

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING (AI&ML)

B.Tech Course Structure (R23 Regulations)

II B.Tech (I Sem)

Branch/Specialization: CSE (AI&ML)

S.No.	Course Code	Name of the Course	Contact Hours per Week			Credits
			L	T	P	
1	HS204	Mathematical and Statistical Foundations	3	1	0	4
2	EC216	Digital Electronics	3	0	0	3
3	CS206	Computer Organization and Architecture	3	0	0	3
4	CS201	Data Structures using C++	3	0	0	3
5	CS301	Software Engineering	3	0	0	3
6	CS208	Operating Systems	3	0	0	3
7	CS202	Data Structures using C++ Lab	0	0	2	1
8	CS209	Operating Systems Lab	0	0	2	1
9	MC202	Gender Sensitization Lab	0	0	2	0
Total			18	1	6	21

L: Lecture Hour

T: Tutorial Hour

P: Lab/Practice Hour



MATHEMATICAL AND STATISTICAL FOUNDATIONS

B.Tech. II Year I Sem.

Course Objectives: To Learn

L	T	P	C
3	0	0	3

- The Number Theory basic concepts useful for cryptography etc.
- The theory of Probability and Probability distributions of single and multiple random variables.
- The sampling theory and testing of hypotheses and making statistical inferences.
- Stochastic process and Markov chains.

Course outcomes: At the End of the course, the student will be able to

- Apply the number theory concepts to cryptography domain.
- Apply the concepts probability distributions to some case studies.
- Apply the concept of normal distribution.
- Apply the concept of testing of hypothesis to case studies.
- Apply the concept of Markov chain steady state condition.

UNIT-I: Greatest Common Divisors and Prime Factorization:

Greatest common divisors, The Euclidean algorithm, The fundamental theorem of arithmetic, Factorization of integers and the Fermat numbers.

Congruences: Introduction to Congruences, Linear Congruences, The Chinese remainder theorem, Systems of linear Congruences.

UNIT-II: Simple Linear Regression and Correlation:

Introduction to Linear Regression, The simple Linear Regression Model, Least Squares and the Fitted Model, Properties of the Least Squares Estimators, Inferences Concerning the Regression Coefficients, Prediction, Simple Linear Regression case Study.

Random Variables and Probability Distributions: Concept of a Random Variable, Discrete Probability Distributions, Continuous Probability Distributions, Statistical Independence.

Discrete Probability Distributions: Binomial Distribution, Poisson distribution.

UNIT-III: Continuous Probability Distribution:

Normal Distribution, Areas under the Normal Curve, Applications of the Normal Distribution. Normal Approximation to the binomial.

Fundamental Sampling Distributions: Random Sampling, Sampling Distributions, Sampling Distribution of Means and the central limit Theorem, Sampling Distribution of S^2 , t-Distribution, F-Distribution.

UNIT-IV: Estimation & Tests of Hypotheses:

Introduction, statistical Inference, classical methods of estimation, estimating the Mean, Standard Error of a Point Estimate, Prediction Intervals, Tolerance limits, Estimating the Variance, Estimating a Proportion for single mean, Difference between Two Means, between Two Proportions for Two Samples and Maximum Likelihood Estimation.

UNIT-V: Stochastic Processes and Markov Chains

Introduction to stochastic processes-Markov process, Transition Probability, Transition Probability Matrix, First order and Higher order Markov process, n-step transition probabilities, Markov chain, Steady state condition, Markov analysis.

TEXT BOOK:

1. Ronald E. Walpole, Raymond H. Myers, Sharon L. Myers, Yinying Ye, Probability and statistics for engineers and scientists, 9th Edition, Pearson Publications.
2. SC Gupta and VK Kapoor, Fundamentals of Mathematical statistics, Khanna publications.
3. S.D. Sharma, Operations Research, Kadamath and Ramnath Publishers, Meerut, Delhi.

REFERENCE BOOKS:

1. T.T Soong, Fundamentals of Probability and Statistics for Engineers, John Wiley & Sons, Ltd, 2004.
 2. Sheldon M Ross, Probability and statistics for Engineers and scientists, academic press.
- Miller and Freund's, Probability and Statistics for Engineers, 8th Edition, Pearson Education



R23 B.Tech CSE (AI & ML) Syllabus

AVNIET

DIGITAL ELECTRONICS

L	T	P	C
3	0	0	3

B.Tech. II Year I Sem.

Course Objectives:

This course aims at through understanding of binary number system, logic gates, combination logic and synchronous and asynchronous logic.

UNIT - I:

BOOLEAN ALGEBRA AND LOGIC GATES: Digital Systems, Binary Numbers, Number base conversions, Octal and Hexadecimal Numbers, complements, Signed binary numbers, Binary codes, Binary Storage and Registers, Binary logic.

Basic Definitions, Axiomatic definition of Boolean Algebra, Basic theorems and properties of Boolean algebra, Boolean functions, canonical and standard forms, other logic operations, Digital logic gates.

UNIT - II:

GATE – LEVEL MINIMIZATION: The map method, Four-variable map, Five-Variable map, product of sums simplification Don't-care conditions, NAND and NOR implementation other Two-level implementations, Exclusive – Or function.

UNIT - III:

COMBINATIONAL LOGIC: Combinational Circuits, Analysis procedure Design procedure, Binary Adder-Subtractor Decimal Adder, Binary multiplier, magnitude comparator, Decoders, Encoders, Multiplexers, HDL for combinational circuits.

UNIT - IV:

SEQUENTIAL LOGIC: Sequential circuits, latches, Flip-Flops Analysis of clocked sequential circuits, state Reduction and Assignment, Design Procedure. Registers shift Registers, Ripple counters, synchronous counters, other counters.

UNIT - V

MEMORIES AND ASYNCHRONOUS SEQUENTIAL LOGIC: Introduction, Random-Access Memory, Memory Decoding, Error Detection and correction Read-only memory, Programmable logic Array programmable Array logic, Sequential Programmable Devices.

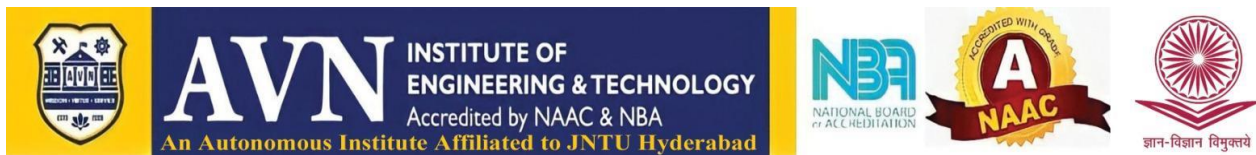
Introduction, Analysis Procedure, Circuits with Latches, Design Procedure, Reduction of state and FlowTables, Race-Free state Assignment Hazards, Design Example.

TEXT BOOKS:

1. Digital Design – Third Edition, M. Morris Mano, Pearson Education/PHI.
2. Digital Principles and Applications Albert Paul Malvino Donald P. Leach TATA McGraw Hill Edition.
3. Fundamentals of Logic Design, Roth, 5th Edition, Thomson.

REFERENCE BOOKS:

1. Switching and Finite Automata Theory by Zvi. Kohavi, Tata McGraw Hill.
2. Switching and Logic Design, C.V.S. Rao, Pearson Education
3. Digital Principles and Design – Donald D.Givone, Tata McGraw Hill, Edition.
4. Fundamentals of Digital Logic and Microcomputer Design, 5TH Edition, M. Rafiquzzaman John Wiley.



R23 B.Tech CSE (AI & ML) Syllabus

AVNIET

COMPUTER ORGANIZATION AND ARCHITECTURE

B.Tech. II Year I Sem.

L	T	P	C
3	0	0	3

Prerequisite(s): A Course on “Digital Electronics”.

Course Objectives:

- The purpose of the course is to introduce principles of computer organization and the basic architectural concepts.
- It begins with basic organization, design, and programming of a simple digital computer and introduces simple register transfer language to specify various computer operations.
- Topics include computer arithmetic, instruction set design, micro programmed control unit, pipelining and vector processing,
- Memory organization and I/O systems, and multiprocessors

Course Outcomes:

- Understand the basics of instructions sets and their impact on processor design.
- Demonstrate an understanding of the design of the functional units of a digital computer system.
- Evaluate cost performance and design trade-offs in designing and constructing a computer processor including memory.
- Design a pipeline for consistent execution of instructions with minimum hazards.
- Recognize and manipulate representations of numbers stored in digital computers

UNIT - I:

Digital Computers: Introduction, Block diagram of Digital Computer, Definition of Computer Organization, Computer Design and Computer Architecture.

Register Transfer Language and Micro operations: Register Transfer language, Register Transfer, Bus and memory transfers, Arithmetic Micro operations, logic micro operations, shift micro operations, Arithmetic logic shift unit.

Basic Computer Organization and Design: Instruction codes, Computer Registers ,Computer instructions, Timing and Control, Instruction cycle, Memory Reference Instructions, Input – Output and Interrupt.

UNIT - II:

Microprogrammed Control: Control memory, Address sequencing, micro program example, design of control unit, Micro Program Sequencer

Central Processing Unit: General Register Organization, Addressing modes, Data Transfer and Manipulation, Program Control.

UNIT - III:

Data Representation: Data types, Complements, Fixed Point Representation, Floating Point Representation.

Computer Arithmetic: Addition and subtraction, multiplication Algorithms, Division Algorithms, Floating-point Arithmetic operations

UNIT - IV:

Input-Output Organization: Input-Output Interface, Modes of Transfer, Priority Interrupt, Direct memory Access.

Memory Organization: Memory Hierarchy, Main Memory, Auxiliary memory, Associate Memory, Cache Memory.

UNIT - V:

Reduced Instruction Set Computer: CISC Characteristics, RISC Characteristics.

Pipeline and Vector Processing: Parallel Processing, Pipelining, Arithmetic Pipeline, Instruction Pipeline, RISC Pipeline, Vector Processing, Array Processor.

Multi Processors: Characteristics of Multiprocessors, Interconnection Structures, Inter-processor arbitration, Cache Coherence

TEXT BOOKS:

1. Computer System Architecture – M. Moris Mano, Third Edition, Pearson/PHI.

REFERENCE BOOKS:

1. Computer Organization – Carl Hamacher, Zvonks Vranesic, Saeed Zaky, Vth Edition, McGraw Hill.
2. Computer Organization and Architecture – William Stallings Sixth Edition, Pearson/PHI.
3. Structured Computer Organization – Andrew S. Tanenbaum, 4th Edition, PHI/Pearson.



R23 B.Tech CSE (AI & ML) Syllabus

AVNIET

DATA STRUCTURES USING C++

B.Tech. II Year I Sem.

L	T	P	C
3	0	0	3

Prerequisite(s): A Course on “Programming for problem solving”.

Course Objectives:

- Exploring basic data structures such as stacks and queues.
- Introduces a variety of data structures such as hash tables, search trees, heaps, graphs.
- Introduces sorting and graph techniques.

Course outcomes:

The students should be able to:

- Learn and implement the basic programmes using C++.
- Learn the basic types of data structures stacks and Queues.
- Design programs using a variety of data structures, including Dictionaries, hash tables.
- Understanding the concepts of tree structure, search trees, and AVL-trees
- Implement and know the application of algorithms for sorting and graph techniques.

UNIT - I:

INTRODUCTION:

An Overview of C++ : The Origins of C++, C++ fundamentals, Introducing C++ Classes, Objects, Functions, constructors, destructors.

Abstract Data Types, Arrays, Linear list – singly linked list implementation, insertion, deletion and searching operations on linear list, doubly linked list operations.

UNIT– II

Stacks-Operations, array and linked representations of stacks, stack applications, Queues-operations, array and linked representations.

Dictionaries: linear list representation, skip list representation, operations - insertion, deletion and searching.

UNIT– III

Hash Table Representation: hash functions, Division Method, Mid Square Method, Folding, Multiplicative Method, collision resolution-separate chaining, open addressing, linear probing, and quadratic probing, double hashing, rehashing, extendible hashing.

UNIT– IV

Basic tree concepts, Binary trees: properties, types, representation of binary trees using arrays and linked lists, traversals of binary tree.

Search Trees: Binary Search Trees, Definition, Implementation, Operations- Searching, Insertion and Deletion, B tree operations, AVL Trees, Definition, Height of an AVL Tree, Operations – Insertion,

Deletion and Searching, Red –Black Trees.

UNIT– V

Sorting: Heap Sort, Quick Sort and Merge Sort

Graphs: Graph Implementation Methods. Graph Traversal Methods.

TEXT BOOKS:

1. "Data Structures Using C++" by D.S. Malik - A beginner-friendly textbook introducing data structures and their implementation in C++.
2. "Object Oriented Programming with C++", E Balaguruswamy.
3. Fundamentals of Data Structures , 2nd Edition, E. Horowitz, S. Sahni and Susan Anderson

REFERENCE BOOKS:

1. Data Structures – A. S. Tanenbaum, Y. Langsam, and M.J. Augenstein, PHI/Pearson.
2. Data Structures: A Pseudocode Approach with C, 2nd Edition, R. F. Gilberg and B.A. Forouzan, Cengage Learning.



R23 B.Tech CSE (AI & ML) Syllabus

AVNIET

SOFTWARE ENGINEERING

B.Tech. II Year, I Sem.

L	T	P	C
3	0	0	3

Course Objectives

- The aim of the course is to provide an understanding of the working knowledge of the techniques for estimation, design, testing and quality management of large software development projects.
- Topics include process models, software requirements, software design, software testing, software process/product metrics, risk management, quality management and UML diagrams

Course Outcomes

- Able to compare and select a process model for a business system.
- To identify and specify the requirements for the development of an application
- apply appropriate software architectures and patterns to carry out high level design of a system and be able to critically compare alternative choices.
- Will to have experience and/or awareness of testing problems and will be able to develop a simple testing report
- To develop and maintain efficient, reliable and cost effective software solution through risk management system

UNIT - I

Introduction to Software Engineering: The evolving role of software, changing nature of software, software myths.

A Generic view of process: Software engineering- a layered technology, the capability maturity model integration (CMMI). **Process models:** The waterfall model, Spiral model and Agile methodology

UNIT - II

Software Requirements: Functional and non-functional requirements, user requirements, system requirements, interface specification, the software requirements document.

Requirements engineering process: requirements elicitation and analysis, requirements validation, requirements management.

UNIT - III

Design Engineering: Design process and design quality, design concepts, the design model. Creating an architectural design: software architecture, data design, architectural design, conceptual model of UML, basic structural modeling, class diagrams, sequence diagrams, collaboration diagrams, use case diagrams, component diagrams.

UNIT - IV

Testing Strategies: A strategic approach to software testing, black-box and white-box testing, validation testing, system testing, the art of debugging.

Metrics for Process and Products: Software measurement, metrics for software quality.

UNIT - V

Risk management: Reactive Vs proactive risk strategies, software risks, risk identification, risk projection, risk refinement, RMMM.

Quality Management: Quality concepts, software quality assurance, software reviews, formal technical reviews, statistical software quality assurance, the ISO 9000 quality standards.

TEXT BOOKS:

1. Software Engineering, A practitioner's Approach- Roger S. Pressman, 6th edition, McGraw Hill International Edition.
2. Software Engineering- Sommerville, 7th edition, Pearson Education

REFERENCE BOOKS:

1. The unified modeling language user guide Grady Booch, James Rumbaugh, Ivar Jacobson, Pearson Education.
2. Software Engineering, an Engineering approach- James F. Peters, Witold Pedrycz, John Wiley.
3. Software Engineering principles and practice- Waman S Jawadekar, The McGraw-Hill Companies.
4. Fundamentals of object-oriented design using UML Meiler page-Jones: Pearson Education.



R23 B.Tech CSE (AI & ML) Syllabus

AVNIET

OPERATING SYSTEMS

B.Tech. II Year I Sem.

L	T	P	C
3	0	0	3

Prerequisites: “Components of Computer systems”, “Digital Logic Design

Course Objectives:

- To understand the basic concepts and functions of operating systems.
- To understand Processes, Threads and Deadlocks.
- To analyze Scheduling algorithms.
- To analyze memory management schemes.
- To understand I/O management and File systems.
- To be familiar with the basics of Linux system.

Course Outcomes:

- Analyze various scheduling algorithms.
- Understand deadlock, prevention and avoidance algorithms.
- Compare and contrast various memory management schemes.
- Understand the functionality of file systems.
- Perform administrative tasks on Linux Servers.
- Compare iOS and Android Operating Systems.

UNIT-I:

Operating System - Introduction, Structures - Simple Batch, Multiprogrammed, Time-shared, Personal Computer, Parallel, Distributed Systems, Real-Time Systems, Objectives and functions, System components, Evolution of Operating Systems, Operating System services, System Calls.

Process - Process concepts and scheduling, Operations on processes, Cooperating Processes, Threads

UNIT - II:

CPU Scheduling - Scheduling Criteria, Scheduling Algorithms, Multiple -Processor Scheduling. System call interface for process management-fork, exit, wait, waitpid, exec.

Deadlocks - System Model, Deadlocks Characterization, Methods for Handling Deadlocks, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, and Recovery from Deadlock

UNIT – III:

Process Management and Synchronization - The Critical Section Problem, Synchronization Hardware, Semaphores, and Classical Problems of Synchronization, Critical Regions, Monitors.

Inter-process Communication Mechanisms - IPC between processes on a single computer system, IPC between processes on different systems, using pipes, FIFOs, message queues, shared memory.

UNIT – IV:

Memory Management and Virtual Memory - Logical versus Physical Address Space, Swapping, Contiguous Allocation, Paging, Segmentation, Segmentation with Paging, Demand Paging, Page Replacement, Page Replacement Algorithms.

UNIT – V:

Mass-storage Structure - Overview of Mass-storage structure, Disk structure, Disk attachment, Disk scheduling, Swap-space management. File System Implementation: Access Methods, File system structure.

File system implementation: Directory implementation, Allocation methods, Free-space management.

TEXTBOOKS:

1. Operating System Principles, 7th Edition by AviSilberschatz, Peter Galvin, Greg Gagne, Wiley Asia Student Edition.

REFERENCEBOOKS:

2. Operating Systems: Internals and Design Principles, 5th Edition, William Stallings, Prentice Hall of India.
3. Operating System: A Design-oriented Approach, 1st Edition by Charles Crowley, Irwin Publishing
4. Operating Systems: A Modern Perspective, 2nd Edition by Gary J. Nutt, Addison- Wesley



R23 B.Tech CSE(AI & ML) Syllabus

AVNIET

DATA STRUCTURES USING C++ LAB

B.Tech. II Year I Sem.

L	T	P	C
0	0	2	1

Course Objectives:

- It covers various concepts of C++ programming language
- It introduces searching and sorting algorithms
- It provides an understanding of data structures such as stacks and queues.

Course Outcomes:

- Ability to develop C++ programs for computing and real-life applications using basic elements like Control statements, arrays, functions, pointers and data structures like stacks, Queues and linked lists.
- Ability to implement searching and sorting algorithms.

LIST OF EXPERIMENTS:

1. Write a program to find the Maximum and Minimum elements in an array.
2. Write a program that uses functions to perform the following operations on singly linked list.
i) Creation ii) Insertion iii) Deletion iv) Traversal
3. Write a program that uses functions to perform the following operations on doubly linked list.
i) Creation ii) Insertion iii) Deletion iv) Traversal
4. Write a program that implement Stack (its operations) using i) Arrays ii) Pointers
5. Write a program that implement Queue (its operations) using i) Arrays ii) Pointers
6. Write a program to implement the tree traversal methods (Recursive and Non Recursive)
7. Write a c program to implement binary search tree operations.
i) creation ii) insertion iii) deletion iv) searching
8. Write a program that implements the following sorting methods to sort a given list of integers in ascending order
i) Quick sort ii) Merge sort iii) Heap Sort
9. Write a program to implement the Breadth First Search graph traversal method.
10. Write a program to implement the Depth First Search graph traversal method.

Additional Programs:

1. Write a program that uses functions to perform the following operations on circular linked list:

- i) Creation ii) Insertion iii) Deletion iv) Traversal
2. Write a program to implement AVL Tree operations.

TEXT BOOKS:

1. "Data Structures Using C++" by D.S. Malik - A beginner-friendly textbook introducing data structures and their implementation in C++.
2. Fundamentals of Data Structures , 2nd Edition, E. Horowitz, S. Sahni and Susan Anderson.
3. Data Structures: A Pseudocode Approach with C, 2nd Edition, R. F. Gilberg and B.A. Forouzan, Cengage Learning.

REFERENCE BOOKS:

1. Data Structures – A. S. Tanenbaum, Y. Langsam, and M.J. Augenstein, PHI/Pearson.



R23 B.Tech CSE(AI & ML) Syllabus

AVNIET

OPERATING SYSTEMS LAB

L	T	P	C
0	0	2	1

B.Tech. II Year I Sem.

Prerequisites: A course on “Programming for Problem Solving”, A course on “Computer Organization and Architecture”.

Co-requisite: A course on “Operating Systems”.

Course Objectives:

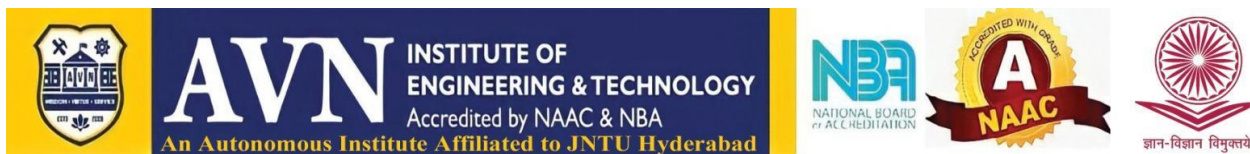
- To implement the scheduling algorithms.
- To write programs in Linux environment using the I/O system calls of UNIX/LINUX operating system.
- To implement deadlock avoidance
- To implement synchronization problems using semaphores using UNIX/LINUX system calls.
- To implement IPC mechanism using Pipes, FIFOs, Message Queues, Shared Memory.

Course Outcomes:

- Able to implement c programs for different CPU scheduling algorithms.
- Able to implement c programs for file and directory I/O system calls
- Able to implement c programs for prevention and avoidance of deadlocks.
- Able to implement c programs for process synchronization using semaphore and IPC mechanisms Using system calls.
- Able to develop c programs for paging and segmentation technique.

List of Experiments:

1. Write C programs to simulate the following CPU scheduling algorithms:
 - a) FCFS
 - b) SJF
 - c) Round Robin
 - d) Priority
2. Write programs using the I/O system calls of UNIX/LINUX operating system
3. Write a C program to simulate Bankers Algorithm for Deadlock Avoidance and Prevention.
4. Write a C program to implement the Producer – Consumer problem using semaphores using UNIX/LINUX system calls
5. Write C programs to illustrate the following IPC mechanisms
 - a) Pipes
 - b) FIFOs
 - c) Message Queues
 - d) Shared Memory
6. Write C programs to simulate the following memory management techniques:
 - a) Paging
 - b) Segmentation
7. Simulate page replacement Algorithms.
 - a) FIFO
 - b) LRU



R23 B.Tech CSE (AI & ML) Syllabus

AVNIET

GENDER SENSITIZATION LAB

B.Tech.II Year I Sem.

L	T	P	C
0	0	2	0

COURSE DESCRIPTION:

This course offers an introduction to Gender Studies, an interdisciplinary field that asks critical questions about the meanings of sex and gender in society. The primary goal of this course is to familiarize students with key issues, questions and debates in Gender Studies, both historical and contemporary. It draws on multiple disciplines – such as literature, history, economics, psychology, sociology, philosophy, political science, anthropology and media studies – to examine cultural assumptions about sex, gender, and sexuality.

This course integrates analysis of current events through student presentations, aiming to increase awareness of contemporary and historical experiences of women, and of the multiple ways that sex and gender interact with race, class, caste, nationality and other social identities. This course also seeks to build an understanding and initiate and strengthen programmes combating gender-based violence and discrimination. The course also features several exercises and reflective activities designed to examine the concepts of gender, gender-based violence, sexuality, and rights. It will further explore the impact of gender-based violence on education, health and development.

Objectives of the Course

- To develop students’ sensibility with regard to issues of gender in contemporary India.
- To provide a critical perspective on the socialization of men and women.
- To introduce students to information about some key biological aspects of genders.
- To expose the students to debates on the politics and economics of work.
- To help students reflect critically on gender violence.
- To expose students to more egalitarian interactions between men and women.

Learning Outcomes

- Students will have developed a better understanding of important issues related to gender in contemporary India.
- Students will be sensitized to basic dimensions of the biological, sociological, psychological and legal aspects of gender. This will be achieved through discussion of materials derived from research, facts, everyday life, literature and film.
- Students will attain a finer grasp of how gender discrimination works in our society and how to counter it.
- Students will acquire insight into the gendered division of labor and its relation to politics and economics.

- Men and women students and professionals will be better equipped to work and live together as equals.
- Students will develop a sense of appreciation of women in all walks of life.
- Through providing accounts of studies and movements as well as the new laws that provide protection and relief to women, the textbook will empower students to understand and respond to gender violence.

Unit-I: UNDERSTANDING GENDER

Introduction: Definition of Gender-Basic Gender Concepts and Terminology-Exploring Attitudes towards Gender-Construction of Gender-Socialization: Making Women, Making Men - Preparing for Womanhood. Growing up Male. First lessons in Caste.

Unit – II: GENDER ROLES AND RELATIONS

Two or Many? -Struggles with Discrimination-Gender Roles and Relations-Types of Gender Roles-Gender Roles and Relationships Matrix-Missing Women-Sex Selection and Its Consequences-Declining Sex Ratio. Demographic Consequences-Gender Spectrum: Beyond the Binary

Unit – III: GENDER AND LABOUR

Division and Valuation of Labour-Housework: The Invisible Labor- “My Mother doesn’t Work.” “Sharethe Load.”-Work: Its Politics and Economics -Fact and Fiction. Unrecognized and Unaccounted work.

-Gender Development Issues-Gender, Governance and Sustainable Development-Gender and Human Rights-Gender and Mainstreaming

Unit – IV: GENDER - BASED VIOLENCE

The Concept of Violence- Types of Gender-based Violence-Gender-based Violence from a Human Rights Perspective-Sexual Harassment: Say No!-Sexual Harassment, not Eve-teasing- Coping with Everyday Harassment- Further Reading: “*Chupulu*”.

Domestic Violence: Speaking OutIs Home a Safe Place? -When Women Unite [Film]. Rebuilding Lives. Thinking about Sexual Violence Blaming the Victim-“I Fought for my Life....”

Unit – V: GENDER AND CULTURE

Gender and Film-Gender and Electronic Media-Gender and Advertisement-Gender and Popular Literature- Gender Development Issues-Gender Issues-Gender Sensitive Language-Gender and Popular Literature - Just Relationships: Being Together as Equals

Mary Kom and Onler. Love and Acid just do not Mix. Love Letters. Mothers and Fathers. Rosa Parks-The Brave Heart.

Note: Since it is Interdisciplinary Course, Resource Persons can be drawn from the fields of English Literature or Sociology or Political Science or any other qualified faculty who has expertise in this field from engineering departments.

➤ **Classes will consist of a combination of activities: dialogue-based lectures, discussions, collaborative learning activities, group work and in-class assignments. Apart from the above prescribed book, Teachers can make use of any authentic materials related to the topics given in the syllabus on “Gender”.**

👉 **ESSENTIAL READING:** The Textbook, “Towards a World of Equals: A Bilingual Textbook on Gender” written by A.Suneetha, Uma Bhrugubanda, Duggirala Vasanta, Rama Melkote, Vasudha Nagaraj, Asma Rasheed, Gogu Shyamala, Deepa Sreenivas and Susie Tharu **published by Telugu Akademi, Telangana Government in 2015.**

ASSESSMENT AND GRADING:

- Discussion & Classroom Participation: 20%
- Project/Assignment: 30%
- End Term Exam: 50%



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING (AI&ML)

B.Tech Course Structure (R23 Regulations)

II B.Tech (II Sem)

Branch/Specialization: CSE (AI&ML)

S.No.	Course Code	Name of the Course	Contact Hours per Week			Credits
			L	T	P	
1	CS203	Object Oriented Programming through Java	3	0	0	3
2	CS205	Discrete Mathematics	3	0	0	3
3	CS219	Introduction to Artificial Intelligence	3	0	0	3
4	CS210	Database Management Systems	3	0	0	3
5	CS221	Automata Theory and Compiler Design	3	0	0	3
6	CS204	Java Programming Lab	0	0	2	1
7	CS211	Database Management Systems Lab	0	0	2	1
8	HS206	Verbal Communication and Soft Skills Lab	0	0	2	1
9	CS222	Real time research project/societal related project	0	0	4	2
10	MC201	Constitution of India	1	0	0	0
Total			16	0	10	20

L: Lecture Hour

T: Tutorial Hour

P: Lab/Practice Hour



OBJECT ORIENTED PROGRAMMING THROUGH JAVA

B.Tech. II Year II Sem

L	T	P	C
3	0	0	3

Course Objectives:

- This course is about the fundamentals of Object-Oriented Programming (OOP) Concept and OOP-based software development methodology.
- Java as a class - based and pure OOP language is used to demonstrate and implement appropriate concepts and techniques.
- The students are exposed to the concepts, fundamental syntax, and the thought processes behind object-oriented programming.
- By end of the course, students will acquire the basic knowledge and skills necessary to implement object-oriented programming techniques in software development using java.

Course outcomes:

- Upon completion of the course, the student will be able to achieve the following outcomes:
- Define, understand and differentiate the Object Oriented concepts and Java Programming concepts.
- Apply object oriented concepts on real time scenarios.
- Use Exception handling and multithreading mechanisms to create efficient software applications.
- Utilize modern tools and collection framework to create Java applications to solve real world problems.
- Design and develop GUI based applications using applets and swings for internet and system based applications.

UNIT - I

Object oriented thinking and Java Basics- Need for oop paradigm, summary of oop concepts, History of Java, Java buzzwords, data types, variables, scope and lifetime of variables, arrays, operators, expressions, control statements, type conversion and casting, simple java program, concepts of classes, objects, constructors, methods, access control, this keyword, garbage collection, overloading methods and constructors, parameter passing, recursion, nested and inner classes, exploring string class, string handling.

UNIT - II

Inheritance, Packages and Interfaces – Hierarchical abstractions, Base class object, subclass, subtype, substitutability, forms of inheritance specialization, specification, construction, extension,

limitation, combination, benefits of inheritance, costs of inheritance. Member access rules, super uses, using final with inheritance, polymorphism- method overriding, abstract classes, the Object class. Defining, Creating and Accessing a Package, Understanding CLASSPATH, importing packages, differences between classes and interfaces, defining an interface, implementing interface, applying interfaces, variables in interface and extending interfaces. Exploring java.io.

UNIT - III

Exception handling and Multithreading-- Concepts of exception handling, benefits of exception handling, Termination or resumptive models, exception hierarchy, usage of try, catch, throw, throws and finally, built in exceptions, creating own exception subclasses. Exploring java.util. Differences between multithreading and multitasking, thread life cycle, creating threads, thread priorities, synchronizing threads, inter thread communication, thread groups, daemon threads. Enumerations, autoboxing, annotations, generics.

UNIT - IV

Event Handling: Events, Event sources, Event classes, Event Listeners, Delegation event model, handling mouse and keyboard events, Adapter classes. The AWT class hierarchy, user interface components- labels, button, canvas, scrollbars, text components, check box, checkbox groups, choices, lists panels – scrollpane, dialogs, menubar, graphics, layout manager – layout manager types – border, grid, flow, card and grid bag.

UNIT - V

Applets – Concepts of Applets, differences between applets and applications, life cycle of an applet, types of applets, creating applets, passing parameters to applets. Swing – Introduction, limitations of AWT, MVC architecture, components, containers, exploring **swing**- JApplet, JFrame and JComponent, Icons and Labels, text fields, buttons – The JButton class, Check boxes, Radio buttons, Combo boxes, Tabbed Panes, Scroll Panes, Trees, and Tables.

TEXT BOOKS:

1. Java the complete reference, 7th edition, Herbert schildt, TMH.
2. Understanding OOP with Java, updated edition, T. Budd, Pearson education.

REFERENCE BOOKS:

1. An Introduction to programming and OO design using Java, J.Nino and F.A. Hosch, John wiley & sons.
2. An Introduction to OOP, third edition, T. Budd, Pearson education.
3. Introduction to Java programming, Y. Daniel Liang, Pearson education.
4. An introduction to Java programming and object-oriented application development, R.A. Johnson- Thomson.
5. Core Java 2, Vol 1, Fundamentals, Cay.S. Horstmann and Gary Cornell, eighth Edition, Pearson Education.
6. Core Java 2, Vol 2, Advanced Features, Cay.S. Horstmann and Gary Cornell, eighth Edition, Pearson Education
7. Object Oriented Programming with Java, R.Buyya, S.T.Selvi, X.Chu, TMH.
8. Java and Object Orientation, an introduction, John Hunt, second edition, Springer.
9. Maurach's Beginning Java2 JDK 5, SPD



R23 B.Tech CSE (AI & ML) Syllabus

AVNIET

DISCRETE MATHEMATICS

B.Tech. II Year II Sem

L	T	P	C
3	0	0	3

Course Objectives:

- Introduces elementary discrete mathematics for computer science and engineering.
- Topics include formal logic notation, methods of proof, induction, sets, relations, algebraic structures, elementary graph theory, permutations and combinations, counting principles; recurrence relations and generating functions.

Course Outcomes:

- Understand and construct precise mathematical proofs
- Apply logic and set theory to formulate precise statements
- Analyze and solve counting problems on finite and discrete structures
- Describe and manipulate sequences
- 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

Algebraic Structures: Semi-group and Monoids, Residue Arithmetic. Induction and Recursion: Mathematical Induction, Strong Induction and Well Ordering, Recursive Definitions and Structural Induction, Recursive Algorithms,

UNIT-IV

Elementary Combinatorics: Basics of Counting, Combinations and Permutations, Enumeration of Combinations and Permutations, Enumerating Combinations and Permutations with Repetitions, Enumerating Permutation with Constrained Repetitions, Binomial Coefficient, The Binomial and Multinomial Theorems, The Principle of Excl.

UNIT -V

Graph Theory: Basic Concepts, Isomorphism and Sub graphs, Trees and their Properties, Spanning Trees, Directed Trees, Binary Trees, Planar Graphs, Euler's Formula, Multi-graphs and Euler Circuits, Hamiltonian Graphs, Chromatic Numbers, The Four-Color Problem.

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 BOOKS:

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.



R23 B.Tech CSE (AI & ML) Syllabus

AVNIET

INTRODUCTION TO ARTIFICIAL INTELLIGENCE

B.Tech. II Year II Sem

L	T	P	C
3	0	0	3

Prerequisite: Knowledge on Data Structures.

Course Objectives:

- To learn the distinction between optimal reasoning Vs. human like reasoning.
- To understand the concepts of state space representation, exhaustive search, heuristic
- Search together with the time and space complexities.
- To learn different knowledge representation techniques.
- To understand the applications of AI, namely game playing, theorem proving, and machine learning.

Course Outcomes:

- Learn the distinction between optimal reasoning Vs human like reasoning and formulate an efficient problem space for a problem expressed in natural language. Also select a search algorithm for a problem and estimate its time and space complexities.
- Apply AI techniques to solve problems of game playing, theorem proving, and machine learning.
- Learn different knowledge representation techniques.
- Understand the concepts of state space representation, exhaustive search, heuristic search together with the time and space complexities.
- Comprehend the applications of Probabilistic Reasoning and Bayesian Networks.
- Analyze Supervised Learning Vs. Learning Decision Trees

UNIT - I

Introduction to AI - Intelligent Agents, Problem-Solving Agents,

Searching for Solutions - Breadth-first search, Depth-first search, Hill-climbing search, Simulated

Annealing search, Local Search in Continuous Spaces.

UNIT-II

Games - Optimal Decisions in Games, Alpha–Beta Pruning, Defining Constraint Satisfaction Problems, Constraint Propagation, Backtracking Search for CSPs, Knowledge-Based Agents, Logic-Propositional Logic,.

Propositional Theorem Proving: Inference and proofs, Proof by resolution, Horn clauses and definite clauses.

UNIT-III

First-Order Logic - Syntax and Semantics of First-Order Logic, Using First Order Logic, Knowledge Engineering in First-Order Logic, Inference in First-Order Logic: Propositional vs. First-Order Inference, Unification, Forward Chaining, Backward Chaining, Resolution.

Knowledge Representation: Ontological Engineering, Categories and Objects, Events.

UNIT-IV

Planning - Definition of Classical Planning, Algorithms for Planning with State Space Search, Planning Graphs, other Classical Planning Approaches, Analysis of Planning approaches. Hierarchical Planning.

UNIT-V

Probabilistic Reasoning:

Acting under Uncertainty, Basic Probability Notation Bayes' Rule and Its Use, Probabilistic Reasoning,

Representing Knowledge in an Uncertain Domain, The Semantics of Bayesian Networks, Efficient Representation of Conditional Distributions, Approximate Inference in Bayesian Networks, Relational and First- Order Probability.

TEXT BOOK:

1. Artificial Intelligence: A Modern Approach, Third Edition, Stuart Russell and Peter Norvig, Pearson Education.

REFERENCE BOOKS:

1. Artificial Intelligence, 3rd Edn., E. Rich and K. Knight (TMH)
2. Artificial Intelligence, 3rd Edn., Patrick Henny Winston, Pearson Education.
3. Artificial Intelligence, Shivani Goel, Pearson Education.
4. Artificial Intelligence and Expert systems – Patterson, Pearson Education.



R23 B.Tech CSE (AI & ML) Syllabus

AVNIET

DATABASE MANAGEMENT SYSTEMS

B.Tech. II Year II Sem

L	T	P	C
3	0	0	3

Prerequisites: A course on “Data Structures”.

Course Objectives:

- To understand the basic concepts and the applications of database systems.
- To master the basics of SQL and construct queries using SQL.
- Topics include data models, database design, relational model, relational algebra, transaction control, concurrency control, storage structures and access techniques.

Course Outcomes:

- Gain knowledge of fundamentals of DBMS, database design.
- To Know the basic concepts of Relational Model and Relational Algebra.
- Gain knowledge of normal forms and the basics of SQL for retrieval and management of data.
- Be acquainted with the basics of transaction processing and concurrency control.
- Familiarity with database storage structures and access techniques.

UNIT - I:

Overview of Database Systems: A Historical Perspective, File Systems versus a DBMS, the Data Models-Hierarchical Model, Network Model, Entity-Relational Model, Relational Model, Levels of Abstraction in a DBMS, Data Independence, Structure of a DBMS

Introduction to Database Design: Database Design and ER Diagrams, Entities, Attributes, and Entity Sets, Relationships and Relationship Sets, Additional Features of the ER Model, Conceptual Design with the ER Model

UNIT - II:

Introduction to the Relational Model: Integrity constraint over relations, enforcing integrity constraints, querying relational data, logical database design, introduction to views, destroying/altering tables and views. Relational Algebra, Tuple relational Calculus, Domain relational calculus.

UNIT – III:

SQL: QUERIES, CONSTRAINTS, TRIGGERS: form of basic SQL query, UNION, INTERSECT, and EXCEPT, Nested Queries, aggregation operators, NULL values, complex integrity constraints in SQL, triggers and active databases.

Schema Refinement: Problems caused by redundancy, decompositions, problems related to decomposition, Functional Dependencies, Reasoning about FDs, First, Second, Third normal forms, BCNF, lossless join decomposition, multivalued dependencies, Fourth normal form, Fifth normal form.

UNIT – IV:

Transaction Concept, Transaction State, Implementation of Atomicity and Durability, Concurrent Executions, Serializability, Recoverability, Implementation of Isolation, Testing for serializability, Lock Based Protocols, Timestamp Based Protocols, Validation - Based Protocols, Recovery and Atomicity, Log-Based Recovery, Recovery with Concurrent Transactions.

UNIT – V:

Data on External Storage, File Organization and Indexing, Cluster Indexes, Primary and Secondary Indexes, Index data Structures, Indexed Sequential Access Methods (ISAM).

TEXT BOOKS:

1. Database System Concepts, Silberschatz, Korth, McGraw hill, V edition.3rd Edition
2. Database Management Systems, Raghurama Krishnan, Johannes Gehrke, Tata Mc Graw Hill

REFERENCE BOOKS:

1. Database Systems design, Implementation, and Management, Peter Rob & Carlos Coronel 7th Edition.
2. Fundamentals of Database Systems, Elmasri Navrate, Pearson Education
3. Introduction to Database Systems, C. J. Date, Pearson Education
4. Oracle for Professionals, The X Team, S.Shah and V. Shah, SPD.
5. Database Systems Using Oracle: A Simplified guide to SQL and PL/SQL, Shah, PHI.
6. Fundamentals of Database Management Systems, M. L. Gillenson, Wiley Student Edition.



R23 B.Tech.CSE(AI&ML) Syllabus

AVNIET

AUTOMATA THEORY AND COMPILER DESIGN

L	T	P	C
3	0	0	3

B.Tech.II Year II sem.

Course Objectives

- To introduce the fundamental concepts of formal languages, grammars and automata theory.
- To understand deterministic and non-deterministic machines and the differences between decidability and undecidability.
- Introduce the major concepts of language translation and compiler design and impart the knowledge of practical skills necessary for constructing a compiler.
- Topics include phases of compiler, parsing, syntax directed translation, type checking use of symbol tables, intermediate code generation

Course Outcomes

- Able to employ finite state machines for modeling and solving computing problems.
- Able to design context free grammars for formal languages.
- Able to distinguish between decidability and undecidability.
- Demonstrate the knowledge of patterns, tokens & regular expressions for lexical analysis.
- Acquire skills in using lex tool and design LR parsers

UNIT - I

Introduction to Finite Automata: Structural Representations, Automata and Complexity, the Central Concepts of Automata Theory – Alphabets, Strings, Languages, Problems.

Nondeterministic Finite Automata: Formal Definition, an application, Text Search, Finite Automata with Epsilon-Transitions.

Deterministic Finite Automata: Definition of DFA, How A DFA Process Strings, The language of DFA, Conversion of NFA with ϵ -transitions to NFA without ϵ -transitions. Conversion of NFA to DFA

UNIT - II

Regular Expressions: Finite Automata and Regular Expressions, Applications of Regular Expressions, Algebraic Laws for Regular Expressions, Conversion of Finite Automata to Regular Expressions.

Pumping Lemma for Regular Languages: Statement of the pumping lemma, Applications of the Pumping Lemma.

Context-Free Grammars: Definition of Context-Free Grammars, Derivations Using a Grammar, Leftmost and Rightmost Derivations, the Language of a Grammar, Parse Trees, Ambiguity in Grammars and Languages.

UNIT - III

Push Down Automata: Definition of the Pushdown Automaton, the Languages of a PDA, Equivalence of

PDA and CFG's, Acceptance by final state

Turing Machines: Introduction to Turing Machine, Formal Description, Instantaneous description, The language of a Turing machine

Undecidability: Undecidability, A Language that is Not Recursively Enumerable, An Undecidable Problem That is RE, Undecidable Problems about Turing Machines

UNIT - IV

Introduction: The structure of a compiler,

Lexical Analysis: The Role of the Lexical Analyzer, Input Buffering, Recognition of Tokens, The Lexical- Analyzer Generator Lex,

Syntax Analysis: Introduction, Context-Free Grammars, Writing a Grammar, Top-Down Parsing, Bottom- Up Parsing, Introduction to LR Parsing: Simple LR, More Powerful LR Parsers

UNIT - V

Syntax-Directed Translation: Syntax-Directed Definitions, Evaluation Orders for SDD's, Syntax-Directed Translation Schemes, Implementing L-Attributed SDD's.

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

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

TEXT BOOKS:

1. Introduction to Automata Theory, Languages, and Computation, 3rd Edition, John E. Hopcroft, Rajeev Motwani, Jeffrey D. Ullman, Pearson Education.
2. Theory of Computer Science – Automata languages and computation, Mishra and Chandrashekar, 2nd Edition, PHI.

REFERENCE BOOKS:

1. Compilers: Principles, Techniques and Tools, Alfred V. Aho, Monica S. Lam, Ravi Sethi, Jeffrey D. Ullman, 2nd Edition, Pearson.
2. Introduction to Formal languages Automata Theory and Computation, Kamala Krithivasan, Rama R, Pearson.
3. Introduction to Languages and The Theory of Computation, John C Martin, TMH.
4. lex & yacc – John R. Levine, Tony Mason, Doug Brown, O'reilly Compiler Construction, Kenneth C. Loudon, Thomson. Course Technology.



R23 B.Tech CSE (AI & ML) Syllabus

AVNIET

DATABASE MANAGEMENT SYSTEMS LAB

B.Tech. II Year II Sem

L	T	P	C
0	0	2	1

Co-requisites: “Database Management Systems”

Course Objectives:

- Introduce ER data model, database design and normalization
- Learn SQL basics for data definition and data manipulation

Course Outcomes:

- Design database schema for a given application and apply normalization
- Acquire skills in using SQL commands for data definition and data manipulation.
- Develop solutions for database applications using procedures, cursors and triggers

List of Experiments:

1. Concept design with E-R Model
2. Relational Model
3. Normalization
4. Practicing DDL commands
5. Practicing DML commands
6. A. Querying (using ANY, ALL, UNION, INTERSECT, JOIN, Constraints etc.) B. Nested, Correlated sub queries
7. Queries using Aggregate functions, GROUP BY, HAVING and Creation and dropping of Views.
8. Triggers (Creation of insert trigger, delete trigger, update trigger)
9. Procedures
10. Usage of Cursors

TEXT BOOKS:

1. Database Management Systems, Raghurama Krishnan, Johannes Gehrke, Tata Mc Graw Hill, 3rd Edition
2. Database System Concepts, Silberschatz, Korth, McGraw Hill, V edition.

REFERENCE BOOKS:

1. Database Systems design, Implementation, and Management, Peter Rob & Carlos Coronel 7th Edition.
2. Fundamentals of Database Systems, Elmasri Navrate, Pearson Education
3. Introduction to Database Systems, C.J. Date, Pearson Education
4. Oracle for Professionals, The X Team, S. Shah and V. Shah, SPD.
5. Database Systems Using Oracle: A Simplified guide to SQL and PL/SQL, Shah, PHI.
6. Fundamentals of Database Management Systems, M. L. Gillenson, Wiley Student Edition.



R23 B.Tech CSE (AI & ML) Syllabus

AVNIET

JAVA PROGRAMMING LAB

B.Tech. II Year II Sem

L	T	P	C
0	0	2	1

Course Objectives:

- To write programs using abstract classes.
- To write programs for solving real world problems using java collection framework.
- To write multithreaded programs.
- To write GUI programs using swing controls in Java.
- To introduce java compiler and eclipse platform.
- To impart hands on experience with java programming.

Course Outcomes:

- Able to write programs for solving real world problems using java collection framework.
- Able to write programs using abstract classes.
- Able to write multithreaded programs.
- Able to write GUI programs using swing controls in Java.

LIST OF EXPERIMENTS:

1. Use Eclipse or Net bean platform and acquaint yourself with the various menus. Create a test project, add a test class, and run it. See how you can use auto suggestions, auto fill. Try code formatter and code refactoring like renaming variables, methods, and classes. Try debug step by step with a small program of about 10 to 15 lines which contains at least one if else condition and a for loop.
2. Write a Java program that works as a simple calculator. Use a grid layout to arrange buttons for the digits and for the +, -, *, % operations. Add a text field to display the result. Handle any possible exceptions like divided by zero.
3. A) Develop an applet in Java that displays a simple message. B) Develop an applet in Java that receives an integer in one text field, and computes its factorial Value and returns it in another text field, when the button named “Compute” is clicked.

4. Write a Java program that creates a user interface to perform integer divisions. The user enters two numbers in the text fields, Num1 and Num2. The division of Num1 and Num 2 is displayed in the Result field when the Divide button is clicked. If Num1 or Num2 were not an integer, the program would throw a Number Format Exception. If Num2 were Zero, the program would throw an Arithmetic Exception. Display the exception in a message dialog box.
5. Write a Java program that implements a multi-thread application that has three threads. First thread generates a random integer every 1 second and if the value is even, the second thread computes the square of the number and prints. If the value is odd, the third thread will print the value of the cube of the number.
6. Write a Java program for the following: Create a doubly linked list of elements.Delete a given element from the above list. Display the contents of the list after deletion.
7. Write a Java program that simulates a traffic light. The program lets the user select one of three lights: red, yellow, or green with radio buttons. On selecting a button, an appropriate message with “Stop” or “Ready” or “Go” should appear above the buttons in the selected color. Initially, there is no message shown.
8. Write a Java program to create an abstract class named Shape that contains two integers and an empty method named print Area (). Provide three classes named Rectangle, Triangle, and Circle such that each one of the classes extends the class Shape. Each one of the classes contains only the method print Area () that prints the area of the given shape.
9. Suppose that a table named Table.txt is stored in a text file. The first line in the file is the header, and the remaining lines correspond to rows in the table. The elements are separated by commas. Write a java program to display the table using Labels in Grid Layout.
10. Write a Java program that handles all mouse events and shows the event name at the center of the window when a mouse event is fired (Use Adapter classes).
11. Write a Java program that loads names and phone numbers from a text file where the data is organized as one line per record and each field in a record are separated by a tab (\t). It takes a name or phone number as input and prints the corresponding other value from the hash table (hint: use hash tables).
12. Write a Java program that correctly implements the producer – consumer problem using the concept of inter thread communication.
13. Write a Java program to list all the files in a directory including the files present in all its subdirectories.

REFERENCE BOOKS:

1. Java for Programmers, P. J. Deitel and H. M. Deitel, 10th Edition Pearson education.
2. Thinking in Java, Bruce Eckel, Pearson Education.
3. Java Programming, D. S. Malik and P. S. Nair, Cengage Learning.
4. Core Java, Volume 1, 9th edition, Cay S. Horstmann and G Cornell, Pearson.



R23 B.Tech CSE (AI & ML) Syllabus

AVNIET

Verbal Communication and Soft Skills Lab

B.Tech. II Year II Sem

L	T	P	C
0	0	2	1

Course Objectives:

- To equip the students with common employability skills (the skills required for gaining employment and performing successfully in different careers) which can enable them to perform communication tasks of increasing length and complexity.
- To develop in them the interactional communication strategies and social graces which have the potential to add to the effectiveness of professional communication.

Course Outcomes:

- Upon successful completion of Professional Communication, the students will be able to
- Speak with a reasonable degree of fluency and accuracy in professional communication situations (such as arriving at a consensus through discussion, making a presentation, and taking part in a telephone conversation)
- Add to the effectiveness of their oral communication by using communication strategies, conventions of politeness and courtesy, and stress and intonation.
- Listen to short audio and video clips in native English accent (British and American), and gain both understanding of messages and sensitivity to native-speaker accents
- Read fluently, comprehending texts of different kinds using multiple strategies and higher-order skills
- Produce written discourses of different kinds (e.g. texts expressing opinions and making a convincing case for one's standpoint, professional emails, and summaries of lengthy texts) with attention to elements of writing such as content, organization, language, style, and mechanics guard against grammatical errors Indians typically make in their speech and writing in English

VERBAL COMMUNICATION AND SOFT SKILLS

1. WORD GAMES (Word connect ; word collect ;word formation)
2. TIME MANAGEMENT (Goal-setting; decision making; planning; delegation; stress management)
3. TEAM WORK (Team building ;leadership; co-ordination)

4. INFORMATION TRANSFER (Pictorial presentation; verbal to non-verbal ;non-verbal to verbal; collection ;storage; dissemination and retrieval of information)
5. CRITICAL THINKING AND PROBLEM SOLVING (Observation skills; Open-mindedness reasoning; skepticism; clarity and precision)
6. BLOG WRITING (Proofread and editing; description; concise paragraphs)
7. CRITICAL THINKING AND PROBLEM SOLVING (Observation skills; Open-mindedness; reasoning; skepticism; clarity and precision)
8. INTER-PERSONAL AND PUBLIC COMMUNICATION (Informative; persuasive and demonstrative speaking)
9. DEBATE (spoken and written)

CONSTITUTION OF INDIA

B.Tech. II Year, I Sem.

L	T	P	C
1	0	0	0

Course Objectives: Students will be able to:

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

Course Outcomes: Students will be able to:

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

Unit - 1

History of Making of the Indian Constitution- History of Drafting Committee.
Philosophy of the Indian Constitution- Preamble Salient Features.

Unit – 2

Contours of Constitutional Rights & Duties - Fundamental Rights

- Right to Equality
- Right to Freedom

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

Unit - 3 Organs of Governance: Parliament, Composition, Qualifications and Disqualifications, Powers and Functions, Executive, President, Governor, Council of Ministers, Judiciary, Appointment and Transfer of Judges, Qualifications, Powers and Functions

Unit - 4 Local Administration: District's Administration head: Role and Importance, Municipalities: Introduction, Mayor and role of Elected Representative, CEO of Municipal Corporation. Panchayat raj: Introduction, PRI: Zila Panchayat. Elected officials and their roles, CEO ZilaPanchayat: Position and role. Block level: Organizational Hierarchy (Different departments), Village level: Role of Elected and Appointed officials, Importance of grass root democracy

Unit - 6 Election Commission: Election Commission: Role and Functioning. Chief Election Commissioner and Election Commissioners. State Election Commission: Role and Functioning. Institute and Bodies for the welfare of SC/ST/OBC and women.

Suggested Reading:

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