

COURSE STRUCTURE

AND

DETAILED SYLLABUS

COMPUTER SCIENCE AND ENGINEERING

FOR

M.TECH TWO YEARS DEGREE COURSE

(Applicable for the batches admitted from 2025-2026)



**VAAGDEVI COLLEGE OF ENGINEERING
AUTONOMOUS**

Bollikunta, Warangal-506005

Telangana State, India.

VAAGDEVI COLLEGE OF ENGINEERING
AUTONOMOUS
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
M.Tech in COMPUTER SCIENCE AND ENGINEERING

COURSE STRUCTURE

(R25 Regulations applicable for the batches admitted from Academic Year 2025-26 onwards)

I YEAR I – SEMESTER

| S.No | Course Code | Course Title | L | T | P | Credits |
|--------------|-------------|--|-----------|----------|----------|-----------|
| 1 | M25CS01 | Mathematical Foundations of Computer Science | 3 | 0 | 0 | 3 |
| 2 | M25CS02 | Advanced Data Structures | 3 | 0 | 0 | 3 |
| 3 | | Professional Elective – I | 3 | 0 | 0 | 3 |
| | M25CS03 | 1. Database Programming with PL/SQL | | | | |
| | M25AI07 | 2. Deep Learning | | | | |
| | M25CS04 | 3. Natural Language Processing | | | | |
| | M25CS05 | 4. Advanced UNIX Programming | | | | |
| 4 | | Professional Elective – II | 3 | 0 | 0 | 3 |
| | M25CS06 | 1. Applied Cryptography | | | | |
| | M25CS07 | 2. Software Quality Engineering | | | | |
| | M25CS08 | 3. Mining Massive Datasets | | | | |
| | M25CS09 | 4. Agile Methodologies | | | | |
| 5 | M25CS10 | Advanced Data Structures Lab | 0 | 0 | 4 | 2 |
| 6 | | Professional Elective - I Lab | 0 | 0 | 4 | 2 |
| | M25CS11 | 1. Database Programming with PL/SQL Lab | | | | |
| | M25AI25 | 2. Deep Learning Lab | | | | |
| | M25CS12 | 3. Natural Language Processing Lab | | | | |
| | M25CS13 | 4. Advanced UNIX Programming Lab | | | | |
| 7 | M25MC01 | Research Methodology & IPR | 2 | 0 | 0 | 2 |
| 8 | M25AC01 | Audit Course- I | 2 | 0 | 0 | 0 |
| Total | | | 16 | 0 | 8 | 18 |



VAAGDEVI COLLEGE OF ENGINEERING
AUTONOMOUS
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
M.Tech in COMPUTER SCIENCE AND ENGINEERING

COURSE STRUCTURE

(R25 Regulations applicable for the batches admitted from Academic Year 2025-26 onwards)

I YEAR II – SEMESTER

| S.No | Course Code | Course Title | L | T | P | Credits |
|------|-------------|--|-----------|----------|-----------|-----------|
| 1 | M25CS14 | Advanced Algorithms | 3 | 0 | 0 | 3 |
| 2 | M25CS15 | Advanced Computer Architecture | 3 | 0 | 0 | 3 |
| 3 | | Professional Elective – III | 3 | 0 | 0 | 3 |
| | M25CS16 | 1. Enterprise Cloud Concepts | | | | |
| | M25CS17 | 2. Cyber Security | | | | |
| | M25CS18 | 3. Parallel computing | | | | |
| | M25CS19 | 4. Large Language Models | | | | |
| 4 | | Professional Elective – IV | 3 | 0 | 0 | 3 |
| | M25CS20 | 1. Bioinformatics | | | | |
| | M25CS21 | 2. Adhoc Sensor Networks | | | | |
| | M25CS22 | 3. Robotic Process Automation | | | | |
| | M25AI10 | 4. Generative AI | | | | |
| 5 | M25CS23 | Advanced Algorithms Lab | 0 | 0 | 4 | 2 |
| 6 | | Professional Elective - III Lab | 0 | 0 | 4 | 2 |
| | M25CS24 | 1. Enterprise Cloud Concepts Lab | | | | |
| | M25CS25 | 2. Cyber Security Lab | | | | |
| | M25CS26 | 3. Parallel computing Lab | | | | |
| | M25CS27 | 4. Large Language Models Lab | | | | |
| 7 | M25CS28 | Mini Project with Seminar | 0 | 0 | 4 | 2 |
| 8 | M25AC02 | Audit Course- II | 2 | 0 | 0 | 0 |
| | | Total | 14 | 0 | 12 | 18 |



VAAGDEVI COLLEGE OF ENGINEERING
AUTONOMOUS
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
M.Tech in COMPUTER SCIENCE AND ENGINEERING

COURSE STRUCTURE

(R25 Regulations applicable for the batches admitted from Academic Year 2025-26 onwards)

II YEAR I – SEMESTER

| S.No | Course Code | Course Title | L | T | P | Credits |
|------|-------------|----------------------------------|----------|----------|-----------|-----------|
| | | Professional Elective - V | | | | |
| 1 | M25CS29 | 1. Digital Forensics | 3 | 0 | 0 | 3 |
| | M25CS30 | 2. Advanced Operating Systems | | | | |
| | M25CS31 | 3. Quantum Computing | | | | |
| | M25AI21 | 4. Prompt Engineering | | | | |
| 2 | | Open Elective | 3 | 0 | 0 | 3 |
| 3 | M25CS32 | Dissertation Work Review - II | 0 | 0 | 18 | 6 |
| | | Total | 6 | 0 | 18 | 12 |



VAAGDEVI COLLEGE OF ENGINEERING
AUTONOMOUS
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
M.Tech in COMPUTER SCIENCE AND ENGINEERING

COURSE STRUCTURE

(R25 Regulations applicable for the batches admitted from Academic Year 2025-26 onwards)

II YEAR II – SEMESTER

| S.No | Course Code | Course Title | L | T | P | Credits |
|------|-------------|--------------------------------|----------|----------|-----------|-----------|
| 1 | M25CS33 | Dissertation Work Review – III | 0 | 0 | 18 | 6 |
| 2 | M25CS34 | Dissertation Viva-Voce | 0 | 0 | 42 | 14 |
| | | Total | 0 | 0 | 60 | 20 |

*For Dissertation Work Review - I, Please refer R25 Academic Regulations.

Audit Course I&II:

1. English for Research Paper Writing
2. Disaster Management
3. Sanskrit for Technical Knowledge
4. Value Education
5. Constitution of India
6. Pedagogy Studies
7. Stress Management by yoga
8. Personality Development Through Life Enlightenment Skills

Open Electives for other Departments:

1. M25CS35 - Intellectual Property Rights
2. M25AI10- Generative AI
3. M25CS36 - Intrusion Detection Systems
4. M25CS29 - Digital Forensics

**VAAGDEVI COLLEGE OF ENGINEERING
(AUTONOMOUS)
COMPUTER SCIENCE AND ENGINEERING**

**MATHEMATICAL FOUNDATIONS OF COMPUTER SCIENCE
(PC-I)**

M.Tech CSE I Year I Sem.

| | | | |
|----------|----------|----------|----------|
| L | T | P | C |
| 3 | 0 | 0 | 3 |

Pre-requisites: An understanding of Math in general is sufficient.

Course Outcomes: After learning the contents of this paper the student must be able to

1. Ability to understand and construct precise mathematical proofs.
2. Ability to use logic and set theory to formulate precise statements.
3. Ability to analyze and solve counting problems on finite and discrete structures.
4. Ability to describe and manipulate sequences.
5. Ability to apply graph theory in solving computing problems.

UNIT-I

The Foundations Logic and Proofs: Propositional Logic, Applications of Propositional Logic, Propositional Equivalence, Predicates and Quantifiers, Nested Quantifiers, Rules of Inference, Introduction to Proofs, Proof Methods and Strategy.

UNIT-II

Basic Structures, Sets, Functions, Sequences, Sums, Matrices and Relations: Sets, Functions, Sequences & Summations, Cardinality of Sets and Matrices Relations, Relations and Their Properties, n-ary Relations and Their Applications, Representing Relations, Closures of Relations, Equivalence Relations, Partial Orderings.

UNIT-III

Algorithms, Induction and Recursion: Algorithms, The Growth of Functions, Complexity of Algorithms. Induction and Recursion: Mathematical Induction, Strong Induction and Well-Ordering, Recursive Definitions and Structural Induction, Recursive Algorithms, Program Correctness.

UNIT-IV

Discrete Probability and Advanced Counting Techniques:

An Introduction to Discrete Probability. Probability Theory, Bayes' Theorem, Expected Value and Variance.

Advanced Counting Techniques:

Recurrence Relations, Solving Linear Recurrence Relations, Divide-and-Conquer Algorithms and Recurrence Relations, Generating Functions, Inclusion-Exclusion, Applications of Inclusion-Exclusion.

UNIT-V

Graphs: Graphs and Graph Models, Graph Terminology and Special Types of Graphs, Representing Graphs and Graph Isomorphism, Connectivity, Euler and Hamilton Paths, Shortest-Path Problems, Planar Graphs, Graph Coloring.

Trees: Introduction to Trees, Applications of Trees, Tree Traversal, Spanning Trees, Minimum Spanning Trees.

TEXT BOOKS:

1. Discrete Mathematical Structures with Applications to Computer Science: J.P. Tremblay, R. Manohar, McGraw-hill, 1st ed.
2. Discrete Mathematics for Computer Scientists & Mathematicians: Joe I. Mott, Abraham Kandel, Theodore P. Baker, Prentis Hall of India, 2nd ed.

REFERENCE:

1. Discrete and Combinatorial Mathematics - an applied introduction: Ralph.P. Grimald, Pearson education, 5th edition.
2. Discrete Mathematical Structures: Thomas Kosy, Tata McGraw Hill publishing co.

**VAAGDEVI COLLEGE OF ENGINEERING
(AUTONOMOUS)
COMPUTER SCIENCE AND ENGINEERING**

**ADVANCED DATA STRUCTURES
(PC-II)**

M.Tech CSE I Year I Sem.

| | | | |
|----------|----------|----------|----------|
| L | T | P | C |
| 3 | 0 | 0 | 3 |

Prerequisites

1. A course on “Data Structures”.

Course Outcomes

1. Students will be able to apply various heap structures to solve priority queue and optimization problems efficiently.
2. Students will be able to design efficient hashing techniques for effective data storage and retrieval.
3. Students will be able to construct and analyze balanced search structures for optimized searching.
4. Students will be able to implement digital search structures for efficient string processing and retrieval applications.
5. Apply pattern matching algorithms for efficient text searching.

UNIT - I

Heap Structures

Introduction, Min-Max Heaps, Leftist trees, Binomial Heaps, Fibonacci heaps.

UNIT - II

Hashing and Collisions

Introduction, Hash Tables, Hash Functions, different Hash Functions:- Division Method, Multiplication Method, Mid-Square Method, Folding Method, Collisions

UNIT - III

Search Structures

OBST, AVL trees, Red-Black trees, Splay trees,
Multiway Search Trees
B-trees, 2-3 trees

UNIT - IV

Digital Search Structures

Digital Search trees, Binary tries and Patricia, Multiway Tries, Suffix trees, Standard Tries, Compressed Tries

UNIT - V

Pattern matching

Introduction, Brute force, the Boyer –Moore algorithm, Knuth-Morris-Pratt algorithm, Naïve String, Harspool, Rabin Karp

TEXT BOOKS:

1. Fundamentals of data structures in C++ Sahni, Horowitz, Mehatha, Universities Press.
2. Introduction to Algorithms, TH Cormen, PHI

REFERENCES:

1. Design methods and analysis of Algorithms, SK Basu, PHI.
2. Data Structures & Algorithm Analysis in C++, Mark Allen Weiss, Pearson Education.
Fundamentals of Computer Algorithms, 2nd Edition, Ellis Horowitz, Sartaj Sahni, Sanguthevar Rajasekaran, Universities Press.

**VAAGDEVI COLLEGE OF ENGINEERING
(AUTONOMOUS)
COMPUTER SCIENCE AND ENGINEERING
DATABASE PROGRAMMING WITH PL/SQL
(PROFESSIONAL ELECTIVE - I)**

M.Tech CSE I Year I Sem.

| | | | |
|----------|----------|----------|----------|
| L | T | P | C |
| 3 | 0 | 0 | 3 |

Prerequisites : Data Base Concepts & SQL.

Course Outcomes:

1. Understand the structure of PL/SQL blocks, variables, data types, control structures, and exception handling.
2. Apply cursors, collections, and bulk operations to process and manage data efficiently.
3. Design and implement functions and procedures with different parameter-passing methods.
4. Develop and manage packages to achieve security, modularity and reusability in PL/SQL applications.
5. Create various types of triggers to enforce business rules and automate database operations.

Unit I

PL/SQL Basics

Block Structure, Behavior of Variables in Blocks, Basic Scalar and Composite Data Types, Control Structures, Exceptions, Bulk Operations, Functions, Procedures, and Packages, Transaction Scope

Unit II

Language Fundamentals & Control Structures

Lexical Units, Variables and Data Types, Conditional Statements, Iterative Statements, Cursor Structures, Bulk Statements, Introduction to Collections, Object Types: Varray and Table Collections, Associative Arrays, Oracle Collection API

Unit III

Functions and Procedures

Function and Procedure Architecture, Transaction Scope, Calling Subroutines, Positional Notation, Named Notation, Mixed Notation, Exclusionary Notation, SQL Call Notation, Functions, Function Model Choices, Creation Options, Pass-by-Value Functions, Pass-by-Reference Functions, Procedures, Pass-by-Value Procedures, Pass-by-Reference Procedures, Supporting Scripts.

Unit IV

Packages

Package Architecture, Package Specification, Prototype Features, Serially Reusable Precompiler Directive, Variables, Types, Components: Functions and Procedures, Package Body, Prototype Features, Variables, Types, Components: Functions and Procedures, Definer vs. Invoker Rights Mechanics, Managing Packages in the Database Catalog, Finding, Validating, and Describing Packages, Checking Dependencies, Comparing Validation Methods: Timestamp vs. Signature.

Unit V**Triggers**

Introduction to Triggers, Database Trigger Architecture, Data Definition Language Triggers, Event Attribute Functions, Building DDL Triggers, Data Manipulation Language Triggers, Statement-Level Triggers, Row-Level Triggers, Compound Triggers, INSTEAD OF Triggers, System and Database Event Triggers, Trigger Restrictions, Maximum Trigger Size, SQL Statements, LONG and LONG RAW Data Types.

TEXT BOOKS:

1. Oracle Database 12c PL/SQL Programming Michael McLaughlin, McGraw Hill Education.

REFERENCES:

1. Benjamin Rosenzweig, Elena Silvestrova Rakhimov, Oracle PL/SQL by example Fifth Edition.
2. Dr. P. S. Deshpande, SQL & PL / SQL for Oracle 11g Black Book.

**VAAGDEVI COLLEGE OF ENGINEERING
(AUTONOMOUS)
COMPUTER SCIENCE AND ENGINEERING**

**DEEP LEARNING
(PROFESSIONAL ELECTIVE - I)**

M.Tech CSE I Year I Sem.

| | | | |
|----------|----------|----------|----------|
| L | T | P | C |
| 3 | 0 | 0 | 3 |

Pre-requisites: Machine Learning, AI, Python programming.

Course Outcomes:

1. Implement deep learning algorithms, understand neural networks and traverse the layers of data
2. Learn topics such as convolutional neural networks, recurrent neural networks, training deep networks and high-level interfaces
3. Understand applications of Deep Learning to Computer Vision
4. Understand and analyze Applications of Deep Learning to NLP
5. Analyze the sentence and dialogue generation using CNN and LSTMs.

UNIT - I

Introduction: Feed forward Neural networks, Gradient descent and the back propagation algorithm, Unit saturation, the vanishing gradient problem, and ways to mitigate it. ReLU Heuristics for avoiding bad local minima, Heuristics for faster training, Nestors accelerated gradient descent, Regularization, Dropout

UNIT - II

Convolutional Neural Networks: Architectures, convolution/pooling layers, Recurrent Neural Networks: LSTM, GRU, Encoder Decoder architectures. Deep Unsupervised Learning: Auto encoders, Variational Auto-encoders, Adversarial Generative Networks, Auto-encoder and DBM Attention and memory models, Dynamic Memory Models

UNIT - III

Applications of Deep Learning to Computer Vision: Image segmentation, object detection, automatic image captioning, Image generation with Generative adversarial networks, video to text with LSTM models, Attention Models for computer vision tasks

UNIT -IV

Applications of Deep Learning to NLP: Introduction to NLP and Vector Space Model of Semantics, Word Vector Representations: Continuous Skip-Gram Model, Continuous Bag-of-Words model (CBOW), Glove, Evaluations and Applications in word similarity

UNIT -V

Analogy reasoning: Named Entity Recognition, Opinion Mining using Recurrent Neural Networks: Parsing and Sentiment Analysis using Recursive Neural Networks: Sentence Classification using Convolutional Neural Networks, Dialogue Generation with LSTMs

TEXT BOOKS:

1. Deep Learning by Ian Goodfellow, Yoshua Bengio and Aaron Courville, MIT Press.
2. The Elements of Statistical Learning by T. Hastie, R. Tibshirani, and J. Friedman, Springer.
3. Probabilistic Graphical Models. Koller, and N. Friedman, MIT Press.

REFERENCES:

1. Bishop, C, M., Pattern Recognition and Machine Learning, Springer, 2006.
2. Yegnanarayana, B., Artificial Neural Networks PHI Learning Pvt. Ltd, 2009.
3. Golub, G.,H., and Van Loan, C.,F., Matrix Computations, JHU Press,2013.
4. Satish Kumar, Neural Networks: A Classroom Approach, Tata McGraw-Hill Education, 2004.

Extensive Reading:

1. <http://www.deeplearning.net>
2. <https://www.deeplearningbook.org/>
3. <https://developers.google.com/machine-learning/crash-course/ml-intro>
4. www.cs.toronto.edu/~fritz/absps/imagenet.pdf
5. <http://neuralnetworksanddeeplearning.com/>

**VAAGDEVI COLLEGE OF ENGINEERING
(AUTONOMOUS)
COMPUTER SCIENCE AND ENGINEERING**

**NATURAL LANGUAGE PROCESSING
(PROFESSIONAL ELECTIVE - I)**

M.Tech CSE I Year I Sem.

| | | | |
|----------|----------|----------|----------|
| L | T | P | C |
| 3 | 0 | 0 | 3 |

Prerequisites:

1. Data structures, finite automata and probability theory. .

Course Outcomes:

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 Able to design different language modeling Techniques.
5. Able to design different language modeling Techniques.

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: Cohesion, Reference Resolution, Discourse Cohesion 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 Cross Lingual Language Modeling

TEXT BOOKS:

1. Multilingual natural Language Processing Applications: From Theory to Practice – Daniel M. Bikel and Imed Zitouni, Pearson Publication.
2. Natural Language Processing and Information Retrieval: Tanvier Siddiqui, U.S. Tiwary.

REFERENCES:

1. Speech and Natural Language Processing - Daniel Jurafsky & James H Martin, Pearson Publications.

**VAAGDEVI COLLEGE OF ENGINEERING
(AUTONOMOUS)
COMPUTER SCIENCE AND ENGINEERING
ADVANCED UNIX PROGRAMMING
(PROFESSIONAL ELECTIVE - I)**

M.Tech CSE I Year I Sem.

| | | | |
|----------|----------|----------|----------|
| L | T | P | C |
| 3 | 0 | 0 | 3 |

Prerequisites: C Programming Language, Fundamental Operating System Concepts

Course Outcomes:

1. Understand the role of Linux utilities, file handling, process management, and shell scripting in operating system environments.
2. Explain the structure and organization of files and directories, including metadata, permissions, and system-level operations.
3. Describe process concepts, process control, and signal handling mechanisms provided by the operating system.
4. Demonstrate the principles of inter process communication (IPC) and synchronization using pipes, message queues, and semaphores.
5. Analyze shared memory and socket-based communication in client-server environments and compare different IPC mechanisms.

UNIT- I

Linux Utilities - File handling utilities, Security by file permissions, Process utilities, Disk utilities, Networking commands, Filters, Text processing utilities and Backup utilities.

Shell programming with Bourne again shell (bash) - Introduction, shell responsibilities, pipes and Redirection, here documents, running a shell script, the shell as a programming language, shell meta characters, file name substitution, shell variables, command substitution, shell commands, the environment, quoting, test command, control structures, arithmetic in shell, shell script examples, interrupt processing, functions, debugging shell scripts.

UNIT- II

Files and Directories - File Concept, File types, File System Structure, file metadata-Inodes, kernel support for files, system calls for file I/O operations- open, creat, read, write, close, lseek, dup2, file status information-stat family, file and record locking- fcntl function, file permissions - chmod, fchmod, file ownership-chown, lchown, fchown, links-soft links and hard links – symlink, link, unlink. Directories - Creating, removing and changing Directories- mkdir, rmdir, chdir, obtaining current working directory getcwd, Directory contents, Scanning Directories-opendir, readdir, closedir, rewinddir functions.

UNIT- III

Process – Process concept, Layout of a C program image in main memory, Process environment list, environment variables, getenv, setenv, Kernel support for process, process identification, process control - process creation, replacing a process image, waiting for a process, process termination, zombie process, orphan process, system call interface for process management fork, vfork, exit, wait, waitpid, exec family, Process Groups, Sessions and Controlling Terminal, Differences between threads and processes.

Signals – Introduction to signals, Signal generation and handling, Kernel support for signals, Signal function, unreliable signals, reliable signals, kill, raise, alarm, pause, abort, sleep functions.

UNIT- IV

Interprocess Communication - Introduction to IPC, IPC between processes on a single computer system, IPC between processes on different systems, pipes-creation, IPC between related processes using unnamed pipes, FIFOs-creation, IPC between unrelated processes using FIFOs (Named pipes), differences between unnamed and named pipes, popen and pclose library functions. Message Queues - Kernel support for messages, APIs for message queues, client/server example. Semaphores - Kernel support for semaphores, APIs for semaphores, file locking with semaphores.

UNIT- V

Shared Memory - Kernel support for shared memory, APIs for shared memory, shared memory example. Sockets - Introduction to Berkeley Sockets, IPC over a network, Client-Server model, Socket address structures (Unix domain and Internet domain), Socket system calls for connection oriented protocol and connectionless protocol, example-client/server programs-Single Server-Client connection, Multiple simultaneous clients, Socket options-setsockopt andfcntl system calls, Comparison of IPC mechanisms.

TEXT BOOKS:

1. Unix System Programming using C++, T. Chan, PHI.
2. Advanced Programming in the Unix Environment, 2nd edition, W. R. Stevens and S. A. Rago, Pearson Education.
3. Unix Concepts and Applications, 4th Edition, Sumitabha Das, TMH.
4. Unix Network Programming, W. R. Stevens, PHI.

REFERENCE BOOKS:

1. C Programming Language, Kernighan and Ritchie, PHI.
2. Beginning Linux Programming, 4th Edition, N. Matthew, R. Stones, Wrox, Wiley India Edition.
3. Unix for programmers and users, 3rd Edition, Graham Glass, King Ables, Pearson.
4. System Programming with C and Unix, A. Hoover, Pearson.
5. Unix System Programming, Communication, Concurrency and Threads, K. A. Robbins and S. Robbins, Pearson Education.
6. Unix shell Programming, S. G. Kochan and P. Wood, 3rd edition, Pearson Education.
7. Shell Scripting, S. Parker, Wiley India Pvt. Ltd.
8. Unix and Shell programming, B. A. Forouzan and R. F. Gilberg, Cengage Learning.

**VAAGDEVI COLLEGE OF ENGINEERING
(AUTONOMOUS)
COMPUTER SCIENCE AND ENGINEERING**

**APPLIED CRYPTOGRAPHY
(PROFESSIONAL ELECTIVE - II)**

M.Tech CSE I Year I Sem.

| | | | |
|----------|----------|----------|----------|
| L | T | P | C |
| 3 | 0 | 0 | 3 |

Prerequisites: Mathematical maturity with strong foundational knowledge in discrete mathematics.

Course Outcomes:

1. Understand the various cryptographic protocols
2. Analyze key length and algorithm types and modes
3. Illustrate different public key algorithms in cryptosystems
4. Understand special algorithms for protocols and usage in the real world.
5. Understand and apply key real-world cryptographic protocols for secure communication and electronic payment systems.

Unit I

Foundations:

Terminology, Steganography, Substitution Ciphers and Transposition Ciphers, Simple XOR, One-Time Pads, Computer Algorithms, Large Numbers, Cryptographic Protocols: Protocol Building Blocks Introduction to Protocols, Communications Using Symmetric Cryptography, One-Way Functions, OneWay Hash Functions, Communications Using Public-Key-Cryptography, Digital Signatures, Digital Signatures with Encryption, Random and Pseudo-Random-Sequence Generation

Unit II

Cryptographic Techniques

Key length: Symmetric Key length, Public key length, comparing symmetric and public key length. Algorithm types and modes: Electronic Codebook Mode, Block Replay, Cipher Block Chaining Mode, Stream Cipher, Self-Synchronizing Stream Ciphers, Cipher-Feedback Mode, Synchronous Stream Ciphers, Output-Feedback Mod, Counter Mode, Other Block-Cipher Modes.

Unit III

Public-Key Algorithms

Background, Knapsack Algorithms, RSA, Pohlig-Hellman, Rabin, ElGamal, McEliece, Elliptic Curve Cryptosystems, LUC, Finite Automaton Public-Key Cryptosystems
Public-Key Digital Signature Algorithms: Digital Signature Algorithm (DSA), DSA Variants, Gost Digital Signature Algorithm, Discrete Logarithm Signature Schemes, Ong-Schnorr-Shamir, ESIGN

Unit IV

Special Algorithms for Protocols

Multiple-Key Public-Key Cryptography, Secret-Sharing Algorithms, Subliminal Channel, Undeniable Digital Signatures, Designated Confirmer Signatures, Computing with Encrypted Data, Fair Coin Flips, One-Way Accumulators, All-or-Nothing Disclosure of Secrets, Fair and Failsafe Cryptosystems, ZeroKnowledge Proofs of Knowledge, Blind Signatures, Oblivious Transfer, Secure Multiparty Computation, Probabilistic Encryption, Quantum Cryptography

Unit V

Real World Approaches

IBM Secret key management protocol, ISDN, Kerberos, KryptoKnight, Privacy enhanced mail (PEM), Message security protocol (MSP), PGP, Public-Key Cryptography Standards (PKCS), Universal Electronic Payment System (UEPS).

TEXT BOOKS:

1. Bruce Schneier, Applied Cryptography, Second Edition: Protocols, Algorithms, and Source Code in C (cloth).

**VAAGDEVI COLLEGE OF ENGINEERING
(AUTONOMOUS)
COMPUTER SCIENCE AND ENGINEERING

SOFTWARE QUALITY ENGINEERING
(PROFESSIONAL ELECTIVE - II)**

M.Tech CSE I Year I Sem.

| | | | |
|----------|----------|----------|----------|
| L | T | P | C |
| 3 | 0 | 0 | 3 |

Prerequisites: Knowledge in Problem Solving skills. *Software Engineering*

Course Outcomes:

1. Understand various perspectives and expectations of software quality in modern software engineering.
2. Analyze defect prevention and defect reduction in software quality assurance
3. Define and distinguish between correctness and defects in software systems.
4. Explore different properties and measurements associated with software defects and correctness.
5. Examine the historical evolution of software quality practices and their influence on current methodologies.

Unit I

Software Quality

Quality: perspectives and expectations, Quality frameworks and ISO-9126, correctness and defects: Definitions, properties and Measurements, A historical perspective of quality, software quality.

Unit II

Quality Assurance

Classification: QA as dealing with defects, Defect prevention- Education and training, Formal method, Other defect prevention techniques, Defect Reduction - Inspection: Direct fault detection and removal, Testing: Failure observation and fault removal, other techniques and risk identification, Defect Containment- software fault tolerance, safety assurance and failure containment

Unit III

Quality Engineering

Quality Engineering: Activities and process, Quality planning: Goal setting and Strategy formation, Quality assessment and Improvement, Quality engineering in software process.

Unit IV

Test Activities, Management and Automation

Test planning and preparation, Test execution, Result checking and measurement, Analysis and follow-up, Activities People and Management, Test Automation.

Unit V

Coverage and usage testing based on checklist and partitions

Checklist based testing and its limitations, Testing for partition Coverage, Usage based Statistical testing with Musa's operational profiles, Constructing operational profiles Case Study: OP for the cartridge Support Software

TEXT BOOKS:

1. Jeff Tian, Software Quality Engineering, Testing, Quality Assurance, and Quantifiable improvement
2. Richard N. Taylor, Software Architecture: Foundations, Theory, and Practice

**VAAGDEVI COLLEGE OF ENGINEERING
(AUTONOMOUS)
COMPUTER SCIENCE AND ENGINEERING**

**MINING MASSIVE DATASETS
(PROFESSIONAL ELECTIVE - II)**

M.Tech CSE I Year I Sem.

| | | | |
|----------|----------|----------|----------|
| L | T | P | C |
| 3 | 0 | 0 | 3 |

Prerequisites:

1. Students should be familiar with Data mining, DataStructures and Algorithms, Probability and Statistics,SQL,Distributed Databases and Discrete Mathematics

Course Outcomes:

1. Understand the Principles and Techniques of Data Mining.
2. Analyze and Process Large-Scale Datasets.
3. Exploring Graph Mining Techniques.
4. Data Mining for Social Good and Public Policy
5. Mining E-Commerce Data for Business Insights.

UNIT I

Data Mining-Introduction-Definition of Data Mining-Statistical Limits on Data Mining, MapReduce and the New Software Stack-Distributed File Systems, MapReduce, Algorithms Using MapReduce.

UNIT II

Similarity Search: Finding Similar Items-Applications of Near-Neighbor Search, Shingling of Documents, Similarity-Preserving Summaries of Sets, Distance Measures.
Streaming Data: Mining Data Streams-The Stream Data Model , Sampling Data in a Stream, Filtering Streams

UNIT III

Link Analysis-PageRank, Efficient Computation of PageRank, Link Spam
Frequent Itemsets-Handling Larger Datasets in Main Memory, Limited-Pass Algorithms, Counting Frequent Items in a Stream.
Clustering-The CURE Algorithm, Clustering in Non-Euclidean Spaces, Clustering for Streams and Parallelism

UNIT IV

Advertising on the Web-Issues in On-Line Advertising, On-Line Algorithms, The Matching Problem, The Adwords Problem, Adwords Implementation.
Recommendation Systems-A Model for Recommendation Systems, Content-Based Recommendations, Collaborative Filtering, Dimensionality Reduction, The NetFlix Challenge.

UNIT V

Mining Social-Network Graphs-Social Networks as Graphs, Clustering of Social-Network Graphs, Partitioning of Graphs, Simrank, Counting Triangles

TEXT BOOKS:

1. Jure Leskovec, Anand Rajaraman, Jeff Ullman, Mining of Massive Datasets, 3rd Edition.

REFERENCE BOOKS:

1. Jiawei Han & Micheline Kamber , Data Mining – Concepts and Techniques 3rd Edition Elsevier.
2. Margaret H Dunham, Data Mining Introductory and Advanced topics, PEA.
3. Ian H. Witten and Eibe Frank, Data Mining: Practical Machine Learning Tools and Techniques, Morgan Kaufmann.

**VAAGDEVI COLLEGE OF ENGINEERING
(AUTONOMOUS)
COMPUTER SCIENCE AND ENGINEERING**

**AGILE METHODOLOGIES
(PROFESSIONAL ELECTIVE - II)**

M.Tech CSE I Year I Sem.

**L T P C
3 0 0 3**

Prerequisites: Knowledge on basic Software Engineering, Object-Oriented Programming (OOP), Database Management Systems, Operating Systems and Project Management Fundamentals

Course Outcomes

1. Understand basic concepts of agile methods and extreme programming
2. Analyze real customer involvement and ubiquitous language
3. Discuss risk management and iteration planning
4. Summarize incremental requirements, refactoring, incremental design and architecture
5. Apply modular design principles, perform code refactoring, and implement architectural patterns in real-world software systems.

UNIT - I

Introduction Extreme Programming (XP) - Agile Development

Why Agile - Understanding Success, Beyond Deadlines, Importance of Organizational Success, Introduction to Agility How to Be Agile - Agile methods, Don't make your own method, Road to mastery Understanding XP (Extreme Programming) - XP life cycle, XP team, XP Concepts Adopting XP - Knowing whether XP is suitable, Implementing XP, assessing Agility Practicing XP - Thinking - Pair Programming, Energized work, Informative Workspace, Root cause Analysis, Retrospectives

UNIT - II

Collaborating

Trust, Sit together, Real customer involvement, Ubiquitous language, meetings, coding standards, Iteration demo, Reporting

UNIT - III

Releasing

Bugfree Release, Version Control, fast build, continuous integration, Collective ownership, Documentation

UNIT – IV

Planning

Version, Release Plan, Risk Management, Iteration Planning, Slack, Stories, Estimating

UNIT – V

Developing:

Incremental requirements, Customer tests, Test driven development, Refactoring, Incremental design and architecture, spike solutions, Performance optimization, Exploratory testing

TEXT BOOK:

1. The art of Agile Development, James Shore and Shane Warden, 11th Indian Reprint, O'Reilly,2018.

REFERENCES:

1. Learning Agile, Andrew Stellman and Jennifer Greene, O'Reilly, 4th Indian Reprint, 2018.
2. Practices of an Agile Developer, Venkat Subramaniam and Andy Hunt, SPD, 5th Indian Reprint, 2015.
3. Agile Project Management - Jim Highsmith, Pearson Low price Edition 2004.

**VAAGDEVI COLLEGE OF ENGINEERING
(AUTONOMOUS)
COMPUTER SCIENCE AND ENGINEERING**

**ADVANCED DATA STRUCTURES LAB
(LAB - I)**

M.Tech CSE I Year I Sem.

| | | | |
|----------|----------|----------|----------|
| L | T | P | C |
| 0 | 0 | 4 | 2 |

Prerequisites: A course on Computer Programming & Data Structures

Course Outcomes:

1. Implement and analyze various tree-based data structures for efficient insertion, deletion, and search operations.
2. Design and apply sorting algorithms to solve data organization and retrieval problems.
3. Apply hashing techniques for dictionary operations and implement efficient collision handling strategies for data storage and retrieval.
4. Implement and compare pattern matching algorithms for solving text search problems in large datasets.

List of Programs

1. Write a program to perform the following operations:
 - a) Insert an element into a binary search tree.
 - b) Delete an element from a binary search tree.
 - c) Search for a key element in a binary search tree.
2. Write a program for implementing the following sorting methods:
 - a) Merge sort b) Heap sort c) Quick sort
3. Write a program to perform the following operations:
 - a) Insert an element into a B- tree.
 - b) Delete an element from a B- tree.
 - c) Search for a key element in a B- tree.
4. Write a program to perform the following operations:
 - a) Insert an element into a Min-Max heap
 - b) Delete an element from a Min-Max heap
 - c) Search for a key element in a Min-Max heap
5. Write a program to perform the following operations:
 - a) Insert an element into a Leftist tree
 - b) Delete an element from a Leftist tree
 - c) Search for a key element in a Leftist tree
6. Write a program to perform the following operations:

- a) Insert an element into a binomial heap
 - b) Delete an element from a binomial heap.
 - c) Search for a key element in a binomial heap
7. Write a program to perform the following operations:
 - a) Insert an element into a AVL tree.
 - b) Delete an element from a AVL search tree.
 - c) Search for a key element in a AVL search tree.
 8. Write a program to perform the following operations:
 - a) Insert an element into a Red-Black tree.
 - b) Delete an element from a Red-Black tree.
 - c) Search for a key element in a Red-Black tree.
 9. Write a program to implement all the functions of a dictionary using hashing.
 10. Write a program for implementing Knuth-Morris-Pratt pattern matching algorithm.
 11. Write a program for implementing Brute Force pattern matching algorithm.
 12. Write a program for implementing Boyer pattern matching algorithm.

TEXT BOOKS:

1. Fundamentals of Data structures in C, E. Horowitz, S. Sahni and Susan Anderson Freed, 2nd Edition, Universities Press
2. Data Structures Using C – A.S. Tanenbaum, Y. Langsam, and M.J. Augenstein, PHI/Pearson education.
3. Introduction to Data Structures in C, Ashok Kamthane, 1st Edition, Pearson.

REFERENCES:

1. The C Programming Language, B.W. Kernighan, Dennis M.Ritchie, PHI/Pearson Education
2. 2. C Programming with problem solving, J.A. Jones & K. Harrow, Dreamtech Press
3. 3. Data structures: A Pseudocode Approach with C, R.F. Gilberg And B.A. Forouzan, 2nd Edition, Cengage Learning

**VAAGDEVI COLLEGE OF ENGINEERING
(AUTONOMOUS)
COMPUTER SCIENCE AND ENGINEERING**

**DATABASE PROGRAMMING WITH PL/SQL LAB
(LAB - II)**

M.Tech CSE I Year I Sem.

| | | | |
|----------|----------|----------|----------|
| L | T | P | C |
| 0 | 0 | 4 | 2 |

Prerequisites: C Programming, Database Management systems concepts ,

Course Outcomes:

1. Understand the structure of PL/SQL blocks, variables, control structures, and exception handling.
2. Apply cursors, functions, and procedures to manage and manipulate database information effectively.
3. Develop modular and reusable code using PL/SQL packages and bulk operations.
4. Design and implement triggers to enforce business rules and automate database processes.

List of Experiments:

1. Write a PL/SQL program using FOR loop to insert ten rows into a database table.
2. Given the table EMPLOYEE (EmpNo, Name, Salary, Designation, DeptID), write a cursor to select the five highest paid employees from the table.
3. Illustrate how you can embed PL/SQL in a high-level host language such as C/Java And demonstrates how a banking debit transaction might be done.
4. Given an integer i, write a PL/SQL procedure to insert the tuple (i, 'xxx') into a given relation.
5. Write a PL/SQL program to demonstrate Exceptions.
6. Write a PL/SQL program to demonstrate Cursors.
7. Write a PL/SQL program to demonstrate Functions.
8. Write a PL/SQL program to demonstrate Packages.
9. Write PL/SQL queries to create Procedures.
10. Write PL/SQL queries to create Triggers.

**VAAGDEVI COLLEGE OF ENGINEERING
(AUTONOMOUS)
COMPUTER SCIENCE AND ENGINEERING**

**DEEP LEARNING LAB
(LAB - II)**

M.Tech CSE I Year I Sem.

| | | | |
|----------|----------|----------|----------|
| L | T | P | C |
| 0 | 0 | 4 | 2 |

Pre-requisites: Machine Learning, AI, Python programming.

Course Outcomes:

1. Upon the Successful Completion of the Course, the Students would be able to:
2. Learn the Fundamental Principles of Deep Learning.
3. Identify the Deep Learning Algorithms for Various Types of Learning Tasks in various domains.
4. Implement Deep Learning Algorithms and Solve Real-world problems.

LIST OF EXPERIMENTS:

1. Setting up the Spyder IDE Environment and Executing a Python Program
2. Installing Keras, Tensorflow and Pytorch libraries and making use of them
3. Applying the Convolution Neural Network on computer vision problems
4. Image classification on MNIST dataset (CNN model with Fully connected layer)
5. Applying the Deep Learning Models in the field of Natural Language Processing
6. Train a sentiment analysis model on IMDB dataset, use RNN layers with LSTM/GRU notes
7. Applying the Autoencoder algorithms for encoding the real-world data
8. Applying Generative Adversial Networks for image generation and unsupervised tasks.

TEXT BOOKS:

1. Deep Learning by Ian Goodfellow, Yoshua Bengio and Aaron Courville, MIT Press.
2. The Elements of Statistical Learning by T. Hastie, R. Tibshirani, and J. Friedman, Springer.
3. Probabilistic Graphical Models. Koller, and N. Friedman, MIT Press.

REFERENCES:

1. Bishop, C, M., Pattern Recognition and Machine Learning, Springer, 2006.
2. Yegnanarayana, B., Artificial Neural Networks PHI Learning Pvt. Ltd, 2009.
3. Golub, G.,H., and Van Loan, C.,F., Matrix Computations, JHU Press,2013.
4. Satish Kumar, Neural Networks: A Classroom Approach, Tata McGraw-Hill Education, 2004.

Extensive Reading:

1. <http://www.deeplearning.net>
2. <https://www.deeplearningbook.org/>
3. <https://developers.google.com/machine-learning/crash-course/ml-intro>

4. www.cs.toronto.edu/~fritz/absps/imagenet.pdf
5. <http://neuralnetworksanddeeplearning.com/>

**VAAGDEVI COLLEGE OF ENGINEERING
(AUTONOMOUS)
COMPUTER SCIENCE AND ENGINEERING

NATURAL LANGUAGE PROCESSING LAB
(LAB - II)**

M.Tech CSE I Year I Sem.

**L T P C
0 0 4 2**

Prerequisites: Data structures, finite automata and probability theory

Course Outcomes:

1. Show sensitivity to linguistic phenomena and an ability to model them with formal grammars.
2. Able to manipulate probabilities, construct statistical models over strings and trees, and estimate parameters using supervised and unsupervised training methods.
3. Able to design, implement, and analyze NLP algorithms
4. Apply cognitive and statistical models to language tasks by combining rule-based and machine learning approaches to better understand sentence structure, meaning, and context.

List of Experiments

Implement the following using Python

1. Tokenization
2. Stemming
3. Stop word removal (a, the, are,..)
4. Word Analysis
5. Word Generation
6. Pos tagging
7. Morphology
8. chunking
9. N-Grams
10. N-Grams Smoothing

TEXT BOOKS:

1. Multilingual natural Language Processing Applications: From Theory to Practice – Daniel M. Bikel and Imed Zitouni, Pearson Publication
2. Natural Language Processing and Information Retrieval: Tanvier Siddiqui, U.S. Tiwary

REFERENCES:

1. Speech and Natural Language Processing - Daniel Jurafsky & James H Martin, Pearson Publications

**VAAGDEVI COLLEGE OF ENGINEERING
(AUTONOMOUS)
COMPUTER SCIENCE AND ENGINEERING**

RESEARCH METHODOLOGY & IPR

M.Tech CSE I Year I Sem.

| L | T | P | C |
|----------|----------|----------|----------|
| 2 | 0 | 0 | 2 |

Prerequisite: None

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

1. Formulate and analyze research problems by applying appropriate methods of investigation, data collection, and interpretation.
2. Evaluate and synthesize literature effectively while adhering to research ethics and avoiding plagiarism.
3. Develop technical writing skills for preparing research reports, proposals, and presentations.
4. Understand the nature of Intellectual Property and the process of patenting at national and international levels.
5. Analyze patent rights, licensing, technology transfer, and emerging developments in IPR including biological systems, software, and traditional knowledge.

UNIT-I

Meaning of research problem, Sources of research problem, Criteria Characteristics of a good research problem, Errors in selecting a research problem, Scope and objectives of research problem. Approaches of investigation of solutions for research problem, data collection, analysis, interpretation, Necessary instrumentations

UNIT-II

Effective literature studies approaches, analysis, Plagiarism, Research ethics

UNIT-III

Effective technical writing, how to write report, Paper Developing a Research Proposal, Format of research proposal, a presentation and assessment by a review committee

UNIT-IV

Nature of Intellectual Property: Patents, Designs, Trade and Copyright. Process of Patenting and Development: technological research, innovation, patenting, development. International Scenario: International cooperation on Intellectual Property. Procedure for grants of patents, Patenting under PCT.

UNIT-V

Patent Rights: Scope of Patent Rights. Licensing and transfer of technology. Patent information and databases. Geographical Indications. New Developments in IPR: Administration of Patent

System. New developments in IPR; IPR of Biological Systems, Computer Software etc. Traditional knowledge Case Studies, IPR and IITs.

TEXT BOOKS:

1. Stuart Melville and Wayne Goddard, “Research methodology: an introduction for science & engineering students”
2. C.R. Kothari, Research Methodology, methods & techniques, 2nd edition, New age International publishers

REFERENCES:

1. Ranjit Kumar, 2nd Edition, “Research Methodology: A Step by Step Guide for beginners”
2. Halbert, “Resisting Intellectual Property”, Taylor & Francis Ltd ,2007.
3. Mayall, “Industrial Design”, McGraw Hill, 1992.
4. Niebel, “Product Design”, McGraw Hill, 1974.
5. Asimov, “Introduction to Design”, Prentice Hall, 1962.
6. Robert P. Merges, Peter S. Menell, Mark A. Lemley, “Intellectual Property in New Technological Age”, 2016.
7. T. Ramappa, “Intellectual Property Rights Under WTO”, S. Chand, 2008.

**VAAGDEVI COLLEGE OF ENGINEERING
(AUTONOMOUS)
COMPUTER SCIENCE AND ENGINEERING**

**ADVANCED ALGORITHMS
(PC - III)**

M.Tech CSE I Year II Sem.

| | | | |
|----------|----------|----------|----------|
| L | T | P | C |
| 3 | 0 | 0 | 3 |

Pre-Requisites: UG level course in Algorithm Design and Analysis

Course Outcomes: After completion of course, students would be able to:

1. Analyze the complexity/performance of different algorithms.
2. Determine the appropriate data structure for solving a particular set of problems.
3. Categorize the different problems in various classes according to their complexity.
4. Use dynamic programming and modular techniques to solve graph and algebraic problems.
5. Apply linear programming and analyze NP-complete problems using modern algorithms.

UNIT – I

Sorting:

Review of various sorting algorithms, topological sorting Graph:

Definitions and Elementary Algorithms: Shortest path by BFS, shortest path in edge-weighted case (Dijkstra's), depth-first search and computation of strongly connected components, emphasis on correctness proof of the algorithm and time/space analysis, example of amortized analysis.

UNIT – II

Matroids:

Introduction to greedy paradigm, algorithm to compute a maximum weight maximal independent set.

Application to MST.

Graph Matching:

Algorithm to compute maximum matching. Characterization of maximum matching by augmenting paths, Edmond's Blossom algorithm to compute augmenting path.

UNIT - III

Flow-Networks:

Maxflow-mincut theorem, Ford-Fulkerson Method to compute maximum flow, Edmond-Karp maximum-flow algorithm.

Matrix Computations:

Strassen's algorithm and introduction to divide and conquer paradigm, inverse of a triangular matrix, relation between the time complexities of basic matrix operations, LUP-decomposition.

UNIT - IV

Shortest Path in Graphs:

Floyd-Warshall algorithm and introduction to dynamic programming paradigm. More examples of dynamic programming.

Modulo Representation of integers/polynomials:

Chinese Remainder Theorem, Conversion between base-representation and modulo-representation.

Extension to polynomials. Application: Interpolation problem.

Discrete Fourier Transform (DFT):

In complex field, DFT in modulo ring. Fast Fourier Transform algorithm. Schonhage-Strassen Integer Multiplication algorithm.

UNIT - V

Linear Programming: Geometry of the feasibility region and Simplex algorithm NP-completeness: Examples, proof of NP-hardness and NP-completeness.

Recent Trends in problem solving paradigms using recent searching and sorting techniques by applying recently proposed data structures.

REFERENCES:

1. Cormen, Leiserson, Rivest, Stein, "Introduction to Algorithms".
2. Aho, Hopcroft, Ullman "The Design and Analysis of Computer Algorithms".
3. Kleinberg and Tardos. "Algorithm Design".

**VAAGDEVI COLLEGE OF ENGINEERING
(AUTONOMOUS)
COMPUTER SCIENCE AND ENGINEERING
ADVANCED COMPUTER ARCHITECTURE
(PC - IV)**

M.Tech CSE I Year II Sem.

**L T P C
3 0 0 3**

Prerequisites: Computer Organization

Course Outcomes:

1. Understand computational models and fundamental computer architectures.
2. Explain parallel computer models and scalable architecture principles.
3. Analyze pipelining and superscalar processor designs.
4. Evaluate multiprocessor systems, interconnects, and synchronization mechanisms.
5. Apply SIMD and vector processing concepts, including CM-5 architecture.

UNIT - I

Theory of Parallelism, Parallel computer models, The State of Computing, Multiprocessors and Multicomputers, Multivector and SIMD Computers, PRAM and VLSI models, Architectural development tracks, Program and network properties, Conditions of parallelism, Program partitioning and Scheduling, Program flow Mechanisms, System interconnect Architectures.

UNIT - II

Principles of Scalable performance, Performance metrics and measures, Parallel Processing applications, Speed up performance laws, Scalability Analysis and Approaches, Hardware Technologies, Processes and Memory Hierarchy, Advanced Processor Technology, Superscalar and Vector Processors

UNIT - III

Shared-Memory Organizations, Sequential and weak consistency models, Pipelining and superscalar techniques, Linear Pipeline Processors, Non-Linear Pipeline Processors, Instruction Pipeline design, Arithmetic pipeline design, superscalar pipeline design.

UNIT - IV

Parallel and Scalable Architectures, Multiprocessors and Multicomputers, Multiprocessor system interconnects, cache coherence and synchronization mechanism, Three Generations of Multicomputers, Message-passing Mechanisms, Multivector and SIMD computers.

UNIT - V

Vector Processing Principles, Multivector Multiprocessors, Compound Vector processing, SIMD computer Organizations, The connection machine CM-5.

TEXT BOOK:

1. Advanced Computer Architecture, Kai Hwang, 2nd Edition, Tata McGraw Hill Publishers.

REFERENCES:

1. Computer Architecture, J.L. Hennessy and D.A. Patterson, 4th Edition, ELSEVIER.
2. Advanced Computer Architectures, S.G.Shiva, Special Indian edition, CRC, Taylor & Francis.
3. Introduction to High Performance Computing for Scientists and Engineers, G. Hager and G. Wellein, CRC Press, Taylor & Francis Group.
4. Advanced Computer Architecture, D. Sima, T. Fountain, P. Kacsuk, Pearson education.
5. Computer Architecture, B. Parhami, Oxford Univ. Press.

**VAAGDEVI COLLEGE OF ENGINEERING
(AUTONOMOUS)
COMPUTER SCIENCE AND ENGINEERING**

**ENTERPRISE CLOUD CONCEPTS
(PROFESSIONAL ELECTIVE - III)**

M.Tech CSE I Year II Sem.

| | | | |
|----------|----------|----------|----------|
| L | T | P | C |
| 3 | 0 | 0 | 3 |

Prerequisites: Knowledge on significance of cloud computing and its fundamental concepts and models.

Course Outcomes:

1. Understand importance of cloud architecture
2. Illustrating the fundamental concepts of cloud security
3. Analyze various cloud computing mechanisms
4. Understanding the architecture and working of cloud computing.
5. Understanding Tuning Methodology, Contract Management in the Cloud
Cloud-Instigated IT Transformation.

Unit - I

Understanding Cloud Computing:

Origins and influences, Basic Concepts and Terminology, Goals and Benefits, Risks and Challenges.

Fundamental Concepts and Models:

Roles and Boundaries, Cloud Characteristics, Cloud Delivery Models, Cloud Deployment Models.

Unit - II

Cloud-Enabling Technology:

Broadband Networks and Internet Architecture, Data Center Technology, Virtualization Technology

CLOUD COMPUTING MECHANISMS:

Cloud Infrastructure Mechanisms: Logical Network Perimeter, Virtual Server, Cloud Storage Device, Cloud Usage Monitor, Resource Replication

Unit - III

Cloud Management Mechanisms: Remote Administration System, Resource Management System,

SLA Management System, Billing Management System, Case Study Example

Cloud Computing Architecture

Fundamental Cloud Architectures: Workload Distribution Architecture, Resource Pooling Architecture, Dynamic Scalability Architecture, Elastic Resource Capacity Architecture,

Service Load Balancing Architecture, Cloud Bursting Architecture, Elastic Disk Provisioning Architecture, Redundant Storage Architecture, Case Study Example

Unit - IV

Cloud-Enabled Smart Enterprises

Introduction, Revisiting the Enterprise Journey, Service-Oriented Enterprises, Cloud Enterprises,

Smart Enterprises, The Enabling Mechanisms of Smart Enterprises

Cloud-Inspired Enterprise Transformations

Introduction, The Cloud Scheme for Enterprise Success, Elucidating the Evolving Cloud Idea, Implications of the Cloud on Enterprise Strategy, Establishing a Cloud-Incorporated Business Strategy

UNIT-V

Transitioning to Cloud-Centric Enterprises

The Tuning Methodology, Contract Management in the Cloud

Cloud-Instigated IT Transformations

Introduction, Explaining Cloud Infrastructures, A Briefing on Next-Generation Services, Service Infrastructures, Cloud Infrastructures, Cloud Infrastructure Solutions, Clouds for Business Continuity, The Relevance of Private Clouds, The Emergence of Enterprise Clouds

TEXT BOOKS:

1. Erl Thomas, Puttini Ricardo, Mahmood Zaigham, Cloud Computing: Concepts, Technology & Architecture 1st Edition,
2. Pethuru Raj, Cloud Enterprise Architecture, CRC Press

REFERENCE:

1. James Bond, The Enterprise Cloud, O'Reilly Media, Inc.

**VAAGDEVI COLLEGE OF ENGINEERING
(AUTONOMOUS)
COMPUTER SCIENCE AND ENGINEERING

CYBER SECURITY
(PROFESSIONAL ELECTIVE - III)**

M.Tech CSE I Year II Sem.

**L T P C
3 0 0 3**

Prerequisites: Data Communications and Computer Networks, Network Security and Cryptography

Course Outcomes:

1. Analyze and evaluate the cyber security needs of an organization.
2. Understand Cyber Security Regulations and Roles of International Law
3. Identify various security challenges phased by mobile devices.
4. Design and develop a security architecture for an organization.
5. Understand fundamental concepts of data privacy attacks

UNIT -I

Introduction to Cyber Security: Basic Cyber Security Concepts, layers of security, Vulnerability, threat, Harmful acts, Internet Governance – Challenges and Constraints, Computer Criminals, CIA Triad, Assets and Threat, motive of attackers, active attacks, passive attacks, Software attacks, hardware attacks, Cyber Threats-Cyber Warfare, Cyber Crime, Cyber terrorism, Cyber Espionage, etc., Comprehensive Cyber Security Policy.

UNIT - II

Cyberspace and the Law & Cyber Forensics: Introduction, Cyber Security Regulations, Roles of International Law. The INDIAN Cyberspace, National Cyber Security Policy.

Introduction, Historical background of Cyber forensics, Digital Forensics Science, The Need for Computer Forensics, Cyber Forensics and Digital evidence, Forensics Analysis of Email, Digital Forensics Lifecycle, Forensics Investigation, Challenges in Computer Forensics

UNIT - III

Cybercrime: Mobile and Wireless Devices: Introduction, Proliferation of Mobile and Wireless Devices, Trends in Mobility, Credit card Frauds in Mobile and Wireless Computing Era, Security Challenges Posed by Mobile Devices, Registry Settings for Mobile Devices, Authentication service Security, Attacks on Mobile/Cell Phones, Organizational security Policies and Measures in Mobile Computing Era, Laptops.

UNIT- IV

Cyber Security: Organizational Implications: Introduction, cost of cybercrimes and IPR issues, web threats for organizations, security and privacy implications, social media

marketing: security risks and perils for organizations, social computing and the associated challenges for organizations

UNIT - V

Privacy Issues: Basic Data Privacy Concepts: Fundamental Concepts, Data Privacy Attacks, Data linking and profiling, privacy policies and their specifications, privacy policy languages, privacy in different domains- medical, financial, etc

Cybercrime: Examples and Mini-Cases

Examples: Official Website of Maharashtra Government Hacked, Indian Banks Lose Millions of Rupees, Parliament Attack, Pune City Police Bust Nigerian Racket, e-mail spoofing instances. Mini-Cases: The Indian Case of online Gambling, An Indian Case of Intellectual Property Crime, Financial Frauds in Cyber Domain.

TEXT BOOKS:

1. Nina Godbole and Sunit Belpure, Cyber Security Understanding Cyber Crimes, Computer Forensics and Legal Perspectives, Wiley
2. B.B. Gupta, D.P. Agrawal, Haoxiang Wang, Computer and Cyber Security: Principles, Algorithm, Applications, and Perspectives, CRC Press, ISBN 9780815371335,2018.

REFERENCES:

1. Cyber Security Essentials, James Graham, Richard Howard and Ryan Otson, CRC Press.
2. Introduction to Cyber Security, Chwan-Hwa(john) Wu,J. David Irwin, CRC Press T&FGroup.

**VAAGDEVI COLLEGE OF ENGINEERING
(AUTONOMOUS)
COMPUTER SCIENCE AND ENGINEERING

PARALLEL COMPUTING
(PROFESSIONAL ELECTIVE - III)**

M.Tech CSE I Year II Sem.

| | | | |
|----------|----------|----------|----------|
| L | T | P | C |
| 3 | 0 | 0 | 3 |

Prerequisites:

Computer Organization & Architecture ,Operating Systems and Programming for problem solving

Course Outcomes :

1. Understand the concepts of parallel architectures
2. Select the data structures that efficiently model the information in a problem.
3. Develop an efficient parallel algorithm to solve it.
4. Implement an efficient and correct code to solve it, analyse its performance
5. Solve graph problems using DFS, BFS, Prim's, and Dijkstra's algorithms.

Unit I

Parallel Computing: Introduction, Motivation and scope - Parallel Programming Platforms – Basic Communication Operations

Unit II

Principles of Parallel Algorithm Design - Analytical Modelling of Parallel Programs

Unit III

Programming using Message Passing Paradigm (MPI) – Programming Shared Address Space Platforms (PThreads)

Unit IV

Dense Matrix Algorithms (Matrix-Vector Multiplication, Matrix-Matrix Multiplication) – Sorting Algorithms (Insertion Sort, Bubble Sort, Quick Sort, Bucket Sort, Enumeration Sort, Radix Sort)

Unit V

Graph Algorithms (Minimum Spanning Tree: Prim's Algorithm - Single-Source Shortest Paths: Dijkstra's Algorithm) Search Algorithms (DFS, BFS)

TEXT BOOK:

1. Introduction to Parallel Computing, Second Edition, Ananth Grama, George Karypis, Vipin Kumar, Anshul Gupta, Addison-Wesley, 2003, ISBN: 0201648652

REFERENCES:

1. Parallel Computing – Theory and Practice, Second Edition, Michael J. Quinn, Tata McGraw-Hill Edition.
2. Parallel Computers – Architectures and Programming, V. Rajaraman, C. Siva Ram Murthy, PHI.

**VAAGDEVI COLLEGE OF ENGINEERING
(AUTONOMOUS)
COMPUTER SCIENCE AND ENGINEERING**

**LARGE LANGUAGE MODELS
(PROFESSIONAL ELECTIVE - III)**

M.Tech CSE I Year II Sem.

| | | | |
|----------|----------|----------|----------|
| L | T | P | C |
| 3 | 0 | 0 | 3 |

Prerequisites:

Machine Learning and Natural Language Processing.

Course Outcomes (COs)

1. Understand the architecture and inner workings of transformer-based LLMs.
2. Apply prompt engineering and fine-tuning techniques for domain-specific tasks.
3. Evaluate LLM performance using standard metrics and benchmarks.
4. Identify challenges in LLM training, deployment, and scaling.
5. Analyze ethical, legal, and societal implications of LLM usage.

UNIT 1 – Foundations of Large Language Models

Introduction to LLMs: Definition, scope, and historical evolution from statistical NLP to transformers. The Transformer architecture: Attention mechanisms, self-attention, multi-head attention. Pretraining objectives: Masked language modeling (MLM), Causal language modeling (CLM). Evolution of LLMs: BERT, GPT series, T5, LLaMA, Mistral.

UNIT 2 – Training and Fine-Tuning LLMs

Pretraining datasets and tokenization: BPE, SentencePiece, WordPiece. Fine-tuning approaches: Full fine-tuning, LoRA, adapters, instruction tuning. Domain adaptation and few-shot/zero-shot learning. Data augmentation for LLMs and prompt-based tuning.

UNIT 3 – Prompt Engineering and Applications

Principles of prompt design: Zero-shot, few-shot, and chain-of-thought prompting. System prompts, role prompting, and context length optimization.

Use cases: Text generation, summarization, code generation, question answering, chatbots.

Tools & frameworks: Lang Chain, Llama Index, Hugging Face Transformers.

UNIT 4 – Evaluation and Deployment of LLMs

Evaluation metrics: Perplexity, BLEU, ROUGE, METEOR, human evaluation.

Benchmark datasets: GLUE, SuperGLUE, HELM, BIG-bench.

Deployment strategies: API-based deployment, on-prem deployment, inference optimization.

Scaling and latency considerations; quantization and pruning for LLMs.

UNIT 5 – Ethics, Safety, and Future Directions

Bias, fairness, and toxicity in LLMs. Hallucination problem and mitigation techniques. Legal and regulatory issues: Copyright, data privacy, AI Act. Trends in LLM research: Multimodal LLMs, retrieval-augmented generation (RAG), open-source LLM ecosystems.

TEXT BOOKS:

1. Vaswani, A. et al. (2017) Attention Is All You Need – NIPS Conference Paper.
2. Lewis, P. et al. (2021) Language Models are Few-Shot Learners – OpenAI Research Paper.
3. Tunstall, L., von Werra, L., & Wolf, T. (2022) Natural Language Processing with Transformers – O'Reilly Media.

REFERENCE BOOKS:

1. Bommasani, R. et al. (2021) On the Opportunities and Risks of Foundation Models – Stanford CRFM.
2. Jurafsky, D., & Martin, J. H. (2023) Speech and Language Processing (3rd Edition draft) – Pearson.
3. Mollick, E., & Mollick, L. (2024) Co-Intelligence: Living and Working with AI – Little, Brown Spark.
4. Hugging Face Documentation – <https://huggingface.co/docs/>

**VAAGDEVI COLLEGE OF ENGINEERING
(AUTONOMOUS)
COMPUTER SCIENCE AND ENGINEERING

BIOINFORMATICS
(PROFESSIONAL ELECTIVE - IV)**

M.Tech CSE I Year II Sem.

**L T P C
3 0 0 3**

Prerequisites: Knowledge on concepts of bioinformatics and biological motivations of sequence analysis

Course Outcomes:

1. Understand the Central Dogma & XML (Bio XML) for Bioinformatics.
2. Analyze Perl (Bioperl) for Bioinformatics.
3. Illustrate Databases technology, architecture and its interfaces.
4. Understand Sequence Alignment Algorithms.
5. **Understand the fundamentals of phylogenetic analysis.**

UNIT -I

The Central Dogma & XML (Bio XML) for Bioinformatics: Watson's definition, information flow, from data to knowledge, Convergence, the organization of DNA, the organization of Proteins, Introduction, Differences between HTML and XML, fundamentals of XML, fundamentals of XML namespaces. Introduction to DTDs, Document type Declarations, Declaring elements, declaring attributes, working with entities XML Schemas, Essential Concepts, working with simple types, working with complex types, Basic namespaces issues.

UNIT -II

Perl (Bioperl) for Bioinformatics: Representing sequence data, program to store a DNA sequence, concatenating DNA fragments, Transcription, Calculating the reverse complement in Perl, Proteins, files, reading proteins in files, Arrays, Flow control, finding motifs, counting Nucleotides, exploding strings into arrays, operating on strings, writing to files, subroutines and bugs.

UNIT -III

Databases: Flat file, Relational, object oriented databases, object Relational and Hypertext, Data life cycle, Database Technology, Database Architecture, Database Management Systems and Interfaces.

UNIT -IV

Sequence Alignment Algorithms: Biological motivations of sequence analysis, the models for sequence analysis and their biological motivation, global alignment, local alignment, End free-space alignment and gap penalty, Sequence Analysis tools and techniques.

UNIT -V

Phylogenetic Analysis: Introduction, methods of Phylogenetic analysis, distance methods, the neighbor- Joining (NJ) method, The Fitch/ Margoliash method, character-based methods, Other methods, Tree evaluation and problems in phylogenetic analysis, Clustering, Protein structure visualization and Protein structure prediction.

TEXT BOOKS:

1. S.C. Rastogi, N. Mendiratta, “Bioinformatics Methods and Applications”, CBS publications, 2004
2. James D. Tisdall, “Beginning Perl for Bioinformatics” O’Reilly media, 1st Edition, 2001

REFERENCE BOOKS:

1. D.R. Westhead, J.H. Parish, “Bioinformatics” Viva books private limited, New Delhi (2003)
2. Att Wood, “Bioinformatics” Pearson Education, 2004
3. Bryan Bergeron, M.D, “Bioinformatics Computing” Pearson Education, 2003

**VAAGDEVI COLLEGE OF ENGINEERING
(AUTONOMOUS)
COMPUTER SCIENCE AND ENGINEERING**

**AD-HOC AND SENSOR NETWORKS
(PROFESSIONAL ELECTIVE - IV)**

M.Tech CSE I Year II Sem.

**L T P C
3 0 0 3**

Prerequisites: Computer Networks

Course Outcomes

1. Understand the fundamentals, characteristics, applications, and challenges of MANETs and Wireless Sensor Networks (WSNs).
2. Analyze and implement various routing algorithms in MANETs, including proactive, reactive, hybrid, and position-based approaches.
3. Examine and implement data transmission strategies in MANETs, including broadcast, rebroadcasting, and multicasting protocols.
4. Understand geocasting techniques and TCP protocol issues over MANETs.
5. Evaluate lower and upper layer issues in WSNs, including physical, MAC, link, routing, transport, and application layers, and adapt protocols to dynamic network environments.

UNIT - I

Introduction to Ad Hoc Networks

Characteristics of MANETs, Applications of MANETs and Challenges of MANETs. Routing in MANETs

Criteria for classification, Taxonomy of MANET routing algorithms, Topology-based routing algorithms-Proactive: DSDV, WRP; Reactive: DSR, AODV, TORA; Hybrid: ZRP; Position-based routing algorithms-Location Services-DREAM, Quorum-based, GLS; Forwarding Strategies

Greedy Packet, Restricted Directional Flooding-DREAM, LAR; Other routing algorithms-QoS Routing, CEDAR.

UNIT - II

Data Transmission

Broadcast Storm Problem, Rebroadcasting Schemes-Simple-flooding, Probability-based Methods, Area-based Methods, Neighbour Knowledge-based: SBA, Multipoint Relaying, AHBP. Multicasting: Tree-based: AMRIS, MAODV; Mesh-based: ODMRP, CAMP; Hybrid: AMRoute, MCEDAR.

UNIT - III

Geocasting

Data-transmission Oriented-LBM; Route Creation Oriented-GeoTORA, MGR.

TCP over Ad Hoc TCP protocol overview, TCP and MANETs, Solutions for TCP over Ad hoc

UNIT - IV

Basics of Wireless Sensors and Lower Layer Issues

Applications, Classification of sensor networks, Architecture of sensor network, Physical layer, MAC layer, Link layer, Routing Layer.

UNIT - V

Upper Layer Issues of WSN

Transport layer, High-level application layer support, Adapting to the inherent dynamic nature of WSNs.

TEXT BOOKS:

1. Ad Hoc and Sensor Networks – Theory and Applications, Carlos Corderio Dharma P. Aggarwal, World Scientific Publications, March 2006, ISBN – 981-256-681-3
2. Wireless Sensor Networks: An Information Processing Approach, Feng Zhao, Leonidas Guibas, Elsevier Science, ISBN – 978-1-55860-914-3 (Morgan Kaufman)

REFERENCES:

1. C. Siva Ram Murthy, B.S. Manoj Ad Hoc Wireless Networks: Architectures and Protocols
2. Taieb Znati Kazem Sohraby, Daniel Minoli, Wireless Sensor Networks: Technology, Protocols and Applications, Wiley.

**VAAGDEVI COLLEGE OF ENGINEERING
(AUTONOMOUS)
COMPUTER SCIENCE AND ENGINEERING

ROBOTIC PROCESS AUTOMATION
(PROFESSIONAL ELECTIVE - IV)**

M.Tech CSE I Year II Sem.

**L T P C
3 0 0 3**

Prerequisites: Basic Knowledge on Programming Fundamentals, Database Management system, Software Engineering and Business Process Understanding

Course Outcomes:

1. Describe RPA, where it can be applied and how it's implemented.
2. Identify and understand Web Control Room and Client Introduction
3. Understand how to handle various devices and the workload
4. Understand Bot creators, Web recorders and task editors
5. Understand Terminal Emulator Command and Error Handling Command

Unit I

Introduction to Robotic Process Automation & Bot Creation Introduction to RPA and Use cases – Automation Anywhere Enterprise Platform – Advanced features and capabilities – Ways to create Bots

Unit II

Web Control Room and Client Introduction - Features Panel - Dashboard (Home, Bots, Devices, Audit, Workload, Insights) - Features Panel – Activity (View Tasks in Progress and Scheduled Tasks) - Bots (View Bots Uploaded and Credentials)

Unit III

Devices (View Development and Runtime Clients and Device Pools) - Workload (Queues and SLA Calculator) - Audit Log (View Activities Logged which are associated with Web CR) - Administration (Configure Settings, Users, Roles, License and Migration) - Demo of Exposed API's – Conclusion – Client introduction and Conclusion.

Unit IV

Bot Creator Introduction – Recorders – Smart Recorders – Web Recorders – Screen Recorders - Task Editor – Variables - Command Library – Loop Command – Excel Command – Database Command - String Operation Command - XML Command

Unit V

Terminal Emulator Command - PDF Integration Command - FTP Command - PGP Command - Object Cloning Command - Error Handling Command - Manage Windows Control Command - Workflow Designer - Report Designer

TEXT BOOKS:

1. Learning Robotic Process Automation: Create Software robots and automate business processes with the leading RPA tool - UiPath: Create Software robots. with the leading RPA tool – UiPath Kindle Edition.

REFERENCES:

1. Robotic Process Automation A Complete Guide - 2020 Edition Kindle Edition.

**VAAGDEVI COLLEGE OF ENGINEERING
(AUTONOMOUS)
COMPUTER SCIENCE AND ENGINEERING**

**GENERATIVE AI
(PROFESSIONAL ELECTIVE - IV)**

M.Tech CSE I Year II Sem.

| | | | |
|----------|----------|----------|----------|
| L | T | P | C |
| 3 | 0 | 0 | 3 |

Prerequisite: AI, Machine Learning.

Course Outcomes

1. Demonstrate knowledge of AI foundations, generative models, and advanced neural architectures.
2. Apply generative AI techniques to create solutions for text, image, video, and multimodal tasks.
3. Design, fine-tune, and optimize Large Language Models for specific applications.
4. Evaluate ethical, social, and legal implications of Generative AI deployments and propose mitigation strategies.
5. Gain practical experience with state-of-the-art framework and tools.

UNIT 1

Foundations of AI and Generative Models

Introduction and historical evolution to Artificial Intelligence (AI), Machine Learning (ML), Natural

Language Processing (NLP) and Deep Learning (DL), Structure of Artificial Neural Networks (ANNs), Mathematical and computational foundations of generative modeling, Overview of generative models and their applications across various domains; Importance of Generative AI in modern applications, Transfer learning and in advancing Generative AI

UNIT 2

Advanced Neural Architectures for Generative AI

Variational Autoencoders (VAEs): principles and applications, Generative Adversarial Networks (GANs): architecture and working principles; Transformer architecture and attention mechanisms (in detail); Long Short-Term Memory Networks (LSTMs) and the limitations of traditional RNNs/LSTMs, Advanced Transformer architectures and techniques, Pre-training and transfer learning strategies for generative models

UNIT 3

Large Language Models and Prompt Engineering

Overview of Large Language Models (LLMs), GPT architecture, variants, and working principles, Pr training and fine-tuning GPT models for applications (e.g., chatbots, text generation), Case study: GPT-based customer support chatbot, BERT architecture, pre-training objectives, and fine-tuning, Prompt Engineering: Designing effective prompts, controlling model behavior, and improving output quality, Fine-tuning language models for creative writing and chatbot development

UNIT 4

Multi-Agent Systems and Generative AI Applications

Introduction to Multi-Agent Systems (MAS), Types of agents: reactive, deliberative, hybrid, and learning agents, Multi-agent collaboration and orchestration for generative tasks, Use cases: autonomous research assistants, cooperative creative generation, distributed problem-solving, Frameworks and tools: AutoGen, CrewAI, Hugging GPT for LLM-powered multi-agent systems, Generative AI applications: Art, Creativity, Image/Video generation, Music composition, Healthcare,

Finance, Real-world case studies and deployment challenges

UNIT 5

Frameworks, Multimodal Applications, and Ethics

LangChain framework: components and LLM application development, Retrieval-Augmented Generation (RAG), Embeddings, Indexing networks, and Vector databases, Generative AI across modalities: Text, Code, Image, and Video generation, Image and Video generation using GANs and VAEs, Multimodal Generative AI: integration and training strategies, Ethical considerations: bias, fairness, trust, and responsible AI deployment, Social and legal implications of Generative AI, Risk mitigation strategies and real-world ethical case studies

TEXT BOOKS

1. Altaf Rehmani, Generative AI for Everyone: Understanding the Essentials and Applications of This Breakthrough Technology.
2. Charu C. Aggarwal, Neural Networks and Deep Learning: A Textbook. Joseph Babcock, Raghav Bali, Generative AI with Python and TensorFlow 2, 2024.

REFERENCE BOOKS

1. Josh Kalin, Generative Adversarial Networks Cookbook.
2. Jesse Sprinter, Generative AI in Software Development: Beyond the Limitations of Traditional Coding, 2024.

ONLINE REFERENCES

1. Fabian Gloeckle et al., Better & Faster Large Language Models via Multi-token Prediction, arXiv:2404.19737v1, 2024. Vaswani et al., Attention Is All You Need, NeurIPS 2017.

**VAAGDEVI COLLEGE OF ENGINEERING
(AUTONOMOUS)
COMPUTER SCIENCE AND ENGINEERING**

**ADVANCED ALGORITHMS LAB
(LAB - III)**

M.Tech CSE I Year II Sem.

| | | | |
|----------|----------|----------|----------|
| L | T | P | C |
| 0 | 0 | 4 | 2 |

Pre-Requisites: UG level course in Algorithm Design and Analysis

Course Outcomes:

1. Implement and analyze classical algorithms for optimization and problem-solving, such as assignment and knapsack problems.
2. Apply divide-and-conquer and greedy strategies to solve computational problems efficiently, including long integer multiplication.
3. Design and implement string matching and searching algorithms, including KMP, Rabin-Karp, and Horspool algorithms.
4. Solve matrix and graph-related problems using methods like Gaussian elimination, LU decomposition, Warshall algorithm, and max-flow problem.

List of Experiments

1. Implement assignment problem using Brute Force method
2. Perform multiplication of long integers using divide and conquer method.
3. Implement a solution for the knapsack problem using the Greedy method.
4. Implement Gaussian elimination method.
5. Implement LU decomposition
6. Implement Warshall algorithm
7. Implement the Rabin Karp algorithm.
8. Implement the KMP algorithm.
9. Implement Harspool algorithm
10. Implement max-flow problem.

TEXT BOOK:

1. Design and Analysis of Algorithms, S.Sridhar, OXFORD University Press

REFERENCES:

1. Introduction to Algorithms, second edition, T.H. Cormen, C.E. Leiserson, R.L. Rivest and C.Stein,PHIPvt. Ltd./ Pearson Education.
2. Fundamentals of Computer Algorithms, Ellis Horowitz, Satraj Sahni and Rajasekharam, Universities Press.
3. Design and Analysis of algorithms, Aho, Ullman and Hopcroft, Pearson education.

**VAAGDEVI COLLEGE OF ENGINEERING
(AUTONOMOUS)
COMPUTER SCIENCE AND ENGINEERING**

**ENTERPRISE CLOUD CONCEPTS LAB
(LAB -IV)**

M.Tech CSE I Year II Sem.

| | | | |
|----------|----------|----------|----------|
| L | T | P | C |
| 0 | 0 | 4 | 2 |

Prerequisites:

Basic knowledge of networking, operating systems, virtualization, and programming

Course Outcomes:

1. Understand importance of cloud architecture
2. Illustrating the fundamental concepts of cloud security
3. Analyze various cloud computing mechanisms
4. Understanding the architecture and working of cloud computing.

List of Experiments:

1. Install Virtualbox/VMware Workstation with different flavors of linux or windows OS on top of windows7 or 8.
2. Install a C compiler in the virtual machine created using virtual box and execute Simple Programs
3. Install Google App Engine. Create a hello world app and other simple web applications using python/java..
4. Find a procedure to transfer the files from one virtual machine to another virtual machine.
5. Find a procedure to launch virtual machine using trystack (Online Openstack Demo Version)
6. Install Hadoop single node cluster and run simple applications like word count.

E-Resources:

1. <https://www.iitk.ac.in/nt/faq/vbox.html>
2. <https://www.google.com/urlsa=t&rct=j&q=&esrc=s&source=web&cd=&ved=2ahUK EwjqrNG0za73Ah Zt1YBHZ21DWEQFnoECAMQAQ&url=http%3A%2F%2Fwww.cs.columbia.edu%2F~sedwards%2Fclasses%2F2015%2F1102-fall%2Flinuxvm.pdf&usg=AOvVaw3xZPuF5xVgk-AQnBRsTtHz>
3. <https://www.cloudsimtutorials.online/cloudsim/>
4. <https://edwardsamuel.wordpress.com/2014/10/25/tutorial-creating-openstack-instance-in-trystack/>
5. <https://www.edureka.co/blog/install-hadoop-single-node-hadoop-cluster>

**VAAGDEVI COLLEGE OF ENGINEERING
(AUTONOMOUS)
COMPUTER SCIENCE AND ENGINEERING**

**CYBER SECURITY LAB
(LAB -IV)**

M.Tech CSE I Year II Sem.

| | | | |
|----------|----------|----------|----------|
| L | T | P | C |
| 0 | 0 | 4 | 2 |

Prerequisites

1. A course on "Network Security and Cryptography".

Course Outcomes:

1. Get the skill to identify cyber threats/attacks.
2. Get the knowledge to solve security issues in day to day life.
3. Acquire and analyze digital evidence from disk , mamory and network using forensic tools like Autopsy,FTK imager and Network Miner.
4. Interpret findings and prepare comprehensive reports to support invesitgations.

List of Experiments

1. Perform an Experiment for port scanning with NMAP.
2. Setup a honeypot and monitor the honeypot on the network
3. Install Jcrpt /Cryptool tool (or any other equivalent) and demonstrate Asymmetric, Symmetric crypto algorithm, Hash and Digital/PKI signatures.
4. Generate minimum 10 passwords of length 12 characters using open SSL command
5. Perform practical approach to implement Foot printing-Gathering target information using Dmitry-Dmagic, UAtester.
6. Working with sniffers for monitoring network communication (Wireshark).
7. Use Snort to perform real time traffic analysis and packet logging.
8. Perform email analysis using Autopsy tool.
9. Perform Registry analysis and get boot time logging using process monitor tool
10. Perform File type detection using Autopsy tool
11. Perform Memory capture and analysis using FTK imager tool
12. Perform Network analysis using the Network Miner tool

TEXT BOOKS

1. Real Digital Forensics for Handheld Devices, E. P. Dorothy, Auerback Publications, 2013.
2. Handbook of Digital Forensics and Investigation, E. Casey, Academic Press, 2010

REFERENCES:

1. The Basics of Digital Forensics: The Primer for Getting Started in Digital Forensics, J. Sammons, Syngress Publishing, 2012.
2. Malware Forensics Field Guide for Windows Systems: Digital Forensics Field Guides, C. H. Malin, E. Casey and J. M. Aquilina, Syngress, 2012

3. The Best Damn Cybercrime and Digital Forensics Book Period, J. Wiles and A.Reyes, Syngress, 2007.

**VAAGDEVI COLLEGE OF ENGINEERING
(AUTONOMOUS)
COMPUTER SCIENCE AND ENGINEERING**

**PARALLEL COMPUTING LAB
(LAB -IV)**

M.Tech CSE I Year II Sem.

| L | T | P | C |
|----------|----------|----------|----------|
| 0 | 0 | 4 | 2 |

Prerequisites

Computer Organization & Architecture, Operating Systems Programming for problem solving

Course Outcomes

1. Understand the principles of shared-memory and distributed-memory parallel programming models.
2. Apply OpenMP parallel applications using OpenMP for shared-memory architectures.
3. Develop multithreaded applications using Pthreads with proper synchronization.
4. Analyze and compare speedup, scalability, and efficiency of parallel implementations across MPI, OpenMP, and Pthreads.

List of Programs:

1. Design a parallel program to implement Matrix-Vector and Matrix-Matrix Multiplication using MPI library.
2. Design a parallel program to implement Bubble Sort using OpenMP and Pthread Programming Constructs.
3. Design a parallel program to implement Quick Sort using OpenMP and Pthread Programming Constructs.
4. Design a parallel program to implement Bucket Sort using OpenMP and Pthread Programming Constructs.
5. Design a parallel program to implement Prim's Algorithm using OpenMP and Pthread Programming Constructs.
6. Design a parallel program to implement DFS Algorithm using OpenMP and Pthread Programming Constructs.
7. Design a parallel program to implement BFS Algorithm using OpenMP and Pthread Programming Constructs.
8. Design a parallel program to implement Dijkstra's Algorithm using MPI library.

**VAAGDEVI COLLEGE OF ENGINEERING
(AUTONOMOUS)
COMPUTER SCIENCE AND ENGINEERING**

**DIGITAL FORENSICS
(PROFESSIONAL ELECTIVE - V)**

M.Tech CSE II Year I Sem.

| | | | |
|----------|----------|----------|----------|
| L | T | P | C |
| 3 | 0 | 0 | 3 |

Pre-Requisites: Cybercrime and Information Warfare, Computer Networks

Course Outcomes:

1. Understand relevant legislation and codes of ethics.
2. Computer forensics and digital detective and various processes, policies and procedures.
3. E-discovery, guidelines and standards, E-evidence, tools and environment.
4. Email and web forensics and network forensics.
5. Apply mobile forensic techniques and tools, and understand legal frameworks like the IT Act for handling electronic evidence.

UNIT - I

Digital Forensics Science: Forensics science, computer forensics, and digital forensics.

Computer Crime: Criminalistics as it relates to the investigative process, analysis of cyber criminalistics area, holistic approach to cyber-forensics

UNIT - II

Cyber Crime Scene Analysis:

Discuss the various court orders etc., methods to search and seizure electronic evidence, retrieved and un-retrieved communications, Discuss the importance of understanding what court documents would be required for a criminal investigation.

UNIT - III

Evidence Management & Presentation:

Create and manage shared folders using operating system, importance of the forensic mindset, define the workload of law enforcement, Explain what the normal case would look like, Define who should be notified of a crime, parts of gathering evidence, Define and apply probable cause.

UNIT - IV

Computer Forensics: Prepare a case, Begin an investigation, Understand computer forensics workstations and software, Conduct an investigation, Complete a case, Critique a case, Network Forensics: open-source security tools for network forensic analysis, requirements for preservation of network data.

UNIT - V

Mobile Forensics: mobile forensics techniques, mobile forensics tools.

Legal Aspects of Digital Forensics: IT Act 2000, amendment of IT Act 2008.

Recent trends in mobile forensic technique and methods to search and seizure electronic evidence

TEXT BOOKS:

1. John Sammons, The Basics of Digital Forensics, Elsevier
2. John Vacca, Computer Forensics: Computer Crime Scene Investigation, Laxmi Publications

REFERENCES:

1. William Oettinger, Learn Computer Forensics: A beginner's guide to searching, analyzing, and securing digital evidence, Packt Publishing; 1st edition (30 April 2020), ISBN : 1838648178.
2. Thomas J. Holt , Adam M. Bossler, Kathryn C. Seigfried-Spellar , Cybercrime and Digital Forensics: An Introduction, Routledge.

**VAAGDEVI COLLEGE OF ENGINEERING
(AUTONOMOUS)
COMPUTER SCIENCE AND ENGINEERING**

**ADVANCED OPERATING SYSTEMS
(PROFESSIONAL ELECTIVE - V)**

M.Tech CSE II Year I Sem.

| L | T | P | C |
|----------|----------|----------|----------|
| 3 | 0 | 0 | 3 |

Prerequisites : Operating Systems

Course Outcomes

1. Understand distributed system architectures, operating systems, and logical clock concepts.
2. Analyze mutual exclusion algorithms in distributed systems.
3. Apply deadlock detection techniques in distributed environments.
4. Explain multiprocessor architectures and distributed file system design.
5. Evaluate distributed scheduling and shared memory algorithms.

UNIT - I

Architectures of Distributed Systems: System Architecture Types, Distributed Operating Systems, Issues in Distributed Operating Systems, Communication Primitives. Theoretical Foundations: Inherent Limitations of a Distributed System, Lamport's Logical Clocks, Vector Clocks, Causal Ordering of Messages, Termination Detection.

UNIT - II

Distributed Mutual Exclusion: The Classification of Mutual Exclusion Algorithms, Non-Token – Based Algorithms: Lamport's Algorithm, The Ricart-Agrawala Algorithm, Maekawa's Algorithm, Token-Based Algorithms: Suzuki-Kasami's Broadcast Algorithm, Singhal's Heuristic Algorithm, Raymond's Heuristic Algorithm.

UNIT - III

Distributed Deadlock Detection: Preliminaries, Deadlock Handling Strategies in Distributed Systems, Issues in Deadlock Detection and Resolution, Control Organizations for Distributed Deadlock Detection, Centralized- Deadlock – Detection Algorithms, Distributed Deadlock Detection Algorithms, Hierarchical Deadlock Detection Algorithms

UNIT - IV

Multiprocessor System Architectures: Introduction, Motivation for multiprocessor Systems, Basic Multiprocessor System Architectures Multi Processor Operating Systems: Introduction, Structures of Multiprocessor Operating Systems, Operating Design Issues, Threads, Process Synchronization, Processor Scheduling.

Distributed File Systems: Architecture, Mechanisms for Building Distributed File Systems, Design Issues

UNIT - V

Distributed Scheduling: Issues in Load Distributing, Components of a Load Distributed Algorithm, Stability, Load Distributing Algorithms, Requirements for Load Distributing, Task Migration, Issues in task Migration Distributed Shared Memory: Architecture and Motivation, Algorithms for Implementing DSM, Memory Coherence, Coherence Protocols, Design Issues

TEXT BOOK:

1. Advanced Concepts in Operating Systems, Mukesh Singhal, Niranjana G. Shivaratri, Tata McGraw-Hill Edition 2001.

REFERENCE BOOK:

1. Distributed Systems: Andrew S. Tanenbaum, Maarten Van Steen, Pearson Prentice Hall, Edition – 2, 2007.

**VAAGDEVI COLLEGE OF ENGINEERING
(AUTONOMOUS)
COMPUTER SCIENCE AND ENGINEERING**

**QUANTUM COMPUTING
(PROFESSIONAL ELECTIVE - V)**

M.Tech CSE II Year I Sem.

| | | | |
|----------|----------|----------|----------|
| L | T | P | C |
| 3 | 0 | 0 | 3 |

Prerequisites :

Basic knowledge of linear algebra and probability, Cryptography and Network Security

Course Outcomes

1. Understand the fundamentals of linear algebra, complex numbers, vector spaces, and basic algebra essential for quantum computing.
2. Comprehend the basic principles of quantum physics, including quantum states, entanglement, Hilbert spaces, and quantum decoherence.
3. Analyze and explain quantum architecture, including qubits, quantum gates, quantum circuits, and hardware considerations.
4. Understand and implement key quantum algorithms, such as Deutsch, Deutsch-Jozsa, Bernstein-Vazirani, Simon's, Shor's, and Grover's algorithms.
5. Evaluate the impact of quantum computing on cryptography, including asymmetric algorithms like RSA, Diffie-Hellman, elliptic curve cryptography, and their applications.

Unit I

Introduction to Essential Linear Algebra

Some Basic Algebra, Matrix Math, Vectors and Vector Spaces, Set Theory

Complex Numbers

Definition of Complex Numbers, Algebra of Complex Numbers, Complex Numbers Graphically, Vector

Representations of Complex Numbers, Pauli Matrices, Transcendental Numbers

Unit II

Basic Physics for Quantum Computing

The Journey to Quantum, Quantum Physics Essentials, Basic Atomic Structure, Hilbert Spaces, Uncertainty, Quantum States, Entanglement

Basic Quantum Theory

Further with Quantum Mechanics, Quantum Decoherence, Quantum Electrodynamics, Quantum Chromodynamics, Feynman Diagram Quantum Entanglement and QKD, Quantum Entanglement,

Interpretation, QKE

Unit III

Quantum Architecture

Further with Qubits, Quantum Gates, More with Gates, Quantum Circuits, The D-Wave

Quantum Architecture

Quantum Hardware

Qubits, How Many Qubits Are Needed? Addressing Decoherence, Topological Quantum Computing,

Quantum Essentials

Unit IV

Quantum Algorithms

What Is an Algorithm? Deutsch's Algorithm, Deutsch-Jozsa Algorithm, Bernstein-Vazirani Algorithm,

Simon's Algorithm, Shor's Algorithm, Grover's Algorithm

Unit V

Current Asymmetric Algorithms: RSA, Diffie-Hellman, Elliptic Curve

The Impact of Quantum Computing on Cryptography: Asymmetric Cryptography, Specific Algorithms, Specific Applications.

TEXT BOOKS:

1. Nielsen M. A., Quantum Computation and Quantum Information, Cambridge University Press
2. Dr. Chuck Easttom, Quantum Computing Fundamentals, Pearson

REFERENCES:

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. Basic Concepts, Vol
3. Basic Tools and Special Topics, World Scientific. Pittenger A. O., An Introduction to Quantum Computing Algorithms.

**VAAGDEVI COLLEGE OF ENGINEERING
(AUTONOMOUS)
COMPUTER SCIENCE AND ENGINEERING**

**PROMPT ENGINEERING
(PROFESSIONAL ELECTIVE - V)**

M.Tech CSE II Year I Sem.

| | | | |
|----------|----------|----------|----------|
| L | T | P | C |
| 3 | 0 | 0 | 3 |

Prerequisites : Programming, Machine Learning and Natural Language Processing.

Course Outcomes:

1. Explain and apply the core principles of prompt engineering for guiding generative AI outputs effectively.
2. Describe the underlying architecture and functionality of state-of-the-art large language models (LLMs).
3. Generate and manipulate structured outputs (JSON, YAML, CSV) using ChatGPT with advanced prompting techniques.
4. Implement text chunking, tokenization, and format control using tools like SpaCy, Tiktoken, and Python.
5. Utilize vector databases such as FAISS and Pinecone in Retrieval-Augmented Generation (RAG) pipelines for efficient information retrieval.

UNIT – I

Fundamentals and Principles of Prompting

Overview of the Five Principles of Prompting: Give Direction, Specify Format, Provide Examples, Evaluate Quality, Divide Labor.

UNIT – II

Introduction to Large Language Models for Text Generation

What Are Text Generation Models, Vector Representations: The Numerical Essence of Language, Transformer Architecture: Orchestrating Contextual Relationships, Probabilistic Text Generation: The Decision Mechanism, Historical Underpinnings: The Rise of Transformer Architectures, OpenAI’s Generative Pretrained Transformers, GPT-3.5-turbo and ChatGPT, GPT-4, Google’s Gemini, Meta’s Llama and Open Source.

UNIT – III

Standard Practices for Text Generation with ChatGPT- Part-A

Generating Lists, Hierarchical List Generation, When to Avoid Using Regular Expressions, Generating JSON, YAML Filtering YAML Payloads, Handling Invalid Payloads in YAML, Diverse Format Generation with ChatGPT, Mock CSV Data, Universal Translation Through LLMs, Ask for Context, Text Style Unbundling, Identifying the Desired Textual Features,

Generating New Content with the Extracted Features, Extracting Specific Textual Features with LLMs.

UNIT – IV

Standard Practices for Text Generation with ChatGPT- Part-B

Chunking Text, Benefits of Chunking Text, Scenarios for Chunking Text, Poor Chunking Example, Chunking Strategies, Sentence Detection Using SpaCy, building a Simple Chunking Algorithm in Python, Sliding Window Chunking, Text Chunking Packages, Text Chunking with Tiktoken, Encodings, Understanding the Tokenization of Strings.

UNIT – V

Vector Databases with FAISS and Pinecone

Retrieval Augmented Generation (RAG), Introducing Embeddings, Document Loading Memory Retrieval with FAISS, RAG with Lang Chain, Hosted Vector Databases with Pinecone, SelfQuerying, Alternative Retrieval Mechanisms.

TEXTBOOK:

1. Phoenix J, Taylor M. Prompt engineering for generative AI. " O'Reilly Media, Inc."; 2024 May 16.

REFERENCES:

1. Tunstall L, Von Werra L, Wolf T. Natural language processing with transformers. " O'Reilly Media, Inc."; 2022 Jan 26.
2. Foster D. Generative deep learning. " O'Reilly Media, Inc."; 2022 Jun 28.

**VAAGDEVI COLLEGE OF ENGINEERING
(AUTONOMOUS)
COMPUTER SCIENCE AND ENGINEERING

INTELLECTUAL PROPERTY RIGHTS
(OPEN ELECTIVE)**

M.Tech CSE II Year I Sem.

| | | | |
|----------|----------|----------|----------|
| L | T | P | C |
| 3 | 0 | 0 | 3 |

Prerequisites :

Basic understanding of law, business processes, and innovation concepts is essential for studying Intellectual Property Rights.

Course Outcomes:

1. Distinguish and Explain various forms of IPRs.
2. Identify criteria to fit one's own intellectual work in particular form of IPRs.
3. Apply statutory provisions to protect particular form of IPRs

UNIT – I

Introduction to Intellectual property: Introduction, types of intellectual property, international organizations, agencies and treaties, importance of intellectual property rights.

UNIT – II

Trade Marks: Purpose and function of trademarks, acquisition of trade mark rights, protectable matter, selecting, and evaluating trade mark, trade mark registration processes.

UNIT – III

Law of copyrights: Fundamental of copyright law, originality of material, rights of reproduction, rights to perform the work publicly, copyright ownership issues, copyright registration, notice of copyright, international copyright law.

Law of patents: Foundation of patent law, patent searching process, ownership rights and transfer

UNIT – IV

Trade Secrets: Trade secret law, determination of trade secret status, liability for misappropriations of trade secrets, protection for submission, trade secret litigation. Unfair competition: Misappropriation right of publicity, false advertising.

UNIT – V

New development of intellectual property: new developments in trade mark law; copyright law, patent law, intellectual property audits.

International overview on intellectual property, international – trade mark law, copyright law, international patent law, and international development in trade secrets law.

TEXT & REFERENCE BOOKS:

1. Intellectual property right, Deborah. E. Bouchoux, Cengage learning.
2. Intellectual property right – Unleashing the knowledge economy, prabuddhaganguli, Tata McGraw Hill Publishing company ltd

**VAAGDEVI COLLEGE OF ENGINEERING
(AUTONOMOUS)
COMPUTER SCIENCE AND ENGINEERING
GENERATIVE AI (OPEN ELECTIVE)**

M.Tech CSE II Year I Sem.

| | | | |
|----------|----------|----------|----------|
| L | T | P | C |
| 3 | 0 | 0 | 3 |

Prerequisite: AI, Machine Learning.

Course Outcomes

1. Demonstrate knowledge of AI foundations, generative models, and advanced neural architectures.
2. Apply generative AI techniques to create solutions for text, image, video, and multimodal tasks.
3. Design, fine-tune, and optimize Large Language Models for specific applications.
4. Evaluate ethical, social, and legal implications of Generative AI deployments and propose mitigation strategies.
5. Gain practical experience with state-of-the-art framework and tools.

UNIT-I

Foundations of AI and Generative Models

Introduction and historical evolution to Artificial Intelligence (AI), Machine Learning (ML), Natural

Language Processing (NLP) and Deep Learning (DL), Structure of Artificial Neural Networks (ANNs), Mathematical and computational foundations of generative modeling, Overview of generative models and their applications across various domains; Importance of Generative AI in modern applications, Transfer learning and in advancing Generative AI

UNIT-II

Advanced Neural Architectures for Generative AI

Variational Autoencoders (VAEs): principles and applications, Generative Adversarial Networks (GANs): architecture and working principles; Transformer architecture and attention mechanisms (in detail); Long Short-Term Memory Networks (LSTMs) and the limitations of traditional RNNs/LSTMs, Advanced Transformer architectures and techniques, Pre-training and transfer learning strategies for generative models

UNIT-III

Large Language Models and Prompt Engineering

Overview of Large Language Models (LLMs), GPT architecture, variants, and working principles, Prétraining and fine-tuning GPT models for applications (e.g., chatbots, text generation), Case study: GPT-based customer support chatbot, BERT architecture, pre-training objectives, and fine-tuning, Prompt Engineering: Designing effective prompts, controlling

model behavior, and improving output quality, Fine-tuning language models for creative writing and chatbot development

UNIT-IV

Multi-Agent Systems and Generative AI Applications

Introduction to Multi-Agent Systems (MAS), Types of agents: reactive, deliberative, hybrid, and learning agents, Multi-agent collaboration and orchestration for generative tasks, Use cases: autonomous research assistants, cooperative creative generation, distributed problem-solving, Frameworks and tools: AutoGen, CrewAI, Hugging GPT for LLM-powered multi-agent systems, Generative AI applications: Art, Creativity, Image/Video generation, Music composition, Healthcare, Finance, Real-world case studies and deployment challenges

UNIT-V

Frameworks, Multimodal Applications, and Ethics

LangChain framework: components and LLM application development, Retrieval-Augmented Generation (RAG), Embeddings, Indexing networks, and Vector databases, Generative AI across modalities: Text, Code, Image, and Video generation, Image and Video generation using GANs and VAEs, Multimodal Generative AI: integration and training strategies, Ethical considerations: bias, fairness, trust, and responsible AI deployment, Social and legal implications of Generative AI, Risk mitigation strategies and real-world ethical case studies

TEXT BOOKS:

1. Altaf Rehmani, Generative AI for Everyone: Understanding the Essentials and Applications of This Breakthrough Technology.
2. Charu C. Aggarwal, Neural Networks and Deep Learning: A Textbook. Joseph Babcock, Raghav Bali, Generative AI with Python and TensorFlow 2, 2024.

REFERENCE BOOKS:

1. Josh Kalin, Generative Adversarial Networks Cookbook.
2. Jesse Sprinter, Generative AI in Software Development: Beyond the Limitations of Traditional Coding, 2024.

ONLINE REFERENCES:

1. Fabian Gloeckle et al., Better & Faster Large Language Models via Multi-token Prediction, arXiv:2404.19737v1, 2024. Vaswani et al., Attention Is All You Need, NeurIPS 2017.

**VAAGDEVI COLLEGE OF ENGINEERING
(AUTONOMOUS)
COMPUTER SCIENCE AND ENGINEERING**

**INTRUSION DETECTION SYSTEMS
(OPEN ELECTIVE)**

M.Tech CSE II Year I Sem.

| L | T | P | C |
|---|---|---|---|
| 3 | 0 | 0 | 3 |

Prerequisites: Computer Networks, Computer Programming

Course Outcomes:

1. Understand the fundamental concepts of intrusion detection and prevention.
2. Identify various types of attacks at the network layer and human layer, including code injection techniques.
3. Analyze different anomaly detection algorithms.
4. Evaluate anomaly detection techniques for networks and hosts, including behavioral models, software vulnerabilities, and payload-based methods.
5. Assess advanced threat detection strategies, malware behaviors, insider threats, and future approaches in collaborative cybersecurity.

UNIT - I

The state of threats against computers, and networked systems-Overview of computer security solutions and why they fail-Vulnerability assessment, firewalls, VPN's -Overview of Intrusion Detection and Intrusion Prevention, Network and Host-based IDS

UNIT - II

Classes of attacks - Network layer: scans, denial of service, penetration Application layer: software exploits, code injection-Human layer: identity theft, root access-Classes of attackers-Kids/hackers/sop Hesitated groups-Automated: Drones, Worms, Viruses

UNIT - III

A General IDS model and taxonomy, Signature-based Solutions, Snort, Snort rules, Evaluation of IDS, Cost sensitive IDS

UNIT - IV

Anomaly Detection Systems and Algorithms-Network Behavior Based Anomaly Detectors (rate based)-Host-based Anomaly Detectors-Software Vulnerabilities-State transition, Immunology, Payload Anomaly Detection

UNIT - V

Attack trees and Correlation of alerts- Autopsy of Worms and Botnets-Malware detection Obfuscation, polymorphism- Document vectors. Email/IM security issues-Viruses/Spam-From signatures to thumbprints to zero day detection-Insider Threat issues-Taxonomy-Masquerade and Impersonation Traitors, Decoys and Deception-Future: Collaborative Security

TEXT BOOKS:

1. Peter Szor, The Art of Computer Virus Research and Defense, Symantec Press ISBN 0-321-30545-3.
2. Markus Jakobsson and Zulfikar Ramzan, Crimeware, Understanding New Attacks and Defenses.

REFERENCE BOOKS:

1. Saiful Hasan, Intrusion Detection System, Kindle Edition.
2. Ankit Fadia, Intrusion Alert: An Ethical Hacking Guide to Intrusion Detection.

Online Websites/Materials:

1. <https://www.intechopen.com/books/intrusion-detection-systems/> Online Courses:
2. <https://www.sans.org/course/intrusion-detection-in-depth>
3. <https://www.cybrary.it/skill-certification-course/ids-ips-certification-training-course>

**VAAGDEVI COLLEGE OF ENGINEERING
(AUTONOMOUS)
COMPUTER SCIENCE AND ENGINEERING**

**DIGITAL FORENSICS
(OPEN ELECTIVE)**

M.Tech CSE II Year I Sem.

| L | T | P | C |
|----------|----------|----------|----------|
| 3 | 0 | 0 | 3 |

Pre-Requisites: Cybercrime and Information Warfare, Computer Networks

Course Outcomes: On completion of the course the student should be able to

1. Understand relevant legislation and codes of ethics.
2. Computer forensics and digital detective and various processes, policies and procedures.
3. E-discovery, guidelines and standards, E-evidence, tools and environment.
4. Email and web forensics and network forensics.
5. Apply mobile forensic techniques and tools, and understand legal frameworks like the IT Act for handling electronic evidence.

UNIT - I

Digital Forensics Science: Forensics science, computer forensics, and digital forensics.

Computer Crime: Criminalistics as it relates to the investigative process, analysis of cyber criminalistics area, holistic approach to cyber-forensics

UNIT - II

Cyber Crime Scene Analysis:

Discuss the various court orders etc., methods to search and seizure electronic evidence, retrieved and un-retrieved communications, Discuss the importance of understanding what court documents would be required for a criminal investigation.

UNIT - III

Evidence Management & Presentation:

Create and manage shared folders using operating system, importance of the forensic mindset, define the workload of law enforcement, Explain what the normal case would look like, Define who should be notified of a crime, parts of gathering evidence, Define and apply probable cause.

UNIT - IV

Computer Forensics: Prepare a case, Begin an investigation, Understand computer forensics workstations and software, Conduct an investigation, Complete a case, Critique a case, Network Forensics: open-source security tools for network forensic analysis, requirements for preservation of network data.

UNIT - V

Mobile Forensics: mobile forensics techniques, mobile forensics tools.

Legal Aspects of Digital Forensics: IT Act 2000, amendment of IT Act 2008.

Recent trends in mobile forensic technique and methods to search and seizure electronic evidence

TEXT BOOKS:

1. John Sammons, The Basics of Digital Forensics, Elsevier
2. John Vacca, Computer Forensics: Computer Crime Scene Investigation, Laxmi Publications

REFERENCES:

1. William Oettinger, Learn Computer Forensics: A beginner's guide to searching, analyzing, and securing digital evidence, Packt Publishing; 1st edition (30 April 2020), ISBN : 1838648178.
2. Thomas J. Holt, Adam M. Bossler, Kathryn C. Seigfried-Spellar , Cybercrime and Digital Forensics: An Introduction, Routledge.