher, Playfair cipher, Hill cipher, and transposition techniques. Study rotor machines and basic steganography techniques.

Experiment 4: Study and Implementation of DES and AES

Aim: To understand block cipher structure and implement DES and AES algorithms.

Description: Study traditional block cipher structure, DES algorithm, AES structure, AES transformation functions, key expansion, and compare strength of DES and AES.

Experiment 5: Block Cipher Modes of Operation

Aim: To implement different block cipher modes of operation.

Description: Implement ECB, CBC, CFB, OFB, and CTR modes. Analyze the security implications of each mode and perform multiple encryption including Triple DES.



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

Experiment 6: Implementation of Public Key Cryptography Algorithms

Aim: To implement RSA and Diffie-Hellman algorithms.

Description: Implement RSA algorithm for encryption and decryption. Perform Diffie-Hellman key exchange and study ElGamal and Elliptic Curve Cryptography concepts.

Experiment 7: Cryptographic Hash Functions and Message Authentication Codes

Aim: To implement SHA and HMAC algorithms.

Description: Generate message digests using SHA. Implement HMAC and CMAC for message authentication and analyze security requirements.

Experiment 8: Digital Signature Algorithms

Aim: To implement and verify digital signature schemes.

Description: Implement ElGamal Digital Signature Scheme, Schnorr Digital Signature Scheme, NIST DSA, and ECDSA. Perform signing and verification of messages.

Experiment 9: Key Management and User Authentication

Aim: To study key distribution techniques and authentication mechanisms.

Description: Study symmetric and asymmetric key distribution, X.509 certificates, Public Key Infrastructure, Kerberos authentication, and remote user authentication methods.

Experiment 10: Transport and Network Level Security Protocols

Aim: To study TLS, SSH, IP Security, and Email Security protocols.

Description: Analyze Transport Layer Security (TLS), Secure Shell (SSH), S/MIME, PGP, IP Security (ESP, Security Associations), and Internet Key Exchange.

Compiler Design Laboratory Experiments

Experiment 1: Study of Assembler, Compiler and Interpreter

Aim: To understand the differences between assembler, compiler, and interpreter.

Description: Study the basic structure of assemblers (single pass and two pass), elements of Assembly Language Programming (ALP), and the overall structure of a compiler.

Experiment 2: Implementation of Lexical Analyzer

Aim: To design and implement a lexical analyzer.

Description: Implement a lexical analyzer to recognize tokens such as keywords, identifiers, operators, and constants. Demonstrate input buffering and token specification.



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

Experiment 3: Lexical Analyzer using Lex Tool

Aim: To generate a lexical analyzer using Lex.

Description: Write Lex programs to recognize tokens and generate a lexical analyzer automatically. Test the generated analyzer with sample input programs.

Experiment 4: Construction of Context Free Grammar and Parsing

Aim: To construct CFG for a given language and perform parsing.

Description: Write context free grammars for arithmetic expressions and implement top-down parsing techniques (Recursive Descent Parser).

Experiment 5: Bottom-Up and LR Parsing

Aim: To implement bottom-up parsing techniques.

Description: Implement Shift-Reduce parsing and construct LR parsing tables. Study introduction to LR, SLR, and LALR parsers using YACC.

Experiment 6: Syntax Directed Translation (SDT)

Aim: To implement syntax directed definitions and translation schemes.

Description: Develop SDTs for arithmetic expressions. Demonstrate evaluation order and attribute computation for translation.

Experiment 7: Run Time Environment and Storage Organization

Aim: To study storage allocation strategies.

Description: Implement stack allocation, simulate activation records, access to non-local data, and study heap management techniques.

Experiment 8: Intermediate Code Generation

Aim: To generate intermediate code for given expressions.

Description: Generate syntax trees and three-address code for arithmetic expressions. Perform type checking and handle declarations.

Experiment 9: Control Flow and Backpatching

Aim: To implement translation of control flow statements.

Description: Generate intermediate code for conditional and looping statements. Implement backpatching techniques for boolean expressions.

Experiment 10: Code Generation and Optimization

Aim: To design a simple code generator and apply optimization techniques.

Description: Construct basic blocks and flow graphs. Implement a simple code generator and perform peephole optimization and optimization of basic blocks.



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

Course Outcomes(COs):

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

1. Design a compiler for a simple programming language
2. Understand phases in the design of compiler
3. Design top-down and bottom-up parsers
4. Develop syntax directed translation schemes
5. Comprehend and adapt to Lex and Yacc tools in compiler design

Mapping of Course Outcomes with Program Outcomes:														
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	2	1	2	3				1	1	1	3	1	3
CO2	2	3	3	2	2				1	1	2	2	2	3
CO3	2	2	2	3	1	1	1	1	1	2	2	2	2	2
CO4	3	2	2	2	1	1	2	2	1	2	1	3	1	2
CO5	3	3	3	3	2	1	2	2	1	2	1	3	1	3

1 – Slightly; 2 - Moderately; 3 – Substantially



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

ARTIFICIAL INTELLIGENCE LABORATORY**Credits: 1.5****Assessment: 40+ 60**

Semester	L-T-P	Course Code	Branches
VI	0-0-3	7-6-6-R25P2	CS

At least 10 Experiments should be given covering the topics of the course, Artificial Intelligence.

Experiment 1: Study of History and Foundations of AI

Aim: To understand the evolution and foundations of Artificial Intelligence.

Description: Study the history of AI, concept of intelligence, mechanical intelligence, systems approach, and overview of AI domains.

Experiment 2: Implementation of Depth First and Breadth First Search

Aim: To implement uninformed search algorithms.

Description: Implement DFS, BFS, Depth-Limited Search and Iterative Deepening Search for solving state space problems.

Experiment 3: Uniform Cost and Bi-Directional Search

Aim: To implement cost-based search strategies.

Description: Implement Uniform Cost Search and Bi-directional Search for graph problems.

Experiment 4: Best First and A* Search Algorithms

Aim: To implement informed search techniques.

Description: Implement Best-First Search and A* algorithm to solve Eight Puzzle problem.

Experiment 5: N-Queens and Constraint Satisfaction Problems

Aim: To solve CSP problems using different techniques.

Description: Implement N-Queens using Backtracking, Forward Checking, Min-Conflicts and Generate-and-Test methods.

Experiment 6: Hill Climbing and Simulated Annealing

Aim: To implement local search algorithms.

Description: Implement Hill Climbing, Simulated Annealing and Tabu Search for optimization problems.



Chairperson
(BoS UG)


 DEAN
 FACULTY OF ENGINEERING
 S. V. U. COLLEGE OF ENGINEERING
 TIRUPATI-517 502

Experiment 7: Implementation of Minimax Algorithm

Aim: To implement game playing using Minimax.

Description: Develop Tic-Tac-Toe game using Minimax algorithm.

Experiment 8: Alpha-Beta Pruning in Game AI

Aim: To optimize Minimax using Alpha-Beta pruning.

Description: Implement Minimax with Alpha-Beta pruning for two-player games.

Experiment 9: Path Finding and NPC Behavior

Aim: To implement AI techniques in video games.

Description: Implement pathfinding algorithms and simulate NPC behavior strategies.

Experiment 10: Knowledge Representation using Logic

Aim: To represent knowledge using logical methods.

Description: Implement Propositional Logic and First Order Logic representations. Develop semantic networks and frames.

Experiment 11: Ontology and Semantic Web Basics

Aim: To study ontology and semantic web technologies.

Description: Develop a simple ontology and demonstrate knowledge discovery techniques.

Experiment 12: Supervised Learning using Decision Trees

Aim: To implement supervised learning algorithms.

Description: Implement Decision Tree algorithm and evaluate classification results.

Experiment 13: Nearest Neighbor Classification

Aim: To implement K-NN classifier.

Description: Implement 1-NN and K-NN algorithms with suitable dataset examples.

Experiment 14: Markov Models Implementation

Aim: To implement Markov models.

Description: Build and simulate simple Markov Chains and Hidden Markov Models.

Experiment 15: Genetic Algorithms Implementation

Aim: To implement evolutionary computation techniques.

Description: Implement Genetic Algorithm for optimization problem. Study genetic programming and differential evolution basics.

Experiment 16: Implementation of Perceptron and LMS

Aim: To implement basic neural network models.

Description: Implement Perceptron learning and Least Mean Square algorithm.



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

Experiment 17: Backpropagation Neural Network

Aim: To implement multi-layer neural network.

Description: Implement Backpropagation algorithm and evaluate performance.

Experiment 18: Unsupervised Learning and Clustering

Aim: To implement unsupervised learning techniques.

Description: Implement Hebbian Learning, Competitive Learning and K-Means clustering.

Experiment 19: Robotics and Intelligent Agents

Aim: To study robotics concepts and agent architectures.

Description: Simulate simple robotic control system and design intelligent agent models.

Experiment 20: Biologically Inspired and Hybrid Models

Aim: To implement bio-inspired AI models.

Description: Implement Cellular Automata, Fuzzy Logic system, Ant Colony Optimization and study Artificial Immune Systems concepts.

Course Outcomes(COs):

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

- 1) Understanding of artificial intelligence fundamentals and search Techniques
- 2) Design of Intelligent Game Strategies and Possess the skill for representing knowledge using the appropriate technique for a given problem.
- 3) Ability to apply Machine Learning concepts & Evolutionary Computation concepts for Problem Solving
- 4) Ability to apply Neural Networks for real world problems
- 5) Ability to understand the concepts of Intelligent Agents, Robotics & Hybrid AI

Mapping of Course Outcomes with Program Outcomes:														
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	3	2	2	1								2	1
CO2	3	3	3	2	2			1	1	1			2	2
CO3	3	3	3	3	2			1	1			1	3	2
CO4	3	2	3	3	2			1	1			1	3	2
CO5	3	3	3	2	3	2	1	1	2	2	1	2	2	3

1 – Slightly; 2 - Moderately; 3 – Substantially



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

FULL STACK DEVELOPMENT**Credits: 2****Assessment: 100+0**

Semester	L-T-P	Course Code	Branches
VI	1-0-2	7-6-6-R25S1	CS

Prerequisites: Basic Web Design, Programming Skills, Data Structures.**Course Educational Objectives (CEOs):**

The main objectives of the course are to

1. Make use of Modern- day Java Script with ES6 standards for designing Dynamic web pages
2. Building robust & responsive User Interfaces using popular JavaScript library **React.js**‘.
3. Building robust backend APIs using **_Express. js**‘
4. Establishing the connection between frontend (React) User interfaces and backend APIs (Express) with Data Bases (My SQL)
5. Familiarize students with GitHub for remote repository hosting and collaborative development.

Experiments covering the Topics:

- Introduction to DOM (Document Object Model), Ecma Script (ES6) standards and features like Arrow functions, Spread operator, Rest operator, Type coercion, Type hoisting, String literals, Array and Object Destructuring.
- Basics of React. js like React Components, JSX, Conditional rendering Differences between Real DOM and Virtual DOM.
- Important React.js concepts like React hooks, Props, React forms, Fetch API, Iterative rendering using JavaScript map() function.
- Java Script runtime environment node. js and its uses, Express. js and Routing, Micro-Services architecture and MVC architecture, database connectivity using (My SQL)
- Introduction to My SQL, setting up MySQL and configuring, Databases, My SQL queries, sub queries, creating My SQL driver for database connectivity to Express. js server.
- Introduction to Git and Git Hub and upload project& team collaboration

Sample Experiments:**1. Introduction to Modern JavaScript and DOM**

- a. Write a JavaScript program to link JavaScript file with the HTML page
- b. Write a JavaScript program to select the elements in HTML page using selectors
- c. Write a JavaScript program to implement the event listeners
- d. Write a JavaScript program to handle the click events for the HTML button elements
- e. Write a JavaScript program to with three types of functions
 - i. Function declaration
 - ii. Function definition
 - iii. Arrow functions


Chairperson
(BoS UG)


DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

2. Basics of React. js

- a. Write a React program to implement a counter button using react class components
- b. Write a React program to implement a counter button using react functional components
- c. Write a React program to handle the button click events in functional component
- d. Write a React program to conditionally render a component in the browser
- e. Write a React program to display text using String literals

3. Important concepts of React. js

- a. Write a React program to implement a counter button using React use State hook
- b. Write a React program to fetch the data from an API using React use Effect hook
- c. Write a React program with two react components sharing data using Props.
- d. Write a React program to implement the forms in react
- e. Write a React program to implement the iterative rendering using map() function.

4. Introduction to Git and GitHub

a. Setup

- o Install Git on local machine.
- o Configure Git (user name, email).
- o Create GitHub account and generate a personal access token.

b. Basic Git Workflow

- o Create a local repository using git init
- o Create and add files → git add .
- o Commit files → git commit -m "Initial commit"
- o Connect to GitHub remote → git remote add origin <repo_url>
- o Push to GitHub → git push -u origin main

c. Branching and Collaboration

- o Create a branch → git checkout -b feature1
- o Merge branch to main → git merge feature1
- o Resolve merge conflicts (guided)

5. Upload React Project to GitHub

- o Create a new React app using npx create-react-app myapp
- o Initialize a git repo and push to GitHub
- o Use .gitignore to exclude node_modules
- o Create multiple branches: feature/navbar, feature/form
- o Practice merge and pull requests (can use GitHub GUI)

6. Introduction to Node. js and Express. js

- a. Write a program to implement the ‘hello world’ message in the route through the browser using Express



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

- b. Write a program to develop a small website with multiple routes using Express. js
- c. Write a program to print the 'hello world' in the browser console using Express. js
- d. Write a program to implement the CRUD operations using Express. js
- e. Write a program to establish the connection between API and Database using Express – My SQL driver

7. Introduction to My SQL

- a. Write a program to create a Database and table inside that database using My SQL Command line client
- b. Write a My SQL queries to create table, and insert the data, update the data in the table
- c. Write a My SQL queries to implement the subqueries in the My SQL command line client
- d. Write a My SQL program to create the script files in the My SQL workbench
- e. Write a My SQL program to create a database directory in Project and initialize a database. sql file to integrate the database into API

8. Team Collaboration Using GitHub

- o Form groups of 2–3 students
- o Create a shared GitHub repo
- o Assign tasks and work in branches
- o Use Issues, Pull Requests, and Code Reviews
- o Document code with README.md

Textbooks:

1. Web Design with HTML, CSS, JavaScript and JQuery Set Book by Jon Duckett Professional JavaScript for Web Developers Book by Nicholas C. Zakas
2. John Dean, Web Programming with HTML5, CSS and JavaScript, Jones & Bartlett Learning, 2019.
3. Pro MERN Stack: Full Stack Web App Development with Mongo, Express, React, and Node, Vasam Subramanian, 2nd edition, APress, O'Reilly.
4. Learning PHP, MySQL, JavaScript, CSS & HTML5: A Step-by-Step Guide to Creating Dynamic Websites by Robin Nixon
5. AZAT MARDAN, Full Stack Java Script: Learn Back bone. js, Node.js and Mongo DB. 2015

Reference Books:

6. Full-Stack JavaScript Development by Eric Bush.
7. Programming the World Wide Web, 7th Edition, Robert W. Sebesta, Pearson, 2013.
8. Tomasz Dyl, Kamil Przeorski, Maciej Czarnecki, Mastering Full Stack React Web Development 2017



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

Online Learning Resources:

<https://ict.iitk.ac.in/product/full-stack-developer-html5-css3-js-bootstrap-php-4/>

<https://www.w3schools.com/html>

<https://www.w3schools.com/css>

<https://www.w3schools.com/js/>

<https://www.w3schools.com/nodejs>

<https://www.w3schools.com/typescript>

<https://docs.github.com/>

<https://education.github.com/git-cheat-sheet-education.pdf>

Course Outcomes(COs):

1. Building fast and interactive UIs
2. Applying Declarative approach for developing web apps
3. Understanding ES6 features to embrace modern JavaScript
4. Building reliable APIs with Express. Js
5. Create and manage Git repositories, track changes, and push code to GitHub.

Mapping of Course Outcomes with Program Outcomes:														
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	2	3	3	3	-	-	-	-	2	2	2	3	3
CO2	2	2	3	3	3	-	-	-	-	2	2	2	3	3
CO3	2	2	2	3	1	-	-	-	-	2	2	2	3	3
CO4	3	2	2	2	1	-	-	-	-	2	2	2	3	3
CO5	3	3	3	2	2	-	-	-	-	2	2	2	3	3

1 – Slightly; 2 - Moderately; 3 – Substantially



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

PROFESSIONAL ETHICS IN ENGINEERING**Credits: 0****Assessment: 100+0**

Semester	L-T-P	Course Code	Branches
VI	2-0-0	7-1-5-09R25	CS

Prerequisites: NIL**Course Educational Objectives (CEOs):**

1. To understand the core values that shape a professional character, such as integrity, work ethic, and civic virtue.
2. To create awareness about engineering ethics and theories of moral development (Kohlberg & Gilligan).
3. To instil a sense of responsibility regarding safety, risk assessment, and the experimental nature of engineering projects.
4. To educate students on their rights and responsibilities in a workplace, including Intellectual Property Rights (IPR) and confidentiality.
5. To broaden the perspective on global issues, such as environmental ethics, computer ethics, and the role of engineers as managers and consultants.

UNIT-I**HUMAN VALUES**

Morals, values and Ethics–Integrity–Work ethic–Service learning–Civic virtue–Respect for others–Living peacefully – Caring – Sharing – Honesty – Courage – Valuing time – Cooperation – Commitment – Empathy – Self Confidence–Character–Spirituality–Introduction to Yoga and meditation for professional excellence and stress management.

UNIT-II**ENGINEERING ETHICS**

Senses of Engineering Ethics –Variety of moral issues–Types of inquiry–Moral dilemmas–Moral Autonomy – Kohlberg's theory– Gilligan's theory– Consensus and Controversy– Models of professional roles – Theories about right action – Self-interest – Customs and Religion – Uses of Ethical Theories.

UNIT-III**ENGINEERING AS SOCIAL EXPERIMENTATION**

Engineering as Experimentation – Engineers as responsible Experimenters – Codes of Ethics – A Balanced Outlook on Law.

UNIT-IV**SAFETY, RESPONSIBILITIES AND RIGHTS**

Safety and Risk – Assessment of Safety and Risk – Risk Benefit Analysis and Reducing Risk – Respect for Authority– Collective Bargaining – Confidentiality– Conflicts of Interest – Occupational Crime – Professional Rights – Employee Rights – Intellectual Property Rights (IPR) – Discrimination.



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

UNIT-V GLOBAL ISSUES

Multinational Corporations – Environmental Ethics – Computer Ethics – Weapons Development – Engineers as Managers – Consulting Engineers – Engineers as Expert Witnesses and Advisors – Moral Leadership – Code of Conduct – Corporate Social Responsibility.

TEXTBOOKS:

1. Mike W. Martin and Roland Schinzinger, *Ethics in Engineering*, Tata McGraw Hill, New Delhi, 2003.
2. Govindarajan M, Natarajan S, Senthil Kumar V. S, *Engineering Ethics*, Prentice Hall of India, New Delhi, 2004.

REFERENCES:

1. Charles B. Fleddermann, —*Engineering Ethics*, Pearson Prentice Hall, New Jersey, 2004.
2. Charles E. Harris, Michael S. Pritchard and Michael J. Rabins, —*Engineering Ethics – Concepts and Cases*, Cengage Learning, 2009.
3. John R. Boatright, —*Ethics and the Conduct of Business*, Pearson Education, New Delhi, 2003.
4. Edmund G. Seebauer and Robert L. Barry, —*Fundamentals of Ethics for Scientists and Engineers*, Oxford University Press, Oxford, 2001.
5. Laura P. Hartman and Joe Desjardins, —*Business Ethics: Decision Making for Personal Integrity and Social Responsibility* Mc Graw Hill education, India Pvt. Ltd., New Delhi, 2013.
6. World Community Service Centre, *Value Education*, Vethathiri Publications, Erode, 2011

COURSE OUTCOMES (COs):

1. Apply holistic human values, such as integrity, honesty, and empathy, to their personal and professional lives to maintain harmony.
2. Apply ethical theories (like Utilitarianism or Duty ethics) to resolve moral dilemmas and distinguish between ethical and unethical professional behavior.
3. Analyze engineering projects as social experiments, demonstrating an understanding of codes of ethics and the need for responsible experimentation.
4. Assess safety and risk in engineering practices and defend professional rights and responsibilities, including intellectual property and whistleblowing.
5. Evaluate global challenges—such as environmental degradation, computer ethics, and weapons development—and apply ethical principles to multinational corporate scenarios.



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

PO&PSO CO	PO(1)	PO(2)	PO(3)	PO(4)	PO(5)	PO(6)	PO(7)	PO(8)	PO(9)	PO(10)	PO(11)	PO(12)		PSO (1)	PSO (2)
CO1	-	-	-	-	-	2		3	3	-	-	3		1	2
CO2	-	-	-	-	-	2		3	2	-	-	2		1	1
CO3	-	-	-	-	-	3	1	3	2	-	-	2		2	1
CO4	-	-	-	-	-	3		3	2	1	-	2		2	2
CO5	-	-	-	-	-	2	3	3	1	-	-	2		1	2

1 - Slightly; 2 - Moderately; 3 – Substantially



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

PE -I - 1 - OBJECT ORIENTED ANALYSIS AND DESIGN**Credits: 3****Assessment: 40+ 60**

Semester	L-T-P	Course Code	Branches
VI	3-0-0	7-6-6-R25E11	CS

Prerequisites: Data Structures, Object Oriented Programming, Software Engineering.**Course Educational Objectives(CEOs):**

The course is designed to

1. Understand the Object-based view of systems
2. Develop robust object-based models for systems
3. Inculcate necessary skills to handle complexity in software design
4. Understand the notations of Unified Modelling Language

UNIT I**OOAD Basic Concepts – Complexity:** The Structure of Complex Systems, The Inherent Complexity of Software, The Five Attributes of a Complex System, Organized and Disorganized Complexity, Bringing Order to Chaos, On Designing Complex Systems.**The Object Model:** The Evolution of the Object Model, Foundations of the Object Model, Elements of the Object Model, Applying the Object Model.**UNIT II****Classes and Objects:** The Nature of an Object, Relationships among Objects, The Nature of a Class, Relationships among Classes, The Interplay of Classes and Objects, On Building Quality Classes and Objects.**Classification:** The Importance of Proper Classification, Identifying Classes and Objects, Key Abstractions and Mechanisms.**UNIT III****Method – Notation:** The Unified Modeling Language, Package Diagrams, Component Diagrams, Deployment Diagrams, Use Case Diagrams, Activity Diagrams, Class Diagrams, Sequence Diagrams, Interaction Overview Diagrams, Composite Structure Diagrams, State Machine Diagrams, Timing Diagrams, Object Diagrams, Communication Diagrams.**UNIT IV****Process:** First Principles, The Macro Process: The Software Development Lifecycle, The Micro Process: The Analysis and Design Process.**Pragmatics:** Management and Planning, Staffing, Release Management, Reuse, Quality Assurance and Metrics, Documentation, Tools, Special Topics, The Benefits and Risks of Object-Oriented Development.

Chairperson
(BoS UG)

 DEAN
 FACULTY OF ENGINEERING
 S. V. U. COLLEGE OF ENGINEERING
 TIRUPATI-517 502

UNIT V

Applications - System Architecture: Satellite-Based Navigation - Inception, Elaboration, Construction, Post-Transition.

Artificial Intelligence: Cryptanalysis - Inception, Elaboration, Construction, Post-Transition.

Web Application: Vacation Tracking System - Inception, Elaboration, Construction, Transition and Post-Transition.

Text Books:

1. Grady Booch, Robert A. Maksimchuk, Michael W. Engle, Bobbi J. Young, Jim Conallen, Kelli A. Houston, Object-Oriented Analysis and Design with Applications, Addison-Wesley/Pearson, Third Edition, 2015.

Reference Books:

1. Michael Blaha and James Rumbaugh, Object-oriented modelling and design with UML, Prentice-Hall of India, 2005.
2. Craig Larman, Applying UML and Patterns – An introduction to Object-Oriented Analysis and Design and Iterative Development, 3rd edition, Pearson Education, 2005.
3. Ali Bahrami, Object Oriented Systems Development, McGraw-Hill, 1999.
4. Booch Grady, Object Oriented Analysis and Design, 2nd edition Pearson Education, 2000.
5. Fowler, Martin, UML Distilled, 3rd edition, Pearson Education, 2004.
6. Lunn, Ken, Software development with UML, Palgrave Macmillan, 2003.
7. O'Docherty, Mike, Object-Oriented Analysis & Design, Wiley, 2005

Course Outcomes(COs):

Having successfully completed this course the students will be able to:

1. Understand software complexity and object-oriented principles
2. Identify and design classes and objects using proper classification
3. Model systems using UML diagrams.
4. Apply OO development processes and project management practices.
5. Design OO solutions for real-world applications

Mapping of Course Outcomes with Program Outcomes:														
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	2	3	3	3	-	-	-	1	2	1	3	3	3
CO2	3	2	3	3	3	-	-	-	-	2	1	-	3	3
CO3	3	2	2	3	1	-	-	-	-	-	1	-	3	3
CO4	3	2	2	3	1	-	-	-	1	-	1	3	3	3
CO5	3	3	3	3	2	-	-	-	1	2	1	3	3	3

1 – Slightly; 2 - Moderately; 3 – Substantially



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

PE -I - 2 - SOFT COMPUTING**Credits: 3****Assessment: 40+ 60**

Semester	L-T-P	Course Code	Branches
VI	3-0-0	7-6-6-R25E12	CS

Prerequisites: Artificial Intelligence.**Course Educational Objectives(CEOs):**

The course is designed to

1. Understand basic ANN architectures, algorithms and their limitations
2. Understand the concepts of feed forward and feedback ANN.
3. Discuss the Fuzzy logic concepts, Fuzzy principles and relations
4. Familiarize Genetic Algorithm and its applications to Soft Computing.
5. Design and develop ML techniques with assistance of MATLAB

UNIT I

Introduction of soft computing - Soft computing vs. Hard computing- Various types of soft computing techniques- Applications of soft computing-Neuron- Nerve structure and synapse- Artificial Neuron and its model- Activation functions- Neural network architecture- Single layer and Multilayer feed forward networks- McCullochPitts neuron model- Perceptron model- MLP- Back propagation learning methods- Effect of learning rule coefficient.

UNIT II

Counter propagation network - Architecture - Functioning & Characteristics of counter- propagation network - Hopfield/ Recurrent network – Configuration - Stability constraints associative memory- and characteristics- Limitations and applications- Hopfield vs Boltzman machine - Adaptive Resonance Theory – Architecture – Classifications -Implementation and training-Associative Memory.

UNIT III

Different faces of imprecision - Inexactness, Ambiguity, Undecidability, Fuzziness and certainty, Fuzzy sets and Crisp sets. Intersections of Fuzzy sets, Union of Fuzzy sets, the complement of Fuzzy sets - Fuzzy reasoning. Linguistic variables, Fuzzy propositions, Fuzzy compositional rules of inference - Methods of decompositions and Defuzzification.

UNIT IV

Basic concept of Genetic algorithm and detail algorithmic steps - Adjustment of free parameters- Solution of typical control problems using Genetic algorithm - Concept on some other search techniques like tabu search and ant colony search techniques for solving optimization problems.



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

UNIT V

GA application to optimization problems - Case studies: Identification and control of linear and nonlinear dynamic systems using MATLAB - Neural Network toolbox. Stability analysis of Neural Network interconnection systems - Implementation of fuzzy logic controller using MATLAB fuzzy logic toolbox - Stability analysis of fuzzy control systems.

Text Books:

1. Timothy J. Ross, "Fuzzy Logic with Engineering Applications", Wiley India, 3rd edition, 2012.
2. Zimmermann H. J. "Fuzzy set theory and its Applications" Springer international edition, 2011.
3. David E. Goldberg, "Genetic Algorithms in Search, Optimization, and Machine Learning", Pearson Education, 2009.
4. Laurene V. Fausett, "Fundamentals of Neural Networks: Architectures, Algorithms, And Applications", Pearson Education, 1st edition, 1993.
5. W. T. Miller, R. S. Sutton and P. J. Webros, "Neural Networks for Control", MIT Press, 1996.

Reference Books:

6. S. Rajasekaran, and G. A. Vijayalakshmi Pai, Neural Networks, Fuzzy Logic and Genetic Algorithms: Synthesis, and Applications, Prentice Hall of India, 2007.

Course Outcomes(COs):

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

1. Understand basic ANN architectures, algorithms and their limitations
2. Comprehend the fuzzy logic and the concept of fuzziness involved in various systems.
3. Understand the role of soft computing techniques in solving real world applications
4. Build optimal classifiers using genetic algorithms
5. Implement fuzzy logic controller using MATLAB fuzzy logic toolbox

Mapping of Course Outcomes with Program Outcomes:															
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	
CO1	3	2	3	3	3	-	-	-	-	-	1	1	3	3	
CO2	3	2	3	3	3	-	-	-	-	-	2	2	3	2	
CO3	3	2	2	3	1	-	-	-	1	-	2	2	3	2	
CO4	3	2	2	3	1	-	-	-	1	-	1	1	3	3	
CO5	3	3	3	3	2	-	-	-	1	-	1	1	3	3	

1 – Slightly; 2 - Moderately; 3 – Substantially



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

PE -I - 3 - DATA MINING**Credits: 3****Assessment: 40+ 60**

Semester	L-T-P	Course Code	Branches
VI	3-0-0	7-6-6-R25E13	CS

Prerequisites: Artificial Intelligence, DBMS.**Course Educational Objectives (CEOs):**

The course is designed to

1. Understand the principles of Data Warehousing and Data Mining
2. Know the Architecture of a Data Mining system
3. Learn pre-processing techniques and data mining functionalities
4. Compare and contrast classification and clustering algorithms

UNIT I

Introduction: Why Data Mining? What Is Data Mining? What Kinds of Data Can Be Mined? What Kinds of Patterns Can Be Mined? Which Technologies Are Used? Which Kinds of Applications Are Targeted? Major Issues in Data Mining.

Data Objects and Attribute Types, Basic Statistical Descriptions of Data, Data Visualization, Measuring Data Similarity and Dissimilarity.

Data Preprocessing: An Overview, Data Cleaning, Data Integration, Data Reduction, Data Transformation and Data Discretization.

UNIT II

Data Warehouse: Basic Concepts, Data Warehouse Modeling: Data Cube and OLAP, Data Warehouse Design and Usage, Data Warehouse Implementation, Data Generalization by Attribute-Oriented Induction.

Data Cube Computation, Data Cube Computation Methods, Processing Advanced Kinds of Queries by Exploring Cube Technology, Multidimensional Data Analysis in Cube Space.

UNIT III

Mining Frequent Patterns, Associations and Correlations: Basic Concepts, Frequent Item set Mining Methods, Which Patterns Are Interesting?—Pattern Evaluation Methods.

Advanced Pattern Mining : A Road Map, Pattern Mining in Multilevel, Multidimensional Space, Constraint-Based Frequent Pattern Mining, Mining High-Dimensional Data and Colossal Patterns, Mining Compressed or Approximate Patterns, Pattern Exploration and Application.

Classification: Basic Concepts, Decision Tree Induction, Bayes Classification Methods, Rule-Based Classification, Model Evaluation and Selection, Techniques to Improve Classification Accuracy.



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

UNIT IV

Classification-Advanced Methods: Bayesian Belief Networks, Classification by Back propagation, Support Vector Machines, Classification Using Frequent Patterns, Lazy Learners (or Learning from Your Neighbors), Other Classification Methods, Additional Topics Regarding Classification.

Cluster Analysis: Basic Concepts, Partitioning Methods, Hierarchical Methods, Density-Based Methods, Grid-Based Methods, Evaluation of Clustering.

UNIT V

Advanced Cluster Analysis: Probabilistic Model-Based Clustering, Clustering High-Dimensional Data, Clustering Graph and Network Data, Clustering with Constraints.

Outlier Detection: Outliers and Outlier Analysis, Outlier Detection Methods, Statistical Approaches, Proximity-Based Approaches, Clustering-Based Approaches, Classification-Based Approaches, Mining Contextual and Collective Outliers, Outlier Detection in High-Dimensional Data.

Overview of Data Mining Trends and Research Frontiers: Mining Complex Data Types, Other Methodologies of Data Mining, Data Mining Applications, Data Mining and Society, Data Mining Trends.

Text Books:

1. Jiawei Han, Micheline Kamber and Jian Pei, Data Mining: Concepts and Techniques, 3rd Edition, Morgan Kaufmann Publishers, Elsevier, 2012.

Reference Books:

2. Jiawei Han and Micheline Kamber, Data Mining: Concepts and Techniques, 2nd Edition, Morgan Kaufmann Publishers, Elsevier, 2006.
3. Pang-Ning Tan, Vipin Kumar, Michael Steinbach, Introduction to Data Mining, Pearson Education, 2016.
4. Hongbo Du, Data mining Techniques and Applications: An Introduction, 1st Edition, Cengage India Publishing, 2013.
5. Arun K Pujari, Data Mining Techniques, 3rd Edition, Universities Press, 2013.
6. T.V Suresh Kumar, B Eswara Reddy, Jagadish S Kallimani, Data Mining: Principles and Applications, First edition, Elsevier, 2012.
7. Vikram Pudi, P Radha Krishna, Data Mining, Oxford University Press, 2009.
8. Sam Anahory and Dennis Murray, Data Warehousing in the Real World: A Practical Guide for Building Decision Support Systems, First Edition, Pearson Education India, 2002.
9. K.P.Soman, Shyam Diwakar, V.Ajay, Insight Into Data Mining: Theory and Practice, Prentice Hall India, 2006.



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

Course Outcomes(COs):

By the end of this course students will be able to

1. Comprehend the various architectures and its application with data mining
2. Design and develop data mining algorithms to analyze raw real world data
3. Apply preprocessing techniques for data cleansing
4. Analyze multi-dimensional modeling techniques and Classification & Clustering algorithms.
5. Compare and contrast classification and clustering algorithms

Mapping of Course Outcomes with Program Outcomes:														
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	2	3	3	3	-	-	-	-	-	1	3	3	3
CO2	3	2	3	3	3	-	-	-	-	-	1	3	3	3
CO3	3	3	2	3	1	-	-	-	1	-	1	-	3	3
CO4	3	2	3	3	1	-	-	-	1	-	1	-	3	3
CO5	3	3	3	3	2	-	-	-	1	-	1	3	3	3

1 – Slightly; 2 - Moderately; 3 – Substantially



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

PE -II - 1 - CLOUD COMPUTING**Credits: 3****Assessment: 40+ 60**

Semester	L-T-P	Course Code	Branches
VI	3-0-0	7-6-6-R25E21	CS

Prerequisites: Programming Knowledge, Computer Networks, Operating Systems, Database Fundamentals, Web Technologies.

Course Educational Objectives (CEOs):

The course is designed to

1. Understand fundamental concepts of cloud computing and its services.
2. Demonstrate an understanding of Service models, deployment models and Virtualization
3. Understand the programming and implementation issues of Cloud

UNIT I

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

Cloud Computing Fundamentals: Motivation for Cloud Computing, Defining Cloud Computing, 5-4-3 Principles of Cloud computing, Cloud Ecosystem, Requirements for Cloud Services, Cloud Application, Benefits and Drawbacks.

Cloud Computing Architecture and Management: Introduction, Cloud Architecture, Anatomy of the Cloud, Network Connectivity in Cloud Computing, Applications on the Cloud, Managing the Cloud, Migrating Application to Cloud.

Cloud Deployment Models: Introduction, Private Cloud, Public Cloud, Community Cloud, Hybrid Cloud.

UNIT II

Cloud Service Models: Introduction, Infrastructure as a Service, Platform as a Service, Software as a Service, Other Cloud Service Models.

Technological Drivers for Cloud Computing: Introduction, SOA and Cloud, Virtualization, Multicore Technology, Memory and Storage Technologies, Networking Technologies, Web 2.0, Web 3.0, Software Process Models for Cloud, Programming Models, Pervasive Computing, Operating System, Application Environment.

UNIT III

Virtualization: Introduction, Virtualization Opportunities, Approaches to Virtualization, Hypervisors, From Virtualization to Cloud Computing.

Programming Models for Cloud Computing: Introduction, Extended Programming Models for Cloud, New Programming Models Proposed for Cloud,

Software Development in Cloud: Introduction, Different Perspectives on SaaS Development, New Challenges, Cloud-Aware Software Development Using PaaS Technology.



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

UNIT IV

Networking for Cloud Computing: Introduction, Overview of Data Center Environment, Networking Issues in Data Centers, Transport Layer Issues in DCNs, TCP Enhancements for DCNs.

Cloud Service Providers: Introduction, EMC, Google, Amazon Web Services, Microsoft, IBM, SAP Labs, Salesforce, Rackspace, VMware, Manjrasoft.

UNIT V

Open Source Support for Cloud: Introduction, Open Source Tools for IaaS, Open Source Tools for PaaS, Open Source Tools for SaaS, Open Source Tools for Research, Distributed Computing Tools for Management of Distributed Systems.

Security in Cloud Computing: Introduction, Security Aspects, Platform-Related Security Audit and Compliance.

Advanced Concepts in Cloud Computing: Intercloud, Cloud Management, Mobile Cloud, Media Cloud, Interoperability and Standards, Cloud Governance, Computational Intelligence in Cloud, Green Cloud, Cloud Analytics.

Text Books:

1. K. Chandrasekaran, Essentials of Cloud Computing, CRC press, 2014.

Reference Books:

2. Buyya R, Vecchiola C, Selvi S T, Mastering Cloud Computing, McGraw Hill Education (India), 2013.
3. Dan C. Marinescu, Cloud Computing - Theory and Practice, Morgan Kaufmann (an imprint of Elsevier), 2013.
4. Rajkumar Buyya, James Broberg, Andrzej Goscinski, Cloud Computing - Principles and Paradigms, Wiley India, 2011.
5. Kai Hwang, Geoffrey C Fox, Jack J Dongarra, Distributed and Cloud Computing - From Parallel Processing to the Internet of Things, Morgan Kaufmann (an imprint of Elsevier), 2012.
6. Tim Mather, SubraKumaraswamy, ShahedLatif, Cloud Security and Privacy: An Enterprise Perspective on Risks and Compliance, O'Reilly.
7. Rittinghouse J W, Ransome J F, Cloud Computing - Implementation, Management, and Security, CRC Press, 2010.
8. Velte A T, Velte T J, Cloud Computing - A Practical Approach, McGraw Hill, 2011.
9. Shroff G, Enterprise Cloud Computing - Technology, Architecture, Applications, Cambridge University Press, 2010.
10. Antonopoulos N, Gillam L, Cloud Computing - Principles, Systems and Applications, Springer, 2010.
11. Furht B, Escalante A, Handbook of Cloud Computing, Springer, 2010.



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

Course Outcomes(COs):

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

- 1) Develop the ability to understand the basics of cloud computing and different cloud deployment models.
- 2) Handle resource management and security issues of cloud computing.
- 3) Evaluate the performance of different cloud service models.
- 4) Understand the programming and implementation issues of Cloud
- 5) Identify, analyze and use different cloud services/applications/tools available from key cloud providers.

Mapping of Course Outcomes with Program Outcomes:														
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	3	3	3	3	-	-	-	-	-	1	3	3	3
CO2	3	3	3	3	3	-	-	-	-	-	-	-	3	3
CO3	3	3	2	3	3	-	-	-	-	-	-	-	3	3
CO4	3	2	2	3	3	-	-	-	-	-	-	-	3	3
CO5	3	3	3	3	2	1	1	1	1	-	1	3	1	3

1 – Slightly; 2 - Moderately; 3 – Substantially



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

PE -II- 2 - COMPUTER GRAPHICS**Credits: 3****Assessment: 40+ 60**

Semester	L-T-P	Course Code	Branches
VI	3-0-0	7-6-6-R25E22	CS

Prerequisites: Programming Skills, Mathematics Foundations, Data Structures.**Course Educational Objectives (CEOs):**

The course is designed to

1. Understand the basics of various inputs and output computer graphics hardware devices.
2. Exploration of fundamental concepts in 2D and 3D computer graphics.
3. Learn 2D raster graphics techniques, 3D modelling, geometric transformations, 3D viewing and rendering.

UNIT I

Introduction- Image processing as picture analysis, Advantages of Interactive Graphics, Representative uses of computer graphics, Classification of applications, Development of hardware and software for computer graphics, Conceptual framework for Interactive Graphics.

Scan Converting Lines – Basic Incremental algorithm, Midpoint Line algorithm and additional issues; Scan Converting Circles, Scan Converting Ellipses, Solid Filling– Rectangles, Polygons and Ellipse arcs; Pattern filling, Thick primitives, Cohen-Sutherland line clipping algorithm, Parametric line clipping algorithms, Sutherland-Hodgeman polygon clipping algorithm, Generating characters and Antialiasing.

UNIT II

Display Systems - Raster-scan and Random scan.

Geometrical transformations – 2D transformations, Homogeneous coordinates, Matrix representation of 2D transformations, Composition of 2D transformations, Window to view-port transformation, Matrix representation of 3D transformations, Composition of 3D transformations and Transformation as a change in coordinate system.

Representing Curves and surfaces – Polygon meshes, Parametric cubic curves, Parametric bicubic surfaces and Quadric surfaces.

Fractals – Lines and Surfaces.

UNIT III

Viewing in 3D - Projections, Specifying an arbitrary 3D view, Examples of 3D viewing, Mathematics of planar geometric projections, Implementing planar geometric projections, Coordinate systems.

Solid Modeling – Representing solids, Regularized Boolean set operations, Primitive instancing, Sweep representations, Boundary representations, Spatial-Partitioning Representations, Constructive solid geometry, Comparison of representations, User interfaces for solid modelling.



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

UNIT IV

Achromatic and Colored Light – Achromatic light, Chromatic color, Color models for raster graphics, Reproducing color, Using color in computer graphics.

Visible Surface Determination – Functions of two variables, Techniques for efficient visible surface algorithms, z-Buffer algorithm, Scan-line algorithms, Visible surface ray tracing.

UNIT V

Illumination Models - Ambient light, Diffuse reflection, Atmospheric attenuation.

Shading Models – Constant shading, Interpolated shading, Polygon mesh shading, Gouraud shading, Phong shading, Problems with interpolated shading.

Surface Detail – Surface-detail polygons, Texture mapping, Bump mapping.

Animation – Conventional and Computer-Assisted animation, Animation languages, Methods of controlling animation, Basic rules of animation, Problems peculiar to animation.

Text Books:

1. Hughes J F, Van Dam A, Foley J D, et al., Computer Graphics: Principles and Practice, 3rd edition, Addison-Wesley, 2013.

Reference Books:

2. Foley J D, Van Dam A, Feiner S K, John F H, Computer Graphics: Principles & Practice in C, 2nd edition, Pearson Education, 1995.
3. Ragiv Chopra, Computer Graphics, S. Chand & Company, 2012.

Course Outcomes (COs):

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

1. Understand the various computer graphics hardware and display technologies.
2. Implement various 2D and 3D objects transformation techniques.
3. Exploration of fundamental concepts in 2D and 3D computer graphics.
4. Learn 2D raster graphics techniques, 3D modelling, geometric transformations, 3D viewing and rendering.
5. Apply 2D and 3D viewing technologies into the real world applications.

Mapping of Course Outcomes with Program Outcomes:														
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	2	3	3	3	-	-	-	1	2	1	3	3	3
CO2	3	3	3	3	3	-	-	-	-	-	1	3	3	3
CO3	3	3	2	3	2	-	-	-	-	-	-	-	3	3
CO4	3	2	2	3	2	-	-	-	1	2	-	-	3	3
CO5	3	3	3	3	2	-	-	-	1	2	1	3	3	3

1 – Slightly; 2 - Moderately; 3 – Substantially


Chairperson
(BoS UG)


DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

PE -II- 3 - IMAGE PROCESSING**Credits: 3****Assessment: 40+ 60**

Semester	L-T-P	Course Code	Branches
VI	3-0-0	7-6-6-R25E23	CS

Prerequisites: Programming Skills, Mathematics Foundations, Data Structures, AI.**Course Educational Objectives (CEOs):**

The course is designed to

1. Understand the fundamentals of Digital imaging and Image Processing techniques.
2. Discuss the concepts of image compression and segmentation.
3. Evaluate the performance of image processing algorithms and systems.

UNIT I

Introduction: Fundamentals of Image Processing, Applications of Image Processing, Human Visual Perception, Introduction to Image Formation, Sampling and Quantization, Binary Image, Three-Dimensional Imaging, Image file formats. Color and Color Imagery: Perception of Colors.

UNIT II

Image Transformation: Fourier Transforms, Discrete Cosine Transform, Walsh-adamard Transform, Karhaunen-Loeve Transform or PCA. Discrete Wavelet Transform: Wavelet Transform, Extension to 2D Signals, Lifting Implementation of the Discrete Wave Transforms.

UNIT III

Image Enhancement and Restoration : Introduction, Distinction between image enhancement and restoration, Histrogram-based Contrast Enhancement, Frequency Domain Methods of Image Enhancement, Noise Modeling, Image Restoration, Image Reconstruction, Image Segmentation.

UNIT IV

Recognition of Image Patterns : Introduction, Decision Theoretic Pattern Classification, Baesian Decision Theory, Nonparametric Classification, Linear Discriminant Analysis, Unsupervised Classification Strategies-clustering, K-means clustering algorithm, Syntactic Pattern Classification, Syntactic Inference, Symbolic Projection method. Texture and Shape Analysis.

UNIT V

Fuzzy Set Theory in Image Processing : Introduction, Use of Fuzzy Image, Preliminaries and Background, Image as a Fuzzy Set, Fuzzy Methods of Contrast Enhancement, Image Segmentation using Fuzzy Methods, Fuzzy Approaches to Pixel Classification, Fuzzy c-Means Algorithm, Fusion of Fuzzy logic with neural network. Image mining and Content-Based Retrieval.



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

Text Books:

1. Maria Petrou and Costas Petrou , “Image Processing the Fundamentals”, John-Wiley and Sons Publishers, 2nd edition, 2010.
2. Rafael C. Gonzalez , Richard E. Woods, Steven L. Eddins, "Digital Image Processing Using MATLAB", 2nd edition, Gatesmark Publishing, 2009.
3. Tinku Acharya and Ajoy K. Ray, “Image Processing Principles and Applications”, John Wiley & Sons publishers, 2005.

Reference Books:

4. Rafael Gonzalez and Richard E. Woods, Digital Image Processing, 4th edition, Pearson, 2017.
5. Anil K Jain, Fundamentals of Digital Images Processing, First edition, Pearson, 2015.

Course Outcomes(COs):

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

1. Understand Image representation and modeling.
2. Design and apply image enhancement and restoration techniques
3. Discuss the concepts of image compression and segmentation.
4. Evaluate the performance of image processing algorithms and systems
5. Develop image processing techniques for assisting digital forensics

Mapping of Course Outcomes with Program Outcomes:														
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	2	3	3	3	-	-	-	-	-	1	3	2	3
CO2	3	2	3	3	3	-	-	-	-	2	1	3	2	3
CO3	3	2	2	3	1	-	-	-	-	2	1	3	2	3
CO4	3	2	2	3	1	-	-	-	-	1	1	3	2	3
CO5	3	3	3	3	2	1	1	1	1	1	1	3	2	3

1 – Slightly; 2 - Moderately; 3 – Substantially


Chairperson
(BoS UG)


DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

PE-II - 4 - INTRODUCTION TO SEMICONDUCTOR MANUFACTURING**Credits: 3****Assessment: 40 + 60**

Semester	L-T-P	Course Code (7-0-1-01R25)	Branches
VI	3-0-0	7-0-6-R25CE1	CH,CE,EE,EC,ME,CS (All Branches)

Course Overview:

This course deals with semiconductor materials, semiconductor wafer manufacturing and characterization, semiconductor device manufacturing techniques, characterization and advanced processing tools, semiconductor ecosystem and applications.

This course prepares the students to apply their knowledge for designing the processes and manufacturing the equipment and tools for semiconductor device manufacturing in India under the National Semiconductor Mission Programme.

Course Objective(s):

- To explain the physics of semiconductor materials and devices.
- To discuss the processes involved in semiconductor manufacturing.
- To understand the ecosystem of semiconductor manufacturing.
- To get knowledge of safety aspects in the semiconductor industry.
- To design systems and equipment for semiconductor manufacturing.

UNIT – 1:**Semiconductor materials (Si, Ge, GaAs, GaN, SiC):**

Electrical classification of solids (metals, semiconductors and insulators), intrinsic and extrinsic semiconductors, crystal structure and defects, band gap and Fermi level, doping effects on electrical conduction in semiconductors, direct and indirect bandgap semiconductors.

UNIT – 2:**Semiconductor wafer manufacturing and characterization:**

Thermal and chemical processes involved in converting pure sand into electronic grade polycrystalline silicon, pure silicon crystal growth from polysilicon, silicon wafer fabrication and polishing.

UNIT – 3:**Thin film fabrication and characterization:**

Vacuum-based thin film deposition techniques for metals, semiconductors and dielectrics; Physical Vapour Deposition (PVD); Chemical Vapour Deposition (CVD); analytical characterization techniques (XRD, SEM, TEM, XPS etc.); plasma, lasers and ion beams for semiconductor manufacturing.



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

UNIT – 4:**Semiconductor Device Manufacturing Techniques:**

Introduction to IC-chip manufacturing, oxidation, lithography, etching, doping, diffusion, ion-implantation and metallization, realization of on-chip resistor, diode, capacitor and inductor, bonding and packaging.

UNIT – 5:**Semiconductor ecosystem, safety and applications:**

Clean room aspects, gas handling systems in semiconductor manufacturing, safety aspects in device fabrication (electrical, chemical and vacuum), clean water and pure chemical requirements, CMOS-based IC chips for computers and mobiles, semiconductor-based solar cells and displays, lab-grown diamonds for semiconductor applications.

Textbooks:

1. Sze SM and Lee MK, *Semiconductor Devices: Physics and Technology*, 3rd Edition, Wiley India (2016).
2. Swami Nathan P, *Semiconductor Materials, Devices and Fabrication*, Wiley (2019)

References:

1. Neaman D, *Semiconductor Physics and Devices*, 4th Edition, McGraw Hill (2017).
2. MKS Handbook: *Semiconductor Devices and Process Technology*, 2nd Edition (2017).
3. Gandhi S.K, *VLSI Fabrication Principles: Silicon and Ga-As*, 2nd Edition, Wiley (2008).

Course Outcomes: After completing this course, students will be able to

1. Understand the basic properties and behavior of semiconductor materials used in electronic devices.
2. Learn how semiconductor wafers are manufactured from raw materials to finished silicon wafers.
3. Gain knowledge of thin-film deposition methods and basic material characterization techniques.
4. Understand the main steps involved in semiconductor device and IC fabrication processes.
5. Become familiar with the semiconductor manufacturing ecosystem, safety practices, and major applications of semiconductor devices.

Mapping of course outcomes with program outcomes:

PO/ CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	2	-	-	-	-	-	-	-	-	-	1	3	2
CO2	3	2	-	2	-	-	-	-	-	-	-	1	3	1
CO3	2	2	-	3	3	-	-	-	-	-	-	1	2	2
CO4	2	2	3	2	2	-	-	-	-	-	-	1	1	1
CO5	1	-	-	-	-	3	2	2	2	1	-	2	2	2


Chairperson
(BoS UG)


DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

B. Tech. (Computer Science and Engineering)
Semester VII (Fourth Year)

Code		Category	Course Title	Hours/Week			Total	Credits	Marks		
Department	University			L	T	P			CIE	SEE	Total
CS6701C	7-6-7-01R25	CS-PC	Machine Learning	3	0	0	3	3	40	60	100
CS6702E	7-6-7-R25E1	CS-PE	Professional Elective–III	3	0	0	3	3	40	60	100
CS6703E	7-6-7-R25E2	CS-PE	Professional Elective–IV	3	0	0	3	3	40	60	100
CS6704E	7-6-7-R25E3	CS-PE	Professional Elective–V	3	0	0	3	3	40	60	100
CS6705O	7-6-7-R25O1	CS-OE	Open Elective -III (MOOCS)	0	0	0	0	3	100	0	100
CS6706O	7-6-7-R25O2	CS-OE	Open Elective -IV (MOOCS)	0	0	0	0	3	100	0	100
CS6707L	7-6-7-R25P1	CS-PC	Machine Learning Lab	0	0	2	2	1	40	60	100
CS6708S	7-6-7-R25S1	CS-SC	Mobile Application Development	1	0	2	3	2	100	0	100
CS6709K	7-6-7-R25KS	CS-IK	IKS – MOOCS	0	0	0	0	2	100	0	100
CS6710I	7-0-7R25CS	CS-SI	Summer Industrial Internship	0	0	0	0	2	0	100	100
				13	0	4	17	25	600	400	1000
Students shall ensure that IKS MOOKS and IKS Mandatory courses are not same either in title and/or in syllabi.											



Chairperson
(BoS UG)


 DEAN
 FACULTY OF ENGINEERING
 S. V. U. COLLEGE OF ENGINEERING
 TIRUPATI-517 502

COMPUTER SCIENCE & ENGINEERING**PROFESSIONAL ELECTIVE COURSES**

Professional Elective - I: (VI Sem)	7-6-6-R25E11	Object Oriented Analysis and Design
	7-6-6-R25E12	Soft Computing
	7-6-6-R25E13	Data mining
Professional Elective – II (VI Sem)	7-6-6-R25E21	Cloud Computing
	7-6-6-R25E22	Computer Graphics
	7-6-6-R25E23	Image Processing
	7-0-6R25CE1	Introduction to Semiconductor Manufacturing
Professional Elective – III (VII Sem)	7-6-7-R25E11	Cyber Security
	7-6-7-R25E12	Internet of Things
	7-6-7-R25E13	Quantum Technologies and Applications
Professional Elective – IV (VII Sem)	7-6-7-R25E21	Deep Learning
	7-6-7-R25E22	Block Chain Technologies
	7-6-7-R25E23	Fog and Edge Computing
Professional Elective – V (VII Sem)	7-6-7-R25E31	Big Data Analytics
	7-6-7-R25E32	Natural Language Processing
	7-6-7-R25E33	DevOps and Agile Methodologies.


Chairperson
(BoS UG)


DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

MACHINE LEARNING**Credits: 3****Assessment: 40+ 60**

Semester	L-T-P	Course Code	Branches
VII	3-0-0	7-6-7-01R25	CS

Prerequisite: Artificial Intelligence, Mathematics, Python Programming, Basic Algorithms.

Course Educational Objectives (CEOs):

The course is designed to

1. Understand the basic concepts, techniques and algorithms used in machine learning.
2. Formulate machine learning problems corresponding to different applications.
3. Develop skills of using recent machine learning software for solving practical problems.
4. Evaluate performance and to identify the limitations of machine learning algorithms.

UNIT I

Introduction to Machine Learning: Well-Posed Learning Problems, Designing a Learning System, Perspectives and Issues in Machine learning.

Concept Learning: Introduction, A Concept Learning Task, Concept Learning as Search, Find-S: Finding a Maximally Specific Hypothesis, Version Spaces and the Candidate Elimination Algorithm, Remarks on Version Spaces and Candidate Elimination, Inductive Bias.

Decision Tree Learning: Introduction, Decision Tree Representation. Appropriate Problems for Decision Tree Learning, The Basic Decision Tree Algorithm, Hypothesis Space Search in Decision Tree Learning, Inductive Bias in Decision Tree Learning, Issues in Decision Tree Learning.

UNIT II

Artificial Neural Networks: Introduction, Neural Networks Representations, Appropriate Problems for Neural Network Learning, Perceptrons, Multilayer Networks and the Back propagation Algorithm, Remarks on the Back propagation Algorithm, An Illustrative Example: Face Recognition, Advanced Topics in Artificial Neural Networks.

Evaluating Hypothesis: Motivation, Estimating Hypothesis Accuracy, Basics of Sampling Theory, A General Approach for Deriving Confidence Intervals, Difference in Error of Two Hypothesis, Comparing Learning Algorithms.



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

UNIT III

Bayesian Learning: Introduction, Bayes Theorem, Concept Learning, Maximum Likelihood and Least-Squared Error Hypothesis, Maximum Likelihood Hypothesis for Predicting Probabilities, Minimum Description Length Principle, Bayes Optimal Classifier, Gibbs Algorithm, Naïve Bayes Classifier, An Example Learning to Classify Test, Bayesian Belief Networks, The EM Algorithm.

Computational Learning Theory: Introduction, Probably Learning an Approximately Correct Hypothesis, Sample Complexity for Finite Hypothesis Spaces, Sample Complexity for Infinite Hypothesis Spaces, The Mistake Bound Model of Learning.

UNIT IV

Instance-Based Learning: Introduction, k-Nearest Neighbor Learning, Locally Weighted Regression, Radial Basis Functions, Case-Based Reasoning, Remarks on Lazy and Eager Learning.

Genetic Algorithms: Motivation, Genetic Algorithms, An Illustrative Example, Hypothesis Space Search, Genetic Programming, Models of Evaluation and Learning, Parallelizing Genetic Algorithms. Learning Sets of Rules: Introduction, Sequential Covering Algorithms, Learning Rule Sets: Summary, Learning First-Order Rules, Learning Sets of First-Order Rules: FOIL, Induction as Inverted Deduction, Inverting Resolution.

UNIT V

Remarks on Explanation-Based Learning, Explanation-Based Learning of Search Control Knowledge.

Combining Inductive and Analytical Learning: Motivation, Inductive-Analytical Approaches to Learning, Using Prior Knowledge to Initialize the Hypothesis, Using Prior Knowledge to Alter the Search Objective, Using Prior Knowledge to Augment Search Operators, State of the Art.

Reinforcement Learning: Introduction, The Learning Task, Q learning, Non-deterministic Rewards and Actions, Temporal Difference Learning, Generalizing from Examples, Relationship to Dynamic Programming.

Text Books:

1. Tom M. Mitchell, Machine Learning, McGraw Hill Education, 2018.



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

Reference Books:

2. Stephen Marsland, Taylor & Francis, Machine Learning: An Algorithmic Perspective, 2nd Edition, CRC Press, 2015.
3. Ethem Alpaydin, Introduction to Machine Learning, 2nd Edition, MIT Press, 2010.
4. Stuart J. Russell and Peter Norvig, Artificial Intelligence A Modern Approach: 3rd Edition, Pearson Education, 2010.
5. M.Tim Jones, Artificial Intelligence: A Systems Approach, 3rd Edition, Infinity Science Press, New Delhi, 2008.
6. Elaine Ritch, Kevin Knight, Shivashankar B Nair, Artificial Intelligence, 3rd Edition, Tata McGraw Hill Education, 2009.

Course Outcomes (COs):

1. Apply fundamental machine learning concepts to formulate well-posed learning problems and analyze concept learning and decision tree learning algorithms with appropriate inductive bias.
2. Design and evaluate Artificial Neural Network models, including perceptron and multilayer networks, and assess learning performance using statistical hypothesis evaluation techniques.
3. Analyze and implement Bayesian learning methods and computational learning theory concepts to estimate hypothesis reliability, sample complexity, and learning guarantees.
4. Compare and apply instance-based learning, Genetic algorithms, and Rule-based learning techniques to solve complex learning problems using appropriate hypothesis space search strategies.
5. Integrate inductive, analytical, and Reinforcement learning approaches to develop adaptive learning systems capable of decision-making in dynamic environments.

Mapping of Course Outcomes with Program Outcomes:															
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	
CO1	3	2	3	3	3	-	-	-	1	2	1	3	3	3	
CO2	2	3	3	3	3	-	-	-	1	2	1	2	3	3	
CO3	3	3	2	3	1	-	-	-	1	2	1	2	3	3	
CO4	3	2	2	3	1	-	-	-	1	2	1	3	2	3	
CO5	3	3	3	3	2	-	-	-	1	2	1	3	3	3	

1 – Slightly; 2 - Moderately; 3 – Substantially



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

Machine Learning Lab**Credits: 1****Assessment: 40+ 60**

Semester	L-T-P	Course Code	Branches
VII	0-0-2	7-6-7-R25P1	CS

PREREQUISITES: A course on “Machine Learning”**Course Educational Objectives – Machine Learning Lab****1. Practical Implementation Skills**

Enable students to implement core machine learning algorithms such as Decision Trees, Neural Networks, Bayesian models, Instance-based learning, and Reinforcement Learning using Python.

2. Problem-Solving Ability

Develop the ability to analyze real-world problems and select appropriate machine learning techniques to design effective solutions.

3. Experimental Analysis Competence

Train students to evaluate hypotheses, measure model performance, compare algorithms, and interpret experimental results scientifically.

4. Algorithmic Understanding

Strengthen conceptual understanding of learning theories such as inductive bias, hypothesis spaces, Bayesian reasoning, and computational learning theory through experimentation.

5. Model Design & Optimization Skills

Equip students to design, train, tune, and optimize machine learning models using suitable parameters, evaluation methods, and validation techniques.

6. Research and Innovation Orientation

Encourage students to explore advanced topics such as genetic algorithms, reinforcement learning, and hybrid learning approaches for innovative applications.

7. Professional Readiness

Prepare students with hands-on experience in modern ML tools and workflows required for industry, research, and higher studies.

Experiments:**UNIT I – Concept Learning & Decision Trees****Experiment 1: Designing a Simple Learning System**

Implement a basic ML pipeline: dataset loading, preprocessing, training, testing, and evaluation.

Experiment 2: Find-S Algorithm Implementation

Write a Python program to implement Find-S and test it on a concept learning dataset.

Experiment 3: Candidate Elimination Algorithm

Implement Version Space approach and display S and G boundaries after each training example.



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

Experiment 4: Decision Tree Classifier

Build a decision tree using Scikit-learn and visualize it. Compare entropy vs gini index.

UNIT II – Neural Networks & Hypothesis Evaluation

Experiment 5: Perceptron Algorithm

Implement perceptron from scratch and test it on linearly separable data.

Experiment 6: Multilayer Neural Network with Backpropagation

Train a neural network using TensorFlow/Keras or Scikit-learn MLPClassifier.

Experiment 7: Neural Network for Image Classification

Train a simple neural network on digit dataset and evaluate accuracy.

Experiment 8: Hypothesis Evaluation Metrics

Compute accuracy, precision, recall, F1 score, and confusion matrix for a classifier.

Experiment 9: Confidence Interval Estimation

Estimate confidence interval of model accuracy using sampling techniques.

UNIT III – Bayesian Learning & Computational Learning Theory

Experiment 10: Naïve Bayes Classifier

Implement Gaussian Naïve Bayes and test it on a real dataset.

Experiment 11: Bayesian Belief Network (Basic)

Create a simple probabilistic graphical model using a Python library and perform inference.

Experiment 12: EM Algorithm Implementation

Implement Expectation-Maximization for Gaussian Mixture clustering.

Experiment 13: Sample Complexity Simulation

Simulate effect of dataset size on model accuracy and plot learning curves.

UNIT IV – Instance-Based Learning & Genetic Algorithms

Experiment 14: k-Nearest Neighbor Algorithm

Implement KNN from scratch and compare with Scikit-learn version.

Experiment 15: Locally Weighted Regression

Implement regression using distance-based weighting.



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

Experiment 16: Radial Basis Function Network

Implement RBF network and compare with linear regression.

Experiment 17: Genetic Algorithm Optimization

Implement GA to optimize a mathematical function.

Experiment 18: Rule Learning Algorithm

Implement a sequential covering algorithm to generate IF-THEN rules.

UNIT V – Advanced Learning & Reinforcement Learning

Experiment 19: Explanation-Based Learning (Concept Demo)

Simulate rule extraction from trained model predictions.

Experiment 20: Using Prior Knowledge in Learning

Initialize model parameters manually and compare convergence speed.

Experiment 21: Q-Learning Implementation

Implement Q-learning for grid world environment.

Experiment 22: Temporal Difference Learning

Implement TD(0) prediction algorithm.

Experiment 23: Reinforcement Learning vs Dynamic Programming

Compare value iteration and Q-learning performance.

Mini Project (Optional)

Students may design and implement one end-to-end ML system integrating at least two units' concepts.

Examples: Intelligent Recommendation System, Fraud Detection System, Medical Diagnosis Predictor, Autonomous Agent Simulation.

Couesre Outcomes(COs):

- 1) Understand and explain fundamental machine learning algorithms and concepts
- 2) Apply preprocessing, visualization, and feature engineering techniques on datasets.
- 3) Implement supervised, unsupervised, probabilistic, and reinforcement learning algorithms using Python.
- 4) Analyze performance of ML models using appropriate evaluation metrics and validation techniques
- 5) Compare different learning algorithms and interpret results for selecting optimal models.



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

Appropriate datasets from the following repository can be utilized:

1. <https://www.kaggle.com/datasets>
2. <http://sci2s.ugr.es/keel/datasets.php#sub1>

Mapping of Course Outcomes with Program Outcomes:														
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	2	1	3	3	-	-	-	1	-	-	2	3	3
CO2	2	3	2	3	3	-	-	-	-	-	-	2	3	3
CO3	3	3	3	3	2	-	-	-	1	-	-	2	3	3
CO4	2	3	2	3	1	-	-	-	-	-	-	2	3	3
CO5	3	3	3	3	2	1	1	1	1	2	2	2	3	3



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

MOBILE APPLICATION DEVELOPMENT**Credits: 2****Assessment: 100+0**

Semester	L-T-P	Course Code	Branches
VII	1-0-2	7-6-7-R25S1	CS

Prerequisite: Object Oriented Programming, DS, Databases, APIs, Basic UI design.**Course Educational Objectives (CEOs):**

The course is designed to

1. Impart both conceptual and practical knowledge on Android OS.
2. Understand and develop robust applications for Mobile devices on Android platform.

The course shall cover the following topics:**Introduction to Android Programming:** What is Android?, Obtaining the Required Tools, Launching Your First Android Application.**Using Android Studio for Android Development:** Exploring the IDE, Using Code Completion, Debugging Your Application, Publishing Your Application.**Activities, Fragments, and Intents:** Understanding Activities, Linking Activities Using Intents, Fragments, Displaying Notifications.**Introduction to Android User Interface:** Understanding the Components of a Screen, Adapting to Display Orientation, Managing Changes to Screen Orientation, Utilizing the Action Bar, Creating the User Interface Programmatically, and Listening for UI Notifications.**Designing User Interface with Views:** Using Basic Views, Using Picker Views, Using List Views to Display Long Lists, Understanding Specialized Fragments.**Displaying Pictures and Menus with Views:** Using Image Views to Display Pictures, Using Menus with Views, Using WebView.**Data Persistence:** Saving and Loading User Preferences, Persisting Data to Files, Creating and Using Databases.**Content Providers:** Sharing Data in Android, Using a Content Provider, Creating Your Own Content Providers, Using the Content Provider.**Messaging:** SMS Messaging, Sending Email.**Location-Based Services:** Displaying Maps, Getting Location Data, Monitoring a Location.**Networking:** Consuming Web Services Using HTTP, Consuming JSON Services.**Developing Android Services:** Creating Your Own Services, Establishing Communication Between a Service and an Activity, Binding Activities to Services, Understanding Threading.**Publishing Android Applications:** Preparing for Publishing, Deploying APK Files.

Chairperson
(BoS UG)

 DEAN
 FACULTY OF ENGINEERING
 S. V. U. COLLEGE OF ENGINEERING
 TIRUPATI-517 502

Text Books:

1. J. F. DiMarzio, Beginning Android Programming with Android Studio, John Wiley & Sons, Inc., Fourth Edition, 2017.
2. Wei-Meng Lee, Beginning Android™ 4 Application Development, John Wiley & Sons, Inc., 2012.

Reference Books:

3. Ian Darwin, Android Cookbook: Problems and Solutions for Android Developers, O'Reilly, Second Edition, 2017.
4. Bill Phillips, Chris Stewart and Kristin Marsicano, Android Programming: The Big Nerd Ranch Guide, Big Nerd Ranch, Third Edition, 2017.
5. Wei-Meng Lee, Android™ Application Development Cookbook: 93 Recipes for Building Winning Apps, John Wiley & Sons, Inc., 2013.
6. Peter Späth, Pro Android with Kotlin: Developing Modern Mobile Apps, Apress, 2018.
7. Neil Smyth, Android Studio 3.0 Development Essentials– Android 8 Edition, Neil Smyth / Payload Media, Inc., 2017.

Course Outcomes(COs):

Having successfully completed this course the students will be able to:

1. Demonstrate their skills of using Android software development tools
2. Demonstrate knowledge on mobile platforms, mobile user interface and user interface design requirements.
3. Develop mobile applications and publish in different mobile platforms.
4. Impart both conceptual and practical knowledge on Android OS.
5. Understand and develop robust applications for Mobile devices on Android platform.

Mapping of Course Outcomes with Program Outcomes:														
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	2	3	3	3	-	-	-	1	-	-	3	1	3
CO2	3	3	3	3	3	-	-	-	-	-	-	3	2	3
CO3	3	3	2	3	1	-	-	-	-	2	-	3	2	3
CO4	3	2	2	3	1	-	-	-	-	2	-	3	2	3
CO5	3	3	3	3	2	-	-	-	-	2	1	3	1	3

1 – Slightly; 2 - Moderately; 3 – Substantially



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

PE -III- 1 - CYBER SECURITY**Credits: 3****Assessment: 40+ 60**

Semester	L-T-P	Course Code	Branches
VII	3-0-0	7-6-7-R25E11	CS

Prerequisites: Networking , Operating Systems, Programming basics, web fundamentals.

Course Educational Objectives (CEOs):

The course is designed to

1. Analyze and evaluate the cyber security needs of an organization.
2. Design and develop security architecture for an organization.
3. Develop cyber security strategies and policies.
4. Determine and analyze software vulnerabilities and develop necessary security solutions.

UNIT I

Building a Secure Organization, Preventing System Intrusions, Guarding Against Network Intrusions, Internet Security, The Botnet Problem, Intranet Security, Local Area Network Security.

UNIT II

Wireless Network Security, Cellular Network Security, RFID Security, Protecting Mission-Critical Systems, Security Management Systems, Information Technology Security Management, Identity Management.

UNIT III

Intrusion Prevention and Detection Systems, Computer Forensics, Network Forensics, Firewalls, Penetration Testing, Vulnerability Assessment.

UNIT IV

NET Privacy, Personal Privacy Policies, Virtual Private Networks, Identity Theft, VoIP Security, SAN Security, SAN Devices Security.

UNIT V

Risk Management, Physical Security Essentials, Biometrics, Information Warfare, Security Through Diversity, Reputation Management, Content Filtering, Data Loss Protection.



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

Text Books:

1. Vacca J R (Editor), Computer and Information Security Handbook, 2nd edition, Elsevier / Morgan Kaufmann, 2013.
2. Belapure S, Godbole N, Cyber Security, Wiley, 2011.

Reference Books:

3. Gogolin G, Digital Forensics Explained, CRC / Auerbach, 2013.
4. Godbole N, Information Systems Security, Wiley, 2015.
5. Wu C H, Irwin J D, Introduction to Computer Networks and Cyber Security, CRC Press, 2013.
6. Singer P W, Friedman A, Cyber Security and Cyber War: What Everyone Needs to Know, Oxford University Press, 2014.
7. Boddington R, Practical Digital Forensics, Packt, 2016.
8. Drake J J, Lanier Z, et al., Android Hacker's Handbook, Wiley, 2014.
9. Graham J, Howard R, Olson R, Cyber Security Essentials, CRC Press, 2010.
10. Hadnagy C, Wilson P, Social Engineering: The Art of Human Hacking, Wiley, 2010

Course Outcomes (COs):

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

1. Effectively use cyber security and computer forensics software/tools
2. Measure the performance and troubleshoot cyber security systems.
3. Protect the network from both internal and external attacks
4. Provide new security solutions and implement the same confidently.
5. Develop cyber security strategies and policies

Mapping of Course Outcomes with Program Outcomes:														
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	3	3	3	3	-	-	-	-	3	1	3	2	3
CO2	3	3	3	3	3	-	-	-	-	3	1	3	2	3
CO3	3	3	2	3	1	-	-	-	-	2	1	3	3	3
CO4	3	3	2	3	1	-	-	-	-	2	1	3	3	3
CO5	3	3	3	3	2	-	-	-	-	2	1	3	2	3

1 – Slightly; 2 - Moderately; 3 – Substantially


Chairperson
(BoS UG)


DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

PE -III- 2 - INTERNET OF THINGS**Credits: 3****Assessment: 40+ 60**

Semester	L-T-P	Course Code	Branches
VII	3-0-0	7-6-7-R25E12	CS

Prerequisites: Programming , Basic electronics, Networking, Embedded Systems fundamentals.

Course Educational Objectives (CEOs):

1. Introduce the fundamental concepts, architecture, and evolution of the Internet of Things.
2. Provide knowledge of IoT devices, sensors, and embedded platforms.
3. Explain communication technologies and protocols used in IoT systems.
4. Familiarize students with IoT data processing, platforms, and security aspects.
5. Expose students to real-world IoT applications and future trends.

UNIT-I: Introduction to Internet of Things

Introduction to Internet of Things, Evolution of IoT, Characteristics of IoT, IoT Architecture (Three-layer and Five-layer Models), IoT Ecosystem, Enabling Technologies for IoT, Applications of IoT, Challenges and Issues in IoT.

UNIT-II: IoT Devices and Sensors

IoT Hardware Components, Sensors and Actuators, Embedded Systems for IoT, Microcontrollers and Microprocessors (Overview), IoT Development Boards (Arduino, Raspberry Pi – Overview), Interfacing Sensors and Actuators, Power Management in IoT Devices.

UNIT-III: IoT Communication and Networking:

IoT Communication Models, IoT Protocol Stack, Short-Range Communication Technologies (RFID, Bluetooth, ZigBee), Long-Range Communication Technologies (Wi-Fi, LoRa, NB-IoT – Overview), Application Layer Protocols (MQTT, CoAP, HTTP), Cloud Connectivity for IoT.

UNIT-IV: IoT Data Processing and Security

IoT Data Collection and Processing, Edge and Cloud Computing for IoT (Overview), Data Storage and Analytics, Introduction to IoT Platforms, Security Challenges in IoT, Authentication and Authorization, Privacy Issues, Secure IoT Architecture.



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

UNIT-V: IoT Applications and Case Studies

Smart Home and Smart City Applications, Industrial IoT, Smart Healthcare, Smart Agriculture, Wearable Devices, Case Studies of Real-World IoT Systems, IoT Standards, Future Trends and Research Directions in IoT.

TEXTBOOKS

1. Arshdeep Bahga, Vijay Madisetti, Internet of Things: A Hands-On Approach, Universities Press
2. Raj Kamal, Internet of Things: Architecture and Design Principles, McGraw-Hill.

REFERENCE BOOKS

3. Samuel Greengard, The Internet of Things, MIT Press.
4. Olivier Hersent, David Boswarthick, Omar Elloumi, The Internet of Things: Key Applications and Protocols, Wiley.
5. Perry Lea, Internet of Things for Architects, Packt Publishing.

COURSE OUTCOMES (COs):

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

1. Explain IoT concepts, architecture, and enabling technologies.
2. Identify and describe IoT devices, sensors, and embedded platforms.
3. Apply basic communication protocols and networking concepts for IoT systems.
4. CO4: Explain data processing, security, and privacy issues in IoT environments.
5. CO5: Analyze real-world IoT applications and emerging trends.

Mapping of Course Outcomes with Program Outcomes:														
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	3	1	2	3	-	-	-	-	2	1	3	2	3
CO2	2	3	2	3	3	-	-	-	-	2	2	2	2	3
CO3	3	3	3	2	1	-	-	-	-	3	2	2	2	3
CO4	2	3	3	3	1	-	-	-	-	3	1	3	2	3
CO5	3	3	2	3	2	-	-	-	-	2	1	3	2	3

1 – Slightly; 2 - Moderately; 3 – Substantially


Chairperson
(BoS UG)


DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

PE -III- 3 - QUANTUM TECHNOLOGIES AND APPLICATIONS**Credits: 3****Assessment: 40+ 60**

Semester	L-T-P	Course Code	Branches
VII	3-0-0	7-6-7-R25E13	CS

Prerequisites: Maths , Physics fundamentals, Quantum Mechanics Basics, Basic programming.**Course Educational Objectives (CEOs):**

1. Introduce fundamental quantum concepts like superposition and entanglement.
2. Understand theoretical structure of qubits and quantum information.
3. Explore conceptual challenges in building quantum computers.
4. Explain principles of quantum communication and computing.
5. Examine real-world applications and the future of quantum technologies.

UNIT-1

Introduction to Quantum Theory and Technologies:

The transition from classical to quantum physics, Fundamental principles explained conceptually: Superposition, Entanglement, Uncertainty Principle, Wave-particle duality, Classical vs Quantum mechanics – theoretical comparison, Quantum states and measurement: nature of observation, Overview of quantum systems: electrons, photons, atoms, The concept of quantization: discrete energy levels, Why quantum? Strategic, scientific, and technological significance, A snapshot of quantum technologies: Computing, Communication, and Sensing, National and global quantum missions: India's Quantum Mission, EU, USA, China

UNIT-II

Theoretical Structure of Quantum Information Systems:

What is a qubit? Conceptual understanding using spin and polarization, Comparison: classical bits vs quantum bits, Quantum systems: trapped ions, superconducting circuits, photons (non-engineering view), Quantum coherence and decoherence – intuitive explanation, Theoretical concepts: Hilbert spaces, quantum states, operators – only interpreted in abstract, The role of entanglement and non-locality in systems, Quantum information vs classical information: principles and differences, Philosophical implications: randomness, determinism, and observer role.

UNIT-III

Building a Quantum Computer – Theoretical Challenges and Requirements

What is required to build a quantum computer (conceptual overview)?, Fragility of quantum systems: decoherence, noise, and control, Conditions for a functional quantum system: Isolation, Error management, Scalability, Stability, Theoretical barriers:


Chairperson
(BoS UG)


DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

Why maintaining entanglement is difficult, Error correction as a theoretical necessity, Quantum hardware platforms (brief conceptual comparison), Superconducting circuits, Trapped ions, Photonics, Vision vs reality: what's working and what remains elusive, The role of quantum software in managing theoretical complexities.

UNIT-IV

Quantum Communication and Computing – Theoretical Perspective

Quantum vs Classical Information, Basics of Quantum Communication, Quantum Key Distribution (QKD), Role of Entanglement in Communication, The Idea of the Quantum Internet – Secure Global Networking, Introduction to Quantum Computing, Quantum Parallelism (Many States at Once), Classical vs Quantum Gates, Challenges: Decoherence and Error Correction, Real-World Importance and Future Potential

UNIT-V

Applications, Use Cases, and the Quantum Future

Real-world application domains: Healthcare (drug discovery), Material science, Logistics and optimization, Quantum sensing and precision timing, Industrial case studies: IBM, Google, Microsoft, PsiQuantum, Ethical, societal, and policy considerations, Challenges to adoption: cost, skills, standardization, Emerging careers in quantum: roles, skillsets, and preparation pathways, Educational and research landscape – India's opportunity in the global quantum race

Textbooks:

1. Michael A. Nielsen, Isaac L. Chuang, *Quantum Computation and Quantum Information*, Cambridge University Press, 10th Anniversary Edition, 2010.
2. Eleanor Rieffel and Wolfgang Polak, *Quantum Computing: A Gentle Introduction*, MIT Press, 2011.
3. Chris Bernhardt, *Quantum Computing for Everyone*, MIT Press, 2019.

Reference Books:

1. David McMahon, *Quantum Computing Explained*, Wiley, 2008.
2. Phillip Kaye, Raymond Laflamme, Michele Mosca, *An Introduction to Quantum Computing*, Oxford University Press, 2007.
3. Scott Aaronson, *Quantum Computing Since Democritus*, Cambridge University Press, 2013.
4. Alastair I.M. Rae, *Quantum Physics: A Beginner's Guide*, Newworld Publications, Revised Edition, 2005.
5. Eleanor G. Rieffel, Wolfgang H. Polak, *Quantum Computing: A Gentle Introduction*, MIT Press, 2011.
6. Leonard Susskind, Art Friedman, *Quantum Mechanics: The Theoretical Minimum*, Basic Books, 2014.



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

7. Bruce Rosenblum, Fred Kuttner, Quantum Enigma: Physics Encounters Consciousness, Oxford University Press, 2nd Edition, 2011.
8. Giuliano Benenti, Giulio Casati, Giuliano Strini, Principles of Quantum Computation and Information, Volume I: Basic Concepts, World Scientific Publishing, 2004.
9. K.B. Whaley et al., Quantum Technologies and Industrial Applications: European Roadmap and Strategy Document, Quantum Flagship, European Commission, 2020.
10. Department of Science & Technology (DST), Government of India, National Mission on Quantum Technologies & Applications – Official Reports and Whitepapers, MeitY/DST Publications, 2020 onward.

Online Learning Resources:

- [IBM Quantum Experience and Qiskit Tutorials](#)
- [Coursera – Quantum Mechanics and Quantum Computation by UC Berkeley](#)
- [edX – The Quantum Internet and Quantum Computers](#)
- [YouTube – Quantum Computing for the Determined by Michael Nielsen](#)
- Qiskit Textbook – IBM Quantum

Course Outcomes (COs):

1. Explain core quantum principles in a non-mathematical manner.
2. Compare classical and quantum information systems.
3. Identify theoretical issues in building quantum computers.
4. Discuss quantum communication and computing concepts.
5. Recognize applications, industry trends, and career paths in quantum technology.

Mapping of Course Outcomes with Program Outcomes:														
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	2	3	3	3	-	-	-	1	2	1	3	1	3
CO2	2	3	3	3	3	-	-	-	1	2	1	-	2	3
CO3	2	3	2	3	1	-	-	-	1	-	1	-	2	3
CO4	3	2	2	3	1	1	1	1	1	-	1	3	2	3
CO5	3	3	3	3	2	1	1	1	1	2	1	3	1	3

1 – Slightly; 2 - Moderately; 3 – Substantially



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

PE -IV- 1 -DEEP LEARNING**Credits: 3****Assessment: 40+ 60**

Semester	L-T-P	Course Code	Branches
VII	3-0-0	7-6-7-R25E21	CS

Prerequisite: Machine learning, Artificial Neural Networks and Knowledge on basic linear algebra**Course Educational Objectives (CEOs):**

1. To acquire knowledge about different mathematical tools for Deep Learning.
2. To understand fundamentals of artificial neural networks.
3. To Explore various optimizers and Regularization Techniques
4. To Learn about Convolutional Neural Networks.
5. To comprehend Object Detection, Auto encoder and GAN Architectures

UNIT-1**MATHEMATICAL TOOLS FOR DEEPLARNING**

Linear Algebra :Matrix, Vector, Transpose, Tensor, operations on elements, Systems of Linear equations, Rank, Norm, expressing a Matrix, Determinant, Trace, Eigen values and Eigen Vectors, Singular Value Decomposition(SVD).

Statistics – Probability, Random Variable, Binomial Distribution, Poisson Distribution, Normal Distribution, Sampling, Central limit Theorem.

Calculus- Derivatives, rules for derivatives, Partial derivatives.

UNIT-II**FUNDAMENTAL OF ARTIFICIAL NEURAL NETWORK(ANN)**

Understanding the Biological Neuron, Exploring the Artificial Neuron, Early Implementation of ANN – Rm Culloch-Pits model of Neuron, Rosenblatt's Perceptron, Types of Activation Functions-linear function, non-linear function, soft max function. Architectures of neural network-Single layer feed forward network, Multi-layer feed forward ANN, Recurrent Neural Network, Convolutional Networks. Learning Process in ANN – Weight of Interconnection between neurons, Gradient Descent and Back propagation.

UNIT-III**TRAINING DEEP NEURAL NETWORK**

Initializing Weights-He/Kaiming initialization, Xavier initialization. Batch, Mini-batch and stochastic gradient descent. Regularization-L1/ regularization, Early stopping Dropout regularization, data augmentation. Normalization of inputs-Batch Normalization, Batch Normalization regularizer.



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

UNIT-IV**CONVOLUTIONAL NETWORKS**

Building blocks of CNN, Building a Convolution Neural Network, Popular CNN Architectures- LeNet-5, AlexNet, ZFNET, VGG-16, GoogleNet and ResNet Object Detection-one stage detection techniques – YOLO, SSD, Two stage object detection techniques – R-CNN, fast R-CNN, faster R-CNN, Mask R-CNN, Applications of Object Detection

UNIT-V**SEQUENCE-BASED MODELS AND OTHER DL ARCHITECTURES**

Recurrent Neural Network – Data Preparation for RNN Vanishing Gradient problem and RNN, Applications of RNN, Types of RNN, Limitations of RNN, Long short-term memory(LSTM), Gated RNNs, Bidirectional RNNs. Other DL Architectures-Autoencoder, Architecture and its applications, GAN, GAN Architecture and its applications,

TEXT BOOKS:

1. Amit Kumar Das, Saptarsi Goswami, Pabitra Mitra, Amlan Chakrabarti, "Deep Learning", Pearson Paperback, First Edition, 2021.
2. Ian Goodfellow, Yoshua Bengio, Aaron Courville, "Deep Learning", MIT Press, 2016.

REFERENCES:

1. Josh Patterson and Adam Gibson, "Deep learning: A practitioner's approach", O'Reilly Media, First Edition, 2017.
2. Nikhil Buduma, "Fundamentals of Deep Learning, Designing Next-generation machine intelligence algorithms", O'Reilly, Shroff Publishers, 2019.

REFERENCE WEBSITES:

1. <https://keras.io/datasets/>
2. <http://deeplearning.net/tutorial/deeplearning.pdf>
3. <https://arxiv.org/pdf/1404.7828v4.pdf>
4. <https://www.cse.iitm.ac.in/~miteshk/CS7015.html>

Course Outcomes (COs):

1. Understand the mathematical background required for Deep learning.
2. Analyze the fundamental concepts of Artificial neural networks.
3. Understand and apply the deep Neural networks.
4. Explore the Purpose of Convolution Neural Network, Popular Architectures of CNN.
5. To learn about sequence-based models like RNN, LSTM and Gated RNN and other DL architectures and use for real-world problems.

Mapping of Course Outcomes with Program Outcomes:														
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	4	3	3	3	-	-	-	1	2	1	3	1	3
CO2	3	3	3	3	3	-	-	-	1	2	1	-	2	3
CO3	2	3	2	3	2	-	-	-	1	-	1	-	2	3
CO4	3	2	2	3	2	-	-	-	1	-	1	3	2	3
CO5	3	3	3	3	2	-	-	-	1	2	1	3	1	3

1 – Slightly; 2 - Moderately; 3 – Substantially



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

PE -IV- 2 - BLOCK CHAIN TECHNOLOGIES**Credits: 3****Assessment: 40+ 60**

Semester	L-T-P	Course Code	Branches
VII	3-0-0	7-6-7-R25E22	CS

Prerequisite: Cryptography and Network security, Data Structures, Distributed Systems.**Course Educational Objectives (CEOs):**

1. To provide fundamental concepts of Block Chain Mechanisms.
2. To gain knowledge about different Architecture of various Block Chains.
3. To gain knowledge about the Applications of Block Chain.

UNIT - I

Block Chain: Distributed System-History of Block Chain-Introduction to Block chain-Types of Block Chain-CAP Theorem-Benefits & Limitations of Block Chain.

UNIT - II

Decentralization:Decentralization using Block Chain-Method of Block Chain-Toutes to Block Chain-Block Chain & Full Eco System-Decentralized Applications-Platform for Decentralization.

UNIT - III

Bitcoin & Alternative Coin:Bitcoin-Transaction-Block Chain-Bitcoin Payment-Bitcoin Limitation-Namecoin-Litecoin-Primcoin-Zcash.

UNIT - IV

Ethereum 101: Introduction-Ethereum Block Chain-Elements of Ethereum Block Chain-Accounts0 Types of AccountsEther-Messages-Mining-Clients & Wallets-Ethereum Network.

UNIT - V

Block Chain outside of Currencies, Scalability & Other Challenges -Block Chain outside of Currencies: Internet of Things-Government-Health-Finance-Medical. Scalability & Other Challenges: Scalability-Privacy-Security.

TEXT BOOK :

1. Mastering Block Chain, 2017, Imran Bashir, Packt publishing Limited.

REFERENCE BOOKS :

2. Mastering the Bitcoin-Programming the Open Block Chain, 2017, Andreas M Antonopoulos,O'Reilly Publication.
3. Block Chain-Blue Print for a New Economy, 2015,Melanie Swan, O'Reilly Publication.
4. Block Chain Revolution, 2016, Don Tapscott, Portfolio Penguin Publications.



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

Course Outcomes(COs):

1. Understand the preliminary history and principles of block chain architecture components and techniques
2. Relate the platform for decentralization with block chain technology
3. Apply Bit-coin system by using block chain technology
4. Design hyper ledger applications using block chain technology
5. Relate general applications by using block chain technology and other challenge.

Mapping of Course Outcomes with Program Outcomes:														
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	2	3	3	3	-	-	-	1	2	1	3	1	3
CO2	3	2	3	3	3	-	-	-	-	2	1	3	2	3
CO3	3	2	2	3	1	-	-	-	-	-	1	3	2	3
CO4	3	2	2	3	1	-	-	-	1	2	1	3	2	3
CO5	3	3	3	3	2	-	-	-	1	2	1	3	1	3

1 – Slightly; 2 - Moderately; 3 – Substantially



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

PE -IV- 3 - FOG AND EDGE COMPUTING**Credits: 3****Assessment: 40+ 60**

Semester	L-T-P	Course Code	Branches
VII	3-0-0	7-6-7-R25E23	CS

Prerequisite: Computer Networks, Cloud Computing ,OS, IoT fundamentals.**Course Educational Objectives (CEOs):**

1. To introduce basic concepts of Fog and Edge Computing suitable for undergraduate students.
2. To understand simple architectures and components of Fog and Edge Computing systems.
3. To provide foundational knowledge of communication and security in Fog and Edge environments.
4. To expose students to real-world applications and analytics using Fog and Edge Computing.
5. To familiarize students with emerging trends and future scope of Fog and Edge Computing.

UNIT I: Introduction to Fog and Edge Computing

Basics of Cloud Computing, Limitations of Cloud Computing, Need for Fog and Edge Computing, Definitions and Features of Fog and Edge Computing, Comparison of Cloud, Fog, and Edge Computing, Role of IoT, Simple Applications and Use Cases.

UNIT II: Architecture of Fog and Edge Computing

Basic Fog and Edge Architecture, Components of Edge Devices and Fog Nodes, Introduction to Virtualization and Containers, Resource Management Basics, Data Processing at Edge and Fog, Service Placement Concepts, Latency and Bandwidth Issues.

UNIT III: Communication and Security

Introduction to Communication Models, Basic Networking Protocols (MQTT, CoAP, HTTP), Role of SDN and NFV (Overview), Security Issues in Fog and Edge Computing, Authentication and Authorization, Data Privacy and Trust, Common Attacks and Challenges.

UNIT IV: Applications and Analytics

Real-Time Data Processing, Introduction to Edge Analytics, Basics of Machine Learning at the Edge, Applications in Smart Cities, Healthcare, Industrial IoT, Video Surveillance, Energy-Efficient Fog and Edge Systems.



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

UNIT V: Advanced Topics and Future Trends

Introduction to Fog and Edge Platforms, Integration of Cloud with Fog and Edge, Overview of Blockchain and Serverless Computing at the Edge, Mobility Support, Case Studies, Research Challenges and Future Trends.

Text Books

1. Rajkumar Buyya, Amir Vahid Dastjerdi, Fog and Edge Computing: Principles and Paradigms, Wiley.
2. Flavio Bonomi et al., Fog Computing and Its Role in the Internet of Things, Cisco Press.

Reference Books

3. Weisong Shi et al., Edge Computing: Vision and Challenges, IEEE Computer Society.
4. Shancang Li, Li Da Xu, Shanshan Zhao, Edge Computing: Models, Technologies and Applications, CRC Press.
5. Ovidiu Vermesan, Peter Friess, Internet of Things – From Research and Innovation to Market Deployment, River Publishers.

Course Outcomes (COs):

After completing this course, students will be able to:

1. Describe the fundamentals of Fog and Edge Computing and explain their importance in IoT applications.
2. Explain basic architectures and components used in Fog and Edge Computing systems.
3. Apply basic communication protocols and security concepts in Fog and Edge environments.
4. Identify and explain real-world applications and analytics performed at the Fog and Edge layers.
5. Summarize recent advancements and future directions in Fog and Edge Computing.

Mapping of Course Outcomes with Program Outcomes:															
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	
CO1	3	2	3	3	3	-	-	-	-	-	-	3	2	3	
CO2	3	2	3	3	3	-	-	-	-	-	-	3	2	3	
CO3	3	2	2	3	1	-	-	-	1	-	-	3	2	3	
CO4	3	2	2	3	1	-	-	-	1	-	-	3	2	3	
CO5	3	3	3	3	2	-	-	-	1	-	-	3	2	3	

1 – Slightly; 2 - Moderately; 3 – Substantially



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

PE -V- 1 - BIG DATA ANALYTICS**Credits: 3****Assessment: 40+ 60**

Semester	L-T-P	Course Code	Branches
VII	3-0-0	7-6-7-R25E31	CS

Prerequisite: AI, Machine Learning.**Course Educational Objectives (CEOs):**

The course is designed to

1. Discuss the challenges traditional data mining algorithms face when analyzing Big Data.
2. Introduce the tools required to manage and analyze big data like Hadoop, NoSql MapReduce.
3. Comprehend machine learning and deep learning algorithms for data analytics.
4. Understand how big data analytics can leverage into a key component

UNIT I

Introduction to Analytics and Big Data - Characteristics of Big Data, Domain Specific Examples of Big Data Viz. Web, Financial, Healthcare, IoT, Environment, Logistics & Transportation, Industry, Retail; Analytics Flow for Big Data, Big Data Stack, Mapping Analytics Flow to Big Data Stack, Case Studies: Genome Data Analysis and Weather Data Analysis, Analytics Patterns.

UNIT II

Overview of Setting up Big Data Stack, Big Data Patterns - Analytics Architecture Components & Design Styles, MapReduce Patterns.

NoSQL - Key-Value Databases, Document Databases, Column Family Databases, Graph Databases.

UNIT III

Data Acquisition - Data Acquisition Considerations, Publish-Subscribe Messaging Frameworks, Big Data Collection Systems, Messaging Queues, Custom Connectors.

Big Data Storage - HDFS: Architecture and Usage Examples.

Batch Analysis - Hadoop and MapReduce, Hadoop-MapReduce Examples, Pig, Case Study: Batch Analysis of News Articles, Apache Oozie, Apache Spark, Apache Solr.

UNIT IV

Real-time Analysis - Stream Processing, Storm Case Studies, In-Memory Processing, Spark Case Studies.

Interactive Querying - Spark SQL, Hive, Amazon Redshift, Google BigQuery.

Serving Databases & Web Frameworks - Relational (SQL) Databases, Non-Relational (NoSQL) Databases, Python Web Application Framework-Django, Case Study: Django application for viewing weather data.



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

UNIT V

Analytics Algorithms - Frameworks, Clustering, Case Study: Song Recommendation System, Classification & Regression, Case Studies: Classifying Handwritten Digits and Genome Data Analysis (Implementation), Recommendation Systems.

Data Visualization - Frameworks & Libraries, Overview of Visualization Examples.

Text Books:

1. Arshdeep Bhaga, Vijay Madiseti, Big Data Science and Analytics: A Hands-On Approach, 1st Edition, VPT Publishers, 2019.
2. Michael Minelli, Michele Chambers, Ambiga Dhiraj, Big Data, Big Analytics: Emerging Business Intelligence and Analytic Trends for Today's Businesses, Wiley, 2013.

Reference Books:

3. EMC Education Services, "Data Science & Big Data Analytics: Discovering, Analyzing, Visualizing and Presenting Data", Wiley, 2016.
4. Frank J Ohlhorst, "Big Data Analytics: Turning Big Data into Big Money", Wiley and SAS Business Series, 2012.
5. Rajkumar Buyya, Rodrigo N. Calheiros and Amir Vahid Dastjerdi (Editors), "Big Data Principles and Paradigms", Morgan Kaufmann (An imprint of Elsevier), 2016.
6. Colleen McCue, "Data Mining and Predictive Analysis: Intelligence Gathering and Crime Analysis", Elsevier, 2007.
7. Michael Berthold, David J. Hand, Intelligent Data Analysis, Springer, 2007.
8. Anand Rajaraman and Jeffrey David Ullman, Mining of Massive Datasets, Cambridge University Press, 2012.
9. Bill Franks, "Taming the Big Data Tidal Wave: Finding Opportunities in Huge Data Streams with Advanced Analytics", Wiley and SAS Business Series, 2012.
10. Paul Zikopoulos, Chris Eaton, Paul Zikopoulos, "Understanding Big Data: Analytics for Enterprise Class Hadoop and Streaming Data", McGraw Hill, 2011.

Course Outcomes(COs):

By the end of this course students will be able to

1. Understand big data challenges in different domains viz. social media, transportation, finance, medicine and apply the concepts of big data analytics for the said domains.
2. Apply several newer algorithms for Clustering, Classifying and finding associations in Big Data
3. Design and develop Hadoop and Map Reduce Framework
4. Understand how big data analytics can leverage into a key component
5. Handle several Data Intensive tasks using the Map Reduce Paradigm



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

Mapping of Course Outcomes with Program Outcomes:														
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	2	3	3	3	-	-	-	-	-	-	2	3	3
CO2	3	3	3	3	3	-	-	-	-	-	-	2	3	3
CO3	3	3	2	3	1	-	-	-	-	2	3	3	3	3
CO4	3	2	2	3	1	-	-	-	-	2	1	3	2	3
CO5	3	3	3	3	2	-	-	-	-	2	1	3	2	3

1 – Slightly; 2 - Moderately; 3 – Substantially



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

PE -V- 2 - NATURAL LANGUAGE PROCESSING**Credits: 3****Assessment: 40+ 60**

Semester	L-T-P	Course Code	Branches
VII	3-0-0	7-6-7-R25E32	CS

Prerequisite: AI, Machine Learning.**Course Educational Objectives (CEOs):**

1. Introduce to some of the problems and solutions of NLP and their relation to linguistics and statistics.

UNIT I**Finding the Structure of Words:** Words and Their Components, Issues and Challenges, Morphological Models**Finding the Structure of Documents:** Introduction, Methods, Complexity of the Approaches, Performances of the Approaches**UNIT II****Syntax Analysis:** Parsing Natural Language, Treebanks: A Data-Driven Approach to Syntax, Representation of Syntactic Structure, Parsing Algorithms, Models for Ambiguity Resolution in Parsing, Multilingual Issues**UNIT III****Semantic Parsing:** Introduction, Semantic Interpretation, System Paradigms, Word Sense Systems, Software.**UNIT IV**

Predicate-Argument Structure, Meaning Representation Systems, Software.

UNIT V**Discourse Processing:** Cohension, Reference Resolution, Discourse Cohension and Structure**Language Modeling:** Introduction, N-Gram Models, Language Model Evaluation, Parameter Estimation, Language Model Adaptation, Types of Language Models, Language- Specific Modeling Problems, Multilingual and Crosslingual Language Modeling**TEXT BOOKS:**

1. Multilingual natural Language Processing Applications: From Theory to Practice
2. Daniel M. Bikel and Imed Zitouni, Pearson Publication


Chairperson
(BoS UG)


DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

3. Natural Language Processing and Information Retrieval: Tanvier Siddiqui, U.S. Tiwary
4. Jurafsky D, Martin J H, Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics and Speech Recognition, 2nd edition, Pearson Education, 2013. 7

Reference Books:

5. Chopra D, Joshi N, Mastering Natural Language Processing with Python, Packt, 2016.
- Watanabe S, Jen-Tzung Chien J T, Bayesian Speech and Language Processing, Cambridge University Press, 2015.
6. Kurdi M Z, Natural Language Processing and Computational Linguistics: Speech, Morphology and Syntax, Wiley, 2016.
7. Tiwary U S, Siddiqui T, Natural Language Processing & Information Retrieval, Oxford University Press, 2008.
8. Bikel D M, Zitouni I, Multilingual Natural Language Processing Applications: From Theory to Practice, Pearson Education, 2012.
9. Bharati A, Chaitanya V, Sangal R, Natural Language Processing: A Paninian Perspective, PHI, 1995. 7. Steven B, Natural Language Processing with Python, Shroff, 2009. 8. Mariani J, Language and Speech Processing, Wiley, 2013.

Course Outcomes(COs):

1. Show sensitivity to linguistic phenomena and an ability to model them with formal grammars.
2. Understand and carry out proper experimental methodology for training and evaluating empirical NLP systems
3. Able to manipulate probabilities, construct statistical models over strings and trees, and estimate parameters using supervised and unsupervised training methods.
4. Able to design, implement, and analyze NLP algorithms
5. Able to design different language modeling Techniques.

Mapping of Course Outcomes with Program Outcomes:														
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	2	3	3	3	-	-	-	-	-	1	-	3	3
CO2	3	3	3	3	3	-	-	-	-	-	2	-	3	3
CO3	3	2	2	3	3	-	-	-	-	2	2	3	3	3
CO4	3	2	2	3	3	-	-	-	-	2	1	3	3	3
CO5	3	3	3	3	2	-	-	-	-	2	1	3	3	3

1 – Slightly; 2 - Moderately; 3 – Substantially



Chairperson
(BoS UG)


DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

PE -V- 3 - DEVOPS AND AGILE METHODOLOGIES**Credits: 3****Assessment: 40+ 60**

Semester	L-T-P	Course Code	Branches
VII	3-0-0	7-6-7-R25E33	CS

Prerequisite: Software Engineering, Operating Systems, Cloud Computing.**Course Educational Objectives (CEOs):**

The course is designed to

1. To introduce Agile and DevOps concepts for modern software development.
2. To understand Agile frameworks, practices, and team collaboration models.
3. To provide foundational knowledge of DevOps tools and automation techniques.
4. To familiarize students with cloud, containerization, and configuration management concepts.
5. To expose students to industry practices, security aspects, and real-world case studies.

UNIT I: Introduction to Agile and DevOps

Software Development Challenges, Traditional Software Development Models (Waterfall – Overview), Introduction to Agile Methodology, Agile Manifesto and Principles, Agile vs Traditional Methods, Introduction to DevOps, Need for DevOps, DevOps Culture and Benefits, Agile–DevOps Relationship.

UNIT II: Agile Frameworks and Practices

Scrum Framework: Roles, Events, and Artifacts, Sprint Planning and Review, Product Backlog and Sprint Backlog, Kanban Principles and Board, Extreme Programming (XP) Practices, User Stories, Story Points, Agile Estimation Techniques, Agile Testing Basics.

UNIT III: DevOps Tools and Continuous Integration

DevOps Lifecycle, Version Control Systems (Git Basics), Continuous Integration Concepts, Build Tools Overview, Continuous Testing, Continuous Delivery and Continuous Deployment (Concepts), Introduction to Automation, Infrastructure as Code (Overview).

UNIT IV: Configuration Management and Cloud in DevOps

Configuration Management Concepts, Overview of Tools (Ansible, Chef, Puppet), Containerization Concepts, Introduction to Docker, Basics of Kubernetes (Overview), Role of Cloud Computing in DevOps, Monitoring and Logging (Introduction).



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

UNIT V: DevOps Practices, Security, and Case Studies

DevOps Best Practices, DevSecOps Concepts, Security in CI/CD Pipelines (Overview), Performance and Reliability in DevOps, Collaboration and Communication in DevOps Teams, Case Studies, Challenges and Future Trends in Agile and DevOps.

TEXTBOOKS

1. Gene Kim, Jez Humble, Patrick Debois, John Willis, The DevOps Handbook, IT Revolution Press.
2. Ken Schwaber, Jeff Sutherland, The Scrum Guide, Scrum.org.

REFERENCE BOOKS

3. Jez Humble, David Farley, Continuous Delivery, Addison-Wesley.
4. Eric Ries, The Lean Startup, Crown Publishing.
5. Len Bass, Ingo Weber, Liming Zhu, DevOps: A Software Architect's Perspective, Addison-Wesley.

Course Outcomes(COs):

After completing this course, students will be able to:

1. Explain Agile principles and DevOps concepts used in modern software development.
2. Apply Agile frameworks such as Scrum and Kanban for effective project management.
3. Describe DevOps lifecycle stages and basic CI/CD practices.
4. Explain the role of containers, cloud platforms, and configuration management in DevOps.
5. Identify best practices, security considerations, and emerging trends in Agile and DevOps.

Mapping of Course Outcomes with Program Outcomes:														
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	3	3	3	3	-	-	-	1	2	1	3	3	3
CO2	3	3	3	3	3	-	-	-	1	2	1	3	3	3
CO3	3	3	2	3	2	-	-	-	-	2	1	3	3	3
CO4	3	2	2	3	2	-	-	-	-	2	1	3	3	3
CO5	3	3	3	3	2	-	-	-	-	2	1	3	3	3

1 – Slightly; 2 - Moderately; 3 – Substantially



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502

B. Tech. (Computer Science and Engineering)
Semester VIII (Fourth Year)

Code		Category	Course Title	Hours/Week			Total	Credits	Marks		
Department	University			L	T	P			CIE	SEE	Total
CS6801PI	7-0-8-R25PS	CS-PWSI	ProjectWork & Full Semester Internship	0	0	24	24	12	50	50	100
TOTAL				0	0	24	24	12	50	50	100

In the final semester, the student should mandatorily undergo full semester internship and parallelly he/she should ***work on a project with well-defined objectives***. At the end of the semester the candidate shall submit an internship report and its completion certificate in the department. Student is permitted to submit report on the work carried out during the internship as project report provided the internship work is in the discipline of his B.Tech. Program. Internship programs offered by AICTE/APSSDC/NSDC/SWAYAM PLUS /Any State or Central Government organizations involving Industry experts and approved by the concerned BoS/Head/Head of the Institution are also considered equivalent to full semester Internship provided it involves 90 – 120 hours of work. The Internship work area can be in his own discipline or inter-disciplinary (One discipline being his own) or other discipline also. The project work shall be either in the discipline of his own or inter-disciplinary.



Chairperson
(BoS UG)


 DEAN
 FACULTY OF ENGINEERING
 S. V. U. COLLEGE OF ENGINEERING
 TIRUPATI-517 502

EXIT OPTIONS FOR B.TECH

Two months internship in industry or bridge courses mentioned below of 6 to 8 credits on NPTEL – SWAYAM Platform

Semester	Exit Option	Credits	Additional Credits for exit students	List of exit courses
Sem I & II	U.G Certificate	40	6-8	1. Probability & Statistics 2. Database Management Systems 3. Discrete Mathematical Structures 4. Digital Electronics and Logic Design 5. Object Oriented Programming through Java
Sem III & IV	U.G Diploma	40.5	6-8	1. Formal Languages and Automata Theory 2. Computer Networks 3. Operating Systems 4. Software Engineering 5. Advanced Python Programming



Chairperson
(BoS UG)



DEAN
FACULTY OF ENGINEERING
S. V. U. COLLEGE OF ENGINEERING
TIRUPATI-517 502