Second Year (Lateral Entry) Curriculum

Admission Year 2026-27

Bachelor of Technology (Lateral Entry) Computer Science and Engineering

Faculty of Engineering & Technology

Parul University

Vadodara, Gujarat, India

Semester III

Course Name: Design of Data Structures

Course Code: 03010503PC01

Prerequisite: Basic knowledge of programming concepts using C language

Rationale: Data Structures form the foundation of efficient software development and problem-solving. This course introduces students to various linear and non-linear data structures and their practical implementation techniques.

Course Learning Objectives:

CLOBJ 1	Understand the fundamental concepts and classification of data structures.
CLOBJ 2	Apply arrays, stacks, queues, linked lists, trees, graphs, and hashing techniques for problem-solving.
CLOBJ 3	Analyze and implement searching and sorting algorithms.
CLOBJ 4	Develop programs using dynamic memory allocation and pointer-based data structures.

Course Learning Outcomes:

CLO 1	Explain the concepts, classification, and operations of various data structures and analyze algorithm complexity.
CLO 2	Implement stacks, queues, and recursion techniques to solve computational problems efficiently.
CLO 3	Develop and manipulate linked list data structures for dynamic data storage applications.
CLO 4	Apply searching and sorting algorithms to organize and retrieve data efficiently.
CLO 5	Construct and perform operations on binary trees and binary search trees.

Teaching & Examination Scheme:

Teaching Scheme				**Evaluation Scheme**					
L	**T**	**P**	**C**	**Internal Evaluation**			**ESE**		**Total**
				MSE	**CE**	**P**	**Theory**	**P**	
3	**0**	**0**	**3**	**20**	**20**		**60**	**40**	**100**

L- Lectures; T- Tutorial; P- Practical; C- Credit; MSE- Mid-Semester Evaluation, CE- Continuous Evaluation, ESE- End Semester Examination

Course Content:

Sr.No.	**Content**	**Weightage**	**Teaching Hours**
1	**Introduction:** Data Structures, Classifications (Primitive & Non-Primitive), Data structure Operations, Review of Arrays, Structures, Self-Referential Structures, and Unions. Pointers and Dynamic Memory Allocation Functions. Representation of Linear Arrays in Memory, dynamically allocated arrays. Performance analysis of an algorithm and space and time complexities	**10**	**6**
2	**Stacks, Recursion and Queue: Stacks:** Definition, Stack Operations, Array Representation of Stacks, Stacks using Dynamic Arrays, Stack Applications: Polish notation, Infix to postfix conversion, evaluation of postfix expression. **Recursion** - Factorial, GCD, Fibonacci Sequence, Tower of Hanoi, **Queues:** Definition, Array Representation, Queue Operations, Circular Queues, Circular queues using Dynamic arrays, Deque, Priority Queues and its Application.	**15**	**8**
3	**Linked Lists:** Definition, Representation of linked lists in Memory, Memory allocation; Garbage Collection. Linked list operations: Traversing, Searching, Insertion, and Deletion In Singly Linked List, Doubly Linked lists, Circular linked lists. Implementation of Stacks and Queues using Linked List.	**10**	**5**
4	**Searching and Sorting: Searching:** Linear Search, Binary Search, and Interpolation Search. **Sorts:** Selection Sort, Insertion Sort, Bubble Sort, Quick Sort, Merge Sort.	**10**	**5**
5	**Trees:** Terminology, Binary Trees, Properties of Binary trees, Array and linked Representation of Binary Trees, Binary Tree Traversals - In Order, Post Order, Pre Order; Binary Search Trees – Definition, Insertion, Deletion, Traversal, Searching, Height of tree, Evaluation of Expression, Balanced Binary Tree and its operations.	**15**	**9**

List of Practicals:

1.	**Implement Stack and its operations like (creation push pop traverse peek search) using linear data structure**
2.	**Implement Infix to Postfix Expression Conversion using Stack**
3.	**Implement Postfix evaluation using Stack.**
4.	**Implement Towers of Hanoi using Stack.**
5.	**Implement queue and its operations like enqueue, dequeue, traverse, search.**
6.	**Implement Single Linked lists and its operations(creation insertion deletion traversal search reverse)**
7.	**Implement Double Linked lists and its operations(creation insertion deletion traversal search reverse)**
8.	**Implement Circular Linked lists and its operations(creation insertion deletion traversal search reverse)**
9.	**Implement Linear Search, binary search and interpolation search.**
10.	**Implement Bubble sort, selection sort, Insertion sort, quick sort, merge sort.**

Text Book and Reference Book:

1.	**Fundamentals of Data Structures in C, 2ND Edition**, E.Horowitz, S. Sahni and Susan Anderson-Freed, Universities Press (TextBook)
2.	**Seymour Lipschutz, Data Structures Schaum's Outlines, Revised 1st Ed, McGraw Hill, 2014.**

(2)

Course Name: Java Programming

Course Code: 03010503PC03

Prerequisite: Basic knowledge of computer programming concepts

Rationale: Java is one of the most widely used object-oriented programming languages for developing platform-independent, secure, and robust applications.

Course Learning Objectives:

CLOBJ 1	Understand the fundamentals of object-oriented programming and Java platform architecture.
CLOBJ 2	Develop Java programs using variables, operators, control statements, arrays, and strings.
CLOBJ 3	Apply object-oriented principles such as inheritance, polymorphism, abstraction, and encapsulation in software development.
CLOBJ 4	Implement exception handling mechanisms for reliable and robust applications.

Course Learning Outcomes:

CLO 1	Explain Java architecture, JVM, JDK, JRE, and object-oriented programming concepts.
CLO 2	Develop Java programs using data types, operators, control statements, arrays, and strings.
CLO 3	Design applications using classes, objects, constructors, inheritance, polymorphism, abstraction, and interfaces.
CLO 4	Implement exception handling techniques to create reliable Java applications.
CLO 5	Apply Java Collection Framework and Generics for efficient storage and manipulation of data.

Teaching & Examination Scheme:

Teaching Scheme				**Evaluation Scheme**					
L	**T**	**P**	**C**	**Internal Evaluation**			**ESE**		**Total**
				MSE	**CE**	**P**	**Theory**	**P**	
4	**-**	**-**	**4**	**20**	**20**	**-**	**60**	**-**	**100**

L- Lectures; T- Tutorial; P- Practical; C- Credit; MSE- Mid-Semester Evaluation, CE- Continuous Evaluation, ESE- End Semester Examination

Course Content:

Sr.No.	**Content**	**Weightage**	**Teaching Hours**
1	**Overview of OOP and Java:** Programming Paradigms, Procedural vs Object-Oriented Programming, OOP Features, History of Java, Feature of Java, JVM Architecture, JRE, JDK, Java Program Structure, Compilation and Execution Process, Java Environment Setup, JDK Installation, PATH and CLASSPATH, Introduction to IDEs.	**10**	**4**

2	**Java Language Basics:** Variables, Scope of Variables, Primitive Data Types, Reference Data Types, Type Casting, Wrapper Classes, Autoboxing and Unboxing, Arithmetic Operators, Relational Operators, Logical Operators, Assignment Operators, Unary Operators, Bitwise Operators, Operator Precedence, if Statement, if-else Statement, Nested if, switch Statement, for Loop, while Loop, do-while Loop, for-Each Loop, break Statement, continue Statement.	**15**	**7**
3	**Arrays and String:** Arrays, One-Dimensional Arrays, Multi-Dimensional Arrays, Array Operations, String Class, StringBuffer, StringBuilder, String Comparison, String Manipulation Methods.	**17**	**6**
4	**Object-Oriented Programming (OOP):** Classes and Objects, Constructors, Access Specifiers, Encapsulation, Inheritance, Types of Inheritance, super Keyword, Polymorphism, Method Overloading (Compile time Polymorphism), Method Overriding (Runtime Polymorphism), Abstraction, Abstract Classes, Interfaces, Functional Interfaces, this Keyword, final Keyword, Object Class and its Methods.	**30**	**12**
5	**Exception Handling:** Errors and Exceptions, Checked Exceptions, Unchecked Exceptions, try-catch Block, Multiple catch Blocks, finally Block, throw Keyword, throws Keyword, Custom Exceptions, Exception Handling, Best Practices.	**15**	**6**
6	**Collection Framework in Java:** Introduction to Collections Framework, List Interface, ArrayList, LinkedList, Set Interface, HashSet, TreeSet, Map Interface, HashMap, TreeMap, Iterators, Generics.	**10**	**5**
7	**Advanced Foundations and Java Frameworks:** Process Vs Thread, Thread Creation, Thread Life Cycle, Multithreading Concepts, Synchronization, Inter-thread Communication, File Handling, File Class, Byte Streams, Character Streams, Buffered Streams, Serialization, Deserialization, Lambda Expressions, Stream API, Optional Class.	**8**	**5**
	Total	**100**	

List of Practicals:

List of Practical	
1.	**Write a Java program to display "Hello World" and demonstrate the Java program structure and demonstrate use of variables, data types, and type casting.**
2.	**Write a Java program to perform arithmetic, relational, and logical operations, bitwise and other operators.**
3.	**Write a Java program to: Even or Odd using if..else statement. Roots of Quadratic Equation using else if ladder. Largest of three numbers using nested if else. Find out the week day using switch statement.**
4.	**Write a Java program to demonstrate looping constructs: Reverse of a number using while loop. Prime number using do while loop. nth term of fibonacci sequence using for loop.**
5.	**Write a Java program to perform methods and example on recursion.**
6.	**Write a Java program to perform operations on one-dimensional and multi-dimensional arrays.**
7.	**Write a Java program to demonstrate 1D-Arrays & 2-D Arrays. Maximum value and Second Maximum value without duplicates. Sort the names in Ascending Order. Addition of two matrix. 3x3 Matrix Multiplication.**
8.	**Write a Java program to demonstrate string handling using String, StringBuffer, and StringBuilder.**
9.	**Write a Java program to check the word is palindrome or not.**
10.	**Write a Java program to create a class and object and demonstrate constructors.**

Text Book and Reference Book:

Reference Books	
1.	***Java: The Complete Reference*** — Herbert Schildt Publisher: **McGraw-Hill Education** (Latest 13th Edition)
2.	***Object Oriented Programming Through Java*** — P. Radha Krishna Publisher: **Universities Press (India) Pvt. Ltd.**
3.	***Thinking in Java*** — Bruce Eckel Publisher: **Prentice Hall (Pearson)**
4.	***Core Java, Volume I & Volume II*** — Cay S. Horstmann & Gary Cornell Publisher: **Prentice Hall / Pearson**
5.	***JAVA8 Core Java Black Book*** — R. Nageswara Rao Publisher: **Black Book Series (Dreamtech Press)**

Course Name: Operating Systems

Course Code: 03010503PC05

Prerequisite: Fundamentals of Computer Systems

Rationale: This course is an introduction to the theory and practice behind modern computer operating systems. Topics will include what an operating systemdoes (and doesn't) do, system calls and interfaces, processes, concurrent programming, resource
scheduling and management, virtual memory, deadlocks, algorithms , programming, and security. The approach of the subject is from both a theoretical perspective as well as a practical one.

Course Learning Objectives:

CLOBJ 1	Understand the fundamentals of object-oriented programming and Java platform architecture.
CLOBJ 2	Develop Java programs using variables, operators, control statements, arrays, and strings.
CLOBJ 3	Apply object-oriented principles such as inheritance, polymorphism, abstraction, and encapsulation in software development.
CLOBJ 4	Implement exception handling mechanisms for reliable and robust applications.

Course Learning Outcomes:

After completing the course, students shall be able to:

CLO 1	Distinguish different styles of operating systemdesign.
CLO 2	Understand device and I/O management functions in operating systems as part of a uniform device abstraction.
CLO 3	Understand disk organization and file system structure.
CLO 4	Give the rationale for virtual memory abstractions in operating systems.
CLO 5	Understand the main principles and techniques used to implement processes and threads as well as the different algorithms for process scheduling.
CLO 6	Understand the main mechanisms used for inter-process communication.

Teaching & Examination Scheme:

Teaching Scheme				**Seminar Hrs/Week**	**Credit**	**Examination Scheme**		
Lecture Hrs/Week	**Tutorial Hrs/Week**	**Lab Hrs/Week**				**Internal Marks**		**External Marks**
						T	**CE**	**T**
3	**0**	**0**	**0**	**0**	**3**	**20**	**20**	**60**

SEE - Semester End Examination, T - Theory, P - Practical

Course Content:

Sr. No.	**Content**	**Weightage (%)**	**Teaching Hours**
1	**INTRODUCTION:** Concept of Operating Systems, Generations of Operating systems, Types of Operating Systems, OS Services, System Calls, Structure of an OS – Layered, Monolithic, Microkernel Operating Systems, Concept of Virtual Machine.	**10**	**4**
2	**PROCESSES, THREAD & PROCESS SCHEDULING:** Processes: Definition, Process Relationship, Different states of a Process, Process State transitions, Process Control Block (PCB), Context switching. Thread: Definition, Various states, Benefits of threads, Types of threads, Concept of multithreads. Process Scheduling: Foundation and Scheduling objectives, Types of Schedulers, Scheduling criteria: CPU utilization, Throughput, Turnaround Time, Waiting Time, Response Time; Scheduling algorithms: Pre-emptive and Non pre-emptive, FCFS, SJF, RR.	**23**	**11**
3	**INTER-PROCESS COMMUNICATION:** Critical Section, Race Conditions, Mutual Exclusion, Hardware Solution, Strict Alternation, Peterson's Solution, The Producer/Consumer Problem, Semaphores, Event Counters, Monitors, Message Passing, Classical IPC Problems: Reader's & Writer Problem, Dining Philosopher Problem etc.	**15**	**7**
4	**DEADLOCKS:** Definition, Necessary and sufficient conditions for Deadlock, Deadlock Prevention, Deadlock Avoidance: Banker's algorithm, Deadlock detection and Recovery.	**12**	**5**

5	**MEMORY MANAGEMENT & VIRTUAL MEMORY:** Memory Management: Basic concept, Logical and Physical address map, Memory allocation: Contiguous Memory allocation, Fixed and variable partition, Internal and External fragmentation and Compaction; Paging: Page allocation, Hardware support for paging, Protection and sharing, Disadvantages of paging. Virtual Memory: Basics of Virtual Memory, Hardware and control structures, Locality of reference, Page fault, Working Set, Dirty page/Dirty bit, Demand paging, Page Replacement algorithms: Optimal, FIFO, Second Chance (SC), Not Recently Used (NRU) and Least Recently Used (LRU).	**20**	**10**
6	**I/O SYSTEMS, FILE & DISK MANAGEMENT:** I/O Hardware: I/O devices, Device controllers, Direct memory access. Principles of I/O Software: Goals of Interrupt handlers, Device drivers, Device independent I/O software. File Management: Concept of File, Access methods, File types, File operation, Directory structure, File System structure, Allocation methods (contiguous, linked, indexed), Free-space management (bit vector, linked list, grouping), directory implementation (linear list, hash table),	**20**	**8**

	efficiency and performance. Disk Management: Disk structure, Disk scheduling algorithms – FCFS, SSTF, SCAN, C-SCAN, Disk reliability, Disk formatting, Boot-block, Bad blocks.		
	Total	**100**	**45**

List of Practicals:

Sr. No.	Practical
1.	Study of Basic commands of Linux.
2.	Study the basics of shell programming.
3.	Write a Shell script to print given numbers sum of all digits.
4.	Write a shell script to validate the entered date. (e.g. Date format is: dd-mm-yyyy).
5.	Write a shell script to check entered string is palindrome or not.
6.	Write a Shell script to say Good morning/Afternoon/Evening as you log in to system.
7.	Write a C program to create a child process.
8.	Finding out biggest number from given three numbers supplied as command line arguments.
9.	Printing the patterns using for loop.
10.	Shell script to determine whether given file exist or not.
11.	Write a program for process creation using C. (Use of gcc compiler).
12.	Implementation of FCFS & Round Robin Algorithm.
13.	Implementation of Banker's Algorithm.

Text Book and Reference Books:

1.	**Operating System Concepts Essentials (TextBook)** – Avi Silberschatz, Peter Galvin, Greg Gagne | 9th Edition, Wiley Asia Student Edition.
2.	**Operating Systems Internals and Design Principles** – William Stallings | PHI | 5th Edition.
3.	**Operating System: A Design-oriented Approach** – Charles Crowley | 1st Edition, Irwin Publishing.
4.	**Operating Systems: A Modern Perspective** – Gary J. Nutt | Addison-Wesley | 2nd Edition.
5.	**Design of the Unix Operating Systems** – Maurice Bach | Prentice-Hall of India | 8th Edition.
6.	**Understanding the Linux Kernel** – Daniel P. Bovet, Marco Cesati | O'Reilly and Associates |3rd Edition.

a. Course Name: Computer Organization and Microprocessor Architecture

b. Course Code: 03010703ES03

e. Prerequisite: Basic understanding of computer system

f. Rationale: This course provides a detailed understanding of the functional components of a computer system, their characteristics, performance, and interactions, including the systembus, various types of memory, input/output organization, and the CPU. The course also covers key architectural issues such as instruction set design, programming models, and data types. In addition, students are introduced to the increasingly important area of parallel organization. This course serves as a foundation for developing hardware-related projects and is therefore an essential course for students of the Computer Engineering program.

g. Course Learning Objectives:

CLOBJ	Objective

CLOBJ 1	To explain the introduction to microprocessors, architecture and programming model of the 8085 microprocessor, instruction set and computer languages, instruction and data formats, and data storage.
CLOBJ 2	To describe the architecture of the 8085 microprocessor, microprocessor operations, memory and I/O devices, memory interfacing techniques, and interfacing of I/O devices.
CLOBJ 3	To apply the 8085 instruction set for programming including looping techniques, counting and indexing, logical operations, rotate and compare instructions.
CLOBJ 4	To analyze computer organization concepts including register transfer language (RTL), bus design, arithmetic, logic and shift micro-operations, basic computer design, instruction codes, computer registers, timing and control, instruction cycle, memory/register/I/O reference instructions, interrupts, and design of accumulator unit.
CLOBJ 5	To understand assembler concepts, machine and assembly language programming, program loops, arithmetic and logic programming, subroutines, I/O programming, and memory organization including memory hierarchy, main memory, auxiliary memory, flash memory, associative memory, cache memory, and virtual memory.

h. Course Learning Outcomes:

CLO	Outcome
CLO 1	Explain the architecture, programming model, instruction set, and data formats of the 8085 microprocessor.
CLO 2	Analyze the operations of the 8085 microprocessor and design suitable memory and I/O interfacing circuits.
CLO 3	Develop and execute assembly language programs using the 8085 instruction set to solve basic computational problems.
CLO 4	Apply advanced programming techniques to construct efficient assembly language routines for the 8085 microprocessor.
CLO 5	Explain and evaluate the interrupt structure, types, and handling mechanisms of the 8085 microprocessor.

i. Teaching & Examination Scheme:

Teaching Scheme				**Internal Evaluation**			**ESE**		**Total**
L	**T**	**P**	**C**	**MSE**	**CE**	**P**	**Theory**	**P**	
2	-	-	2	-	-	-	60	-	100

L- Lectures; T- Tutorial; P- Practical; C- Credit; MSE- Mid-Semester Evaluation, CE- Continuous Evaluation, ESE- End Semester Examination

j. Course Content:

Sr. No.	**Content**	**Weightage**	**Teaching Hours**
1	**Introduction to Microprocessor 8085** Introduction to microprocessors, 8085 Microprocessor, Instruction set and computer languages, 8085 Programming Model, Instruction, data formats, and data storage.	15%	4
2	**Microprocessor Architecture and Interfacing** Architecture of 8085 microprocessor, Microprocessor operation, Memory devices and I/O devices, Memory interfacing techniques, Interfacing of I/O devices.	20%	6
3	**Programming Methods Using Instructions** 8085 instruction set, Looping techniques, Counting and indexing, Logical operations, Rotate and compare instructions.	20%	6
4	**Computer Organization – Register Transfer and Basic Computer Design** Register Transfer: Register Transfer Language (RTL), Bus design using multiplexers, Bus design using tri-state buffers, Memory transfers, Arithmetic micro-operations, Logic micro-operations, Shift micro-operations, Arithmetic logic shift unit. Basic Computer Design: Instruction codes, Computer registers, Computer instructions, Timing and control, Instruction cycle, Memory reference instructions, Register reference instructions, I/O reference instructions, Interrupts, Design of accumulator unit.	25%	8
5	**Computer Organization – Assembler and Memory Organization** Assembler: Machine language and assembly language, Assembler concepts, Program loops, Programming arithmetic operations, Programming logic operations, Subroutines, I/O programming. Memory Organization: Memory hierarchy, Main memory, Auxiliary memory, Flash memory, Associative memory, Cache memory, Virtual memory.	20%	6
Total		**100%**	**30**

k. Text Book and Reference Books:

1.	Microprocessor Architecture, Programming and Applications with 8085 by Gaonkar, Ramesh S. (TextBook)
2.	8085 Microprocessor and its Applications by A. NagoorKani | TMH Education Pvt. Ltd
3.	Microprocessor 8085 and its Interfacing by Sunil Mathur | PHI Learning Pvt. Ltd

4.	Microprocessor and Interfacing by Douglas V Hall | McGraw Hill Higher Education
5.	Computer System Architecture by M. Morris Mano | PHI | 3rd Edition

l. Course Learning Outcomes:

After Learning the Course the students shall be able to:
•Explain the architecture, programming model, instruction set, and data formats of the 8085 microprocessor. • Analyze the operations of the 8085 microprocessor and design suitable memory and I/O interfacing circuits. • Develop and execute assembly language programs using the 8085 instruction set to solve basic computational problems. • Apply advanced programming techniques to construct efficient assembly language routines for the 8085 microprocessor. • Explain and evaluate the interrupt structure, types, and handling mechanisms of the 8085 microprocessor.

a. Course Name: Indian Constitution

b. Course Code: 17013003MC01

e. Prerequisite: Basic understanding of what law is, its sources (statutes, case law, customs), and the legal system's structure. Familiarity with the distinction between Public Law (e.g., Constitutional and Administrative Law) and Private Law (e.g., Contract Law, Tort Law). Knowledge of the historical evolution of constitutions, particularly the Indian Constitution. Awareness of th e colonial, political, and socio-economic circumstances that led to the framing of the

Constitution.

f. Rationale: The rationale for Constitutional Law lies in its foundational role in structuring and guiding the governance of a nation while protecting the rights and freedoms of its citizens. Constitutional Law provides the legal framework for the formation, powers, functions, and structure of government institutions (Legislature, Executive, and Judiciary). It ensures a clear distribution o f power among the organs of government, preventing abuse and promoting accountability.

g. Course Learning Objectives:

CLOBJ	Objective
CLOBJ 1	To explain the history, nature, salient features, and preamble of the Indian Constitution and understand its foundational principles.
CLOBJ 2	To describe the concept of citizenship under the Indian Constitution (Articles 1–11), the definition of State (Article 12), the doctrine of void laws (Article 13), and the right to equality (Article 14).
CLOBJ 3	To analyze fundamental rights under Articles 15 to 20 of the Indian Constitution including prohibition of discrimination, equality of opportunity, abolition of untouchability, and rights relating to conviction and personal freedom.
CLOBJ 4	To apply the understanding of fundamental rights under Articles 21 to 32 of the Indian Constitution including right to life, right against exploitation, right to constitutional remedies, and fundamental duties.

h. Course Learning Outcomes:

CLO	Outcome
CLO 1	Understand the historical background, nature, salient features, and preamble of the Indian Constitution.
CLO 2	Explain the provisions relating to citizenship, definition of State, and the right to equality under the Indian Constitution.
CLO 3	Analyze and interpret fundamental rights granted under Articles 15 to 20 of the Indian Constitution.
CLO 4	Apply the provisions of Articles 21 to 32 and fundamental duties in legal reasoning and constitutional interpretation.
CLO 5	Appreciate the role of Constitutional Law in upholding democracy, governance, and protection of citizens' rights in India.

i. Teaching & Examination Scheme:

Teaching Scheme				Internal Evaluation			ESE		Total
L	**T**	**P**	**C**	**MSE**	**CE**	**P**	**Theory**	**P**	
1	-	-	0	-	50	-	-	-	50

L- Lectures; T- Tutorial; P- Practical; C- Credit; MSE- Mid-Semester Evaluation, CE- Continuous Evaluation, ESE- End Semester Examination

j. Course Content:

Sr. No.	Content	Weightage	Teaching Hours
1	**UNIT – I: History of Indian Constitution** Nature of the Indian Constitution, Salient Features of the Indian Constitution, The Preamble of the Indian Constitution.	25%	-
2	**UNIT – II: Citizenship and Right to Equality** A] Citizenship [Articles 1 to 11]. B] Definition of State [Article 12]. C] Laws inconsistent with or in derogation of Fundamental Rights [Article 13]. D] Right to Equality [Article 14].	25%	-
3	**UNIT – III: Fundamental Rights (Articles 15 to 20)** Prohibition of discrimination on grounds of religion, race, caste, sex or place of birth [Article 15]. Equality of opportunity in matters of public employment [Article 16]. Abolition of untouchability [Article 17]. Abolition of titles [Article 18]. Protection of certain rights regarding freedom of speech, etc. [Article 19]. Protection in respect of conviction for offences [Article 20].	25%	-
4	**UNIT – IV: Fundamental Rights (Articles 21 to 32) and Fundamental Duties** Protection of life and personal liberty [Article 21]. Right to education [Article 21A]. Protection against arrest and detention [Article 22]. Prohibition of traffic in human beings and forced labour [Article 23]. Prohibition of employment of children in factories [Article 24]. Freedom of conscience and free profession of religion [Articles 25–28]. Cultural and educational rights [Articles 29–30]. Right to constitutional remedies [Article 32]. Fundamental Duties.	25%	-
Total			**100%**

k. Text Book and Reference Books:

1.	History of Democratic Constitution: The Indian Expenditure by Austin G | Oxford | Year 2000 (TextBook)

l. Course Learning Outcomes:

After Learning the Course the students shall be able to:
• Understand the historical background, nature, salient features, and preamble of the Indian Constitution. • Explain the provisions relating to citizenship, definition of State, and the right to equality under the Indian Constitution. • Analyze and interpret fundamental rights granted under Articles 15 to 20 of the Indian Constitution. • Apply the provisions of Articles 21 to 32 and fundamental duties in legal reasoning and constitutional interpretation. • Appreciate the role of Constitutional Law in upholding democracy, governance, and protection of citizens' rights in India.

Course Name: Functional Communication

Skills Course Code: 03010003HM01

Prerequisite: Knowledge of Advanced Communication and Interpersonal Skills

Rationale: This course develops workplace-oriented communication skills by bridging academic language competence with professional communication requirements.

Course Learning Objectives

CLOBJ	Objective
CLOBJ 1	Develop accuracy in written English by identifying and correcting grammatical, usage, and style errors in professional contexts.
CLOBJ 2	Build logical thinking and coherent communication through sentence reordering, para jumbles, and analysis of statements and assumptions.
CLOBJ 3	Develop proficiency in professional written communication including emails, reports, resumes, and cover letters following workplace s tandards.
CLOBJ 4	Enhance verbal communication competence through structured speaking activities such as JAM and telephone/video call etiquette practice.
CLOBJ 5	Equip students with digital professional presence skills including LinkedIn profile building and digital networking ethics.

Course Learning Outcomes

After Learning the Course the students shall be able to:

CLO	Outcome
CLO 1	Identify grammatical, usage, and style errors; logically reorder sentences; and differentiate facts from assumptions in workplace problem scenarios.
CLO 2	Understand the usage of grammatical rules, cohesion markers, professional writing formats, and communication etiquette to produce accurate workplace communication.
CLO 3	Apply principles of professional communication to ensure clarity, coherence, time management, and etiquette in both written and spoken workplace contexts.
CLO 4	Analyse effective verbal, digital, and virtual communication skills through JAM participation, LinkedIn profile optimization, and telephone/video call interactions.
CLO 5	Create ATS-friendly resumes, customized cover letters, professional emails, structured reports, and optimized LinkedIn profiles.

Teaching & Examination Scheme:

Teaching Scheme			Seminar Hrs/Week	Credit	Internal Marks		External Marks	
Lecture Hrs/Week	**Tutorial Hrs/Week**	**Lab Hrs/Week**			**T**	**P**	**T**	**P**
1	—	2	—	**2**	40	20	60	30

SEE - Semester End Examination, T - Theory, P - Practical | Total Marks: 150

Course Content:

Sr. No.	Content	Weightage (%)	Teaching Hours
1	**Sentence Correction:** Error identification (grammar, usage, style), Common workplace errors, Contextual grammar usage	10	2
2	**Para Jumbles & Sentence Reordering:** Logical sequencing, Cohesion markers, Theme identification	8	1
3	**Statement and Assumptions:** Fact vs assumption, Logical reasoning basics, Workplace problem scenarios	10	1
4	**Reading Comprehension (Level of Difficulty – Advanced):** Inferential questions, Author's tone & intent, Vocabulary in context	12	2
5	**Resume and Cover Letter Writing:** Resume formats, Achievement-based bullet points, Customizing cover letters	14	2
6	**Building a Professional LinkedIn Profile:** Professional headline, Summary writing, Digital	8	1

	networking ethics		
7	**Just a Minute (JAM):** Idea organization, Fluency techniques, Time management in speech	8	1
8	**Telephone and Video Call Etiquette:** Opening & closing calls, Voice modulation, Virtual meeting etiquette	8	1
9	**Email Writing:** Format, Professional tone, Subject lines, Email etiquette	10	2
10	**Report Writing:** Types of reports, Structure & formatting, Use of visuals & data	12	2
	Total	**100**	**15**

List of Practicals:

Sr. No.	Practical
1.	Sentence Correction: Grammar & usage error identification · Context-based sentence correction worksheets · Common workplace error correction · Peer editing activities
2.	Para Jumbles and Reordering of Sentences: Identifying topic sentences · Logical sequencing exercises · Use of cohesion markers · Timed para-jumble practice
3.	Reading Comprehension (Advanced): Inferential and analytical questions · Identifying author's tone & intent · Vocabulary-in-context activities · Group discussion on passages
4.	Resume and Cover Letter Writing: Resume formats (chronological, functional) · Achievement-based bullet writing · ATS-friendly resume drafting · Customized cover letter writing
5.	Statement and Assumptions: Identification of implicit assumptions through guided exercises · Application of assumption-testing techniques (possibility test) · Analysis of case-based and real-life reasoning scenarios · Timed practice drills with discussion of common errors
6.	Building a Professional LinkedIn Profile: Writing professional headlines · Summary and About section drafting · Profile optimization task · Digital networking ethics
7.	Just a Minute (JAM): Topic-based JAM speaking practice · Fluency and coherence drills · Time-management techniques · Individual feedback sessions
8.	Telephone and Video Call Etiquette: Professional call role-plays · Voice modulation exercises · Mock video meeting practice · Virtual etiquette evaluation
9.	Report Writing: Types of reports (incident, progress, proposal) · Structure and formatting practice · Use of visuals and data · Writing and reviewing short reports
10.	Email Writing: Professional email drafting · Subject-line writing activities · Tone and etiquette correction · Peer review and rewriting

Text Book and Reference Books:

Sr. No	Reference
1.	Business Communication Today — Bovee, Courtland L. & Thill, John V. | Pearson Education, 2019
2.	Essentials of Business Communication — Guffey, Mary Ellen & Loewy, Dana | Cengage Learning, 2018
3.	Advanced Grammar in Use — Hewings, Martin | Cambridge University Press, 2013
4.	English Vocabulary in Use: Advanced — McCarthy, Michael & O'Dell, Felicity | Cambridge University Press, 2017
5.	Personality Development and Soft Skills — Mitra, Barun K | Oxford University Press, 2011
6.	Technical Communication: Principles and Practice — Raman, Meenakshi & Sharma, Sangeeta | Oxford University Press, 2018

a. Course Name: Design and Analysis of Algorithm

b. Course Code: 03010505PC01

c. Course: B. Tech

d. Semester: 4TH SEM

e. Prerequisite: Data Structures | 03010504PC03 - Fundamental of Programming

f. Rationale: Study of data structures and algorithms, including analysis of time and space complexity, and development of efficient algorithms using mathematical techniques and programming. It also helps students to design new algorithms through mathematical analysis and programming.

---

g. Course Learning Objectives:

CLOBJ 1	To explain the fundamental concepts of algorithms, their importance, and analyze time and space complexity using asymptotic notations and recurrence relations.
CLOBJ 2	To apply divide and conquer techniques including Binary Search, Quick Sort, Merge Sort, and Strassen's Matrix Multiplication.
CLOBJ 3	To design solutions using greedy algorithms for problems like Huffman Coding, Minimum Spanning Tree, Knapsack, and shortest path algorithms.
CLOBJ 4	To apply dynamic programming techniques to solve problems including 0/1 Knapsack, Traveling Salesman Problem, and Floyd-Warshall Algorithm.
CLOBJ 5	To analyze backtracking and branch & bound strategies and understand NP-Hard and NP-Complete problem classifications.

h. Course Learning Outcomes:

CLO 1	Understand the fundamentals of algorithms and their analysis.
CLO 2	Design efficient algorithms for computational problems using various algorithm design techniques.
CLO 3	Explain the major graph algorithms and their analyses. Employ graphs to model engineering problems, when appropriate.
CLO 4	Analyze string matching algorithms.
CLO 5	Explain the complexity classes P, NP, and NP-Complete, and demonstrate the NP-Completeness of a specific problem.

i. Teaching & Examination Scheme:

Teaching Scheme				**Evaluation Scheme**					
L	**T**	**P**	**C**	**Internal Evaluation**			**ESE**		Total
				MSE	**CE**	**P**		**P**	
3	0	-	3	20	20	-	3	-	100

L- Lectures; T- Tutorial; P- Practical; C- Credit; MSE- Mid-Semester Evaluation, CE- Continuous Evaluation, ESE- End Semester Examination

---

j. Course Content:

Sr. No.	Content	Weightage	Teaching Hours
1	**Unit-1: Basics of Algorithms:** Introduction to algorithms and their importance, Analysis of algorithms with time and space complexity, Recurrence relations, Master Theorem, Modified Master Theorem, Recursive tree method, Back substitution, and asymptotic notations. Divide and conquer technique: Binary Search, Quick Sort, Merge Sort, and Strassen's Matrix Multiplication, Sliding window, 2 pointer algorithm, Bit manipulation - masking.	20%	9
2	**Unit-2: Greedy Algorithms:** Introduction to greedy strategy and how it works, Problems based on greedy approach such as Optimal Merge Pattern, Huffman Coding, Minimum Spanning Tree, Knapsack Problem, Job Sequencing with Deadlines, and Single Source Shortest Path algorithms - Dijkstra's Algorithm, Bellman-Ford.	25%	12

3	**Unit-3: Dynamic Programming:** Introduction to dynamic programming and its advantages, Problem-solving using dynamic programming techniques including 0/1 Knapsack, Traveling Salesman Problem, Multistage Graphs, Reliability Design, and Floyd-Warshall Algorithm.	25%	11
4	**Unit-4: Backtracking:** Concept of backtracking and its applications, Problems such as N-Queen Problem, Hamiltonian Cycle, subset and subset-2, Graph Coloring Problem.	20%	9
5	**Unit-5: Branch and Bound & NP Problems:** Introduction to branch and bound technique with examples like Traveling Salesman Problem, Concept of lower bound and its use in solving problems, Introduction to NP-Hard and NP-Complete problems, non-deterministic algorithms, and classification of NP problems.	10%	4
Total		**100%**	**45**

k. Text Book and Reference Book:

1.	Introduction to Algorithms, 4th Edition, Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest and Clifford Stein, MIT Press/McGraw-Hill. (TextBook)
2.	Fundamentals of Algorithms – E. Horowitz et al. (TextBook)
3.	Algorithm Design, 1st Edition, Jon Kleinberg and Eva Tardos, Pearson
4.	Algorithm Design: Foundations, Analysis, and Internet Examples, 2nd Edition, Michael T. Goodrich and Roberto Tamassia, Wiley
5.	Algorithms – A Creative Approach, 3rd Edition, Udi Manber, Addison-Wesley
6.	Harsh Bhasin "Algorithms Design and Analysis" Oxford
7.	I. Chandra Mohan "Design and Analysis of Algorithms" PHI

a. **Course Name:** Azure Fundamentals Laboratory

b. **Course Code:** 03010504PE06

c. **Course:** B. Tech

d. **Semester:** 4

e. **Prerequisite:** Basic computer fundamentals and introductory programming knowledge.

f. **Rationale:** This course introduces students to cloud computing fundamentals using Microsoft Azure, focusing on conceptual clarity, employability skills, digital infrastructure awareness, and industry relevance.

g. **Course Learning Outcomes:**

CLO 1	Explore and navigate the Microsoft Azure portal to create and manage cloud resources.
CLO 2	Deploy and configure virtual machines and cloud-based storage solutions in Azure.
CLO 3	Implement network security controls, resource locks, and resource group management.
CLO 4	Configure Role-Based Access Control (RBAC) and monitor Azure resources using Azure Monitor.
CLO 5	Manage Azure costs effectively and deploy resources using Azure Resource Manager (ARM) templates.

h. **Teaching & Examination Scheme:**

Teaching Scheme				Evaluation Scheme					Total
L	T	P	C	Internal Evaluation			ESE		
				MSE	CE	P	Theory	P	
0	0	2	1	-	-	20	-	30	50

L- Lectures; T- Tutorial; P- Practical; C- Credit; MSE- Mid-Semester Evaluation, CE- Continuous Evaluation, ESE- End Semester Examination

i. **Experiment List:**

Sr. No.	Experiment List
1	A startup is beginning its cloud journey and needs a centralized platform to manage its cloud services. As a cloud administrator, explore the Azure portal and create the first cloud resource.
2	A company wants to move its existing application from an on-premises server to the cloud without changing the application architecture. Deploy a virtual server in Azure.
3	An organization wants its cloud virtual machine to be accessible only to authorized users and restrict unwanted internet traffic. Configure the secure network access.
4	A media company needs scalable and reliable storage for images, videos, and documents that can be accessed from anywhere. Implement the cloud-based storage.
5	A production database hosted in Azure was accidentally deleted earlier, causing downtime. Management now wants to ensure critical resources cannot be deleted or modified unintentionally. Apply Azure Resource Lock.
6	An organization runs multiple projects such as development, testing, and production in Azure. Resources need to be logically grouped for easier management and monitoring. Manage the resource groups.
7	A company has developers, testers, and administrators, each requiring different levels of access to Azure resources. You are responsible for assigning appropriate permissions. Implement Azure Role-Based Access Control (RBAC).
8	An e-commerce application hosted on Azure must be available 24×7. The operations team wants to monitor performance, detect issues, and check service health. Monitor the resources using Azure Monitor.
9	A company notices increased cloud bills and wants to analyze usage and control expenses. Review and manage Azure costs effectively.
10	An organization wants to deploy identical cloud environments repeatedly for development and testing. Manual creation is time-consuming, so automation is required. Deploy Resources Using Azure Resource Manager Templates.

a. **Course Name:** Software Engineering

b. **Course Code:** 303105253

c. **Course:** B. Tech

d. **Semester:** 4 (Four Semester)

e. **Prerequisite:** Basic knowledge of software applications.

f. **Rationale:** This course provides a broad introduction to software engineering. The various process models required to develop software are described. Moreover, the functional and non-functional requirements are also described.

g. **Course Learning Objectives:**

CLOBJ 1	To explain the fundamental concepts of software engineering, software characteristics, process models including Waterfall, Incremental, Evolutionary, and Agile development methodologies.
CLOBJ 2	To describe software project management principles including effort estimation, scheduling, risk management, and team management.
CLOBJ 3	To apply requirements engineering techniques including problem recognition, requirement specification, use cases, and validation.
CLOBJ 4	To design software systems using structured and object-oriented design approaches including architecture, data design, component-level design, and OOD.
CLOBJ 5	To understand coding standards, unit testing, metrics, and software quality assurance including testing strategies, standards, and reliability.
CLOBJ 6	To analyze advanced software engineering practices including CASE tools, SCRUM, dependable systems, security engineering, software reuse, and distributed software engineering.

h. **Course Learning Outcomes:**

CLO 1	Prepare and perform Software Requirement Specification and Software Project Management Plan.
CLO 2	Ensure the quality of software product using different quality standards and software review techniques.
CLO 3	Apply the concept of Functional Oriented and Object-Oriented Approach for Software Design.
CLO 4	Describe modern Agile Development and Service Oriented Architecture concepts of industry.
CLO 5	Analyze, design, verify, validate, implement and maintain software systems.
CLO 6	Execute a Project Management Plan, tabulate Testing Plans and reproduce effective procedures.

i. **Teaching & Examination Scheme:**

Teaching Scheme				Evaluation Scheme					Total
L	T	P	C	Internal Evaluation			ESE		
				MSE	CE	P	Theory	P	
3	0	-	3	20	20	-	60	-	100

L- Lectures; T- Tutorial; P- Practical; C- Credit; MSE- Mid-Semester Evaluation, CE- Continuous Evaluation, ESE- End Semester Examination

j. **Course Content:**

Sr. No.	Content	Weightage	Teaching Hours
1	**Introduction** Study of Different Models, Software Characteristics, Components, Applications, Layered Technologies, Processes, Methods and Tools, Generic View of Software Engineering, Process Models: Waterfall model, Incremental, Evolutionary process models – Prototype, Spiral and Concurrent Development Model. Agile Development: Agility and Agile Process model, Extreme Programming, Other process models of Agile Development and Tools.	10%	6

2	**Software Project Management** Management Spectrum, People–Product–Process–Project, W5HH Principle, Importance of Team Management. Planning a Software Project: Scope and Feasibility, Effort Estimation, Schedule and Staffing, Quality Planning, Risk Management – identification, assessment, control, project monitoring plan, Detailed Scheduling.	10%	5
3	**Requirements Engineering** Problem Recognition, Requirement Engineering Tasks, Processes, Requirements Specification, Use Cases and Functional Specification, Requirements Validation, Requirements Analysis.	10%	5
4	**Structured System Design** Design Concepts, Design Model, Software Architecture, Data Design, Architectural Styles and Patterns, Architectural Design, Alternative Architectural Designs, Modeling Component Level Design and its Modeling, Procedural Design, Object Oriented Design. Data Oriented Analysis & Design: Difference between Data and Information, E-R Diagram, Dataflow Model, Control Flow Model, Control and Process Specification, Data Dictionary.	15%	5
5	**Coding and Unit Testing** Programming Principles and Guidelines, Programming Practices, Coding Standards, Incremental Development of Code, Management of Code Evaluation, Unit Testing – Procedural Units, Classes, Code Inspection. Metrics: Size Measure, Complexity Metrics, Cyclomatic Complexity, Halstead Measure, Knot Count, Comparison of Different Metrics.	10%	4
6	**Software Testing and Quality Assurance** Concepts, Psychology of Testing, Levels of Testing, Testing Process – Test Plan, Test Case Design, Execution, Black-Box Testing, Boundary Value Analysis, Pair-wise Testing, State-based Testing, White-Box Testing Criteria and Test Case Generation and Tool Support. Quality Assurance: Quality Control, Assurance, Cost, Reviews, Software Quality Assurance, Approaches to SQA, Reliability, Quality Standards – ISO 9000 and 9001.	15%	7
7	**CASE Tools and Advanced Practices of System Dependability and Security** Computer Aided Software Engineering Tools, SCRUM Developments, Dependable System, Reliability Engineering, Safety Engineering, Security Engineering, Resilience Engineering.	15%	5
8	**Advanced Software Engineering** Software Reuse, Component Based Software Engineering, Distributed Software Engineering, Service-Oriented Software Engineering, Real-Time Software Engineering, Systems Engineering, Systems of System.	15%	5
	Total	100%	42

k. Text Book and Reference Book:

1.	Software Engineering by R. Pressman – 6th Edition (TextBook)
2.	Software Engineering by Ian Sommerville
3.	Fundamentals of Software Engineering by Rajib Mall | PHI
4.	Software Engineering by Pankaj Jalote | Wiley India

a. **Course Name:** Computer Networks

b. **Course Code:** 03010504PC01

c. **Course:** B. Tech

d. **Semester:** 4

e. **Prerequisite:** Knowledge of Computer and Information System.

f. **Rationale:** This course is designed to provide basic knowledge about data and signals. It also provides basic concepts of computer networks and a firm foundation for understanding how data communication occurs in the Transmission Medium. It will help to develop logical abilities and practically set up the network.

g. **Course Learning Objectives:**

CLOBJ 1	To explain data communication fundamentals including representation of data, network topologies, protocols, standards, OSI model, transmission media, LAN technologies, multiplexing, and spread spectrum concepts.
CLOBJ 2	To describe data link layer functions including error detection and correction, flow control, error control protocols, and multiple access protocols.
CLOBJ 3	To apply network layer concepts including switching, logical addressing (IPv4, IPv6), address mapping protocols, delivery, forwarding, and unicast routing protocols.
CLOBJ 4	To analyze transport layer services including process-to-process communication, UDP, TCP, SCTP, congestion control, quality of service, and traffic shaping algorithms.
CLOBJ 5	To understand application layer protocols and services including DNS, TELNET, EMAIL, FTP, HTTP, SNMP, Bluetooth, firewalls, and basic cryptography concepts.

h. **Course Learning Outcomes:**

CLO 1	Draw the functional block diagram of wide-area networks (WANs), local area networks (LANs), and Wireless LANs (WLANs) and describe the function of each block.
CLO 2	Explain the functions of the different layers of the OSI protocol model.
CLO 3	Describe and design for a given requirement (small scale) of WANs, LANs, and WLANs based on market available components.
CLO 4	Apply TCP/IP protocol concepts to network programming problem scenarios.
CLO 5	Configure DNS, DDNS, TELNET, EMAIL, FTP, WWW, HTTP, SNMP, Bluetooth, and Firewalls using open-source software and tools.

i. **Teaching & Examination Scheme:**

Teaching Scheme				**Evaluation Scheme**					**Total**
L	**T**	**P**	**C**	**Internal Evaluation**			**ESE**		
				MSE	**CE**	**P**	**Theory**	**P**	
3	0	-	3	40	20	-	60	-	100

L- Lectures; T- Tutorial; P- Practical; C- Credit; MSE- Mid-Semester Evaluation, CE- Continuous Evaluation, ESE- End Semester Examination

j. **Course Content:**

Sr. No.	**Content**	**Weightage**	**Teaching Hours**
1	**Data Communication Components** Representation of data and its flow, Networks, Various Connection Topology, Protocols and Standards, OSI model, Transmission Media, LAN: Wired LAN, Wireless LANs, Connecting LAN and Virtual LAN. Techniques for Bandwidth Utilization: Multiplexing – Frequency Division, Time Division and Wave Division. Concepts on Spread Spectrum.	25%	11

2	**Data Link Layer and Medium Access Sub Layer** Error Detection and Error Correction – Fundamentals, Block coding, Hamming Distance, CRC. Flow Control and Error Control Protocols – Stop and Wait, Go Back-N ARQ, Selective Repeat ARQ, Sliding Window, Piggybacking. Random Access, Multiple Access Protocols – Pure ALOHA, Slotted ALOHA, CSMA/CD, CSMA/CA.	25%	11
3	**Network Layer** Switching, Logical Addressing – IPv4, IPv6. Address Mapping – ARP, RARP, BOOTP and DHCP. Delivery, Forwarding and Unicast Routing Protocols. IP Routing – Intra Domain Routing Protocols, Inter Domain Routing Protocols (BGP).	20%	10
4	**Transport Layer** Process to Process Communication, User Datagram Protocol (UDP), Transmission Control Protocol (TCP), SCTP Congestion Control. Quality of Service, QoS Improving Techniques: Leaky Bucket and Token Bucket Algorithm.	15%	7
5	**Application Layer** Domain Name Space (DNS), DDNS, TELNET, EMAIL, File Transfer Protocol (FTP), WWW, HTTP, SNMP, Bluetooth, Firewalls, Basic Concepts of Cryptography.	15%	6
	Total	100%	45

k. Text Book and Reference Book:

1.	Computer Networks by Andrew S. Tanenbaum and David J. Wetherall | Pearson Edition (TextBook)
2.	Internetworking with TCP/IP: Principles, Protocols and Architecture by Douglas E. Comer
3.	TCP/IP Illustrated by Richard Stevens
4.	Data Communication and Networking by Behrouz A. Forouzan
5.	Data and Computer Communications by William Stallings | Prentice Hall

a. Course Name: Database Management System Laboratory

b. Course Code: 03010504PC06

c. Course: B. Tech

d. Semester: 4

e. Prerequisite: Basic Computer Knowledge

f. Rationale: The course will enable students to understand the different issues involved in the design and implementation of a database system as well as execute various database queries using SQL.

g. Course Learning Outcomes:

CLO	After Learning the Course the students shall be able to:
CLO 1	Understand basic concepts of Database and Relational Models and their importance.
CLO 2	Build properly structured databases for a given problem or application using SQL DDL and DML commands.
CLO 3	Apply aggregate functions, sorting, grouping, single-row functions, and multi-table join queries on relational databases.
CLO 4	Understand how various transactions are managed in real-time scenarios using TCL and DCL commands.
CLO 5	Implement SQL concepts and PL/SQL blocks to build dynamic database applications.

h. Teaching & Examination Scheme:

Teaching Scheme				Internal Evaluation			ESE		Total
L	**T**	**P**	**C**	**MSE**	**CE**	**P**	**Theory**	**P**	
0	0	2	1	-	-	20	-	30	50

L- Lectures; T- Tutorial; P- Practical; C- Credit; MSE- Mid-Semester Evaluation, CE- Continuous Evaluation, ESE- End Semester Examination

i. List of Practical:

Sr. No.	List of Practical
1.	Study of DDL (CREATE) and DML (INSERT) commands. Create tables DEPOSIT, BRANCH, CUSTOMERS, BORROW as per given definition, insert data, and perform basic SELECT queries including filtering by amount, name, and date conditions.
2.	Create JOB and EMPLOYEE tables, insert given data, and perform queries using BETWEEN predicate, IN predicate, LIKE predicate, and various relational and logical operators.
3.	Perform various Data Manipulation commands, Aggregate Functions, and Sorting. Use GROUP BY, ORDER BY, CREATE TABLE AS, INSERT INTO SELECT, DELETE, RENAME, DROP, and UPDATE operations.
4.	Study of Single-Row Functions: Apply character functions, arithmetic functions, date functions, and conversion functions on employee and deposit tables. Perform LIKE-based queries across multiple tables.
5.	Displaying data from Multiple Tables using Joins. Implement INNER JOIN, OUTER JOIN, and SELF JOIN with real-world query scenarios on DEPOSIT, BORROW, CUSTOMERS, BRANCH, and EMPLOYEE tables.
6.	Aggregating Data using Group Functions. Use GROUP BY, HAVING clause, and aggregate functions (SUM, AVG, MAX, MIN, COUNT) on multiple tables with filtering conditions.
7.	Solve queries using the concept of Sub-Query. Implement single-row subqueries, multi-row subqueries, and correlated subqueries on DEPOSIT, BORROW, EMPLOYEE, and related tables.
8.	Manipulating Data: Apply percentage-based interest updates, conditional DELETE operations, and simulated bank transfer transactions using UPDATE and DELETE commands.
9.	TCL and DCL Commands in SQL: Use COMMIT, ROLLBACK, SAVEPOINT for transaction control and GRANT, REVOKE for data control.
10.	PL/SQL Blocks: Write PL/SQL programs to add 2 numbers, find area of rectangle/triangle/square, find maximum of 3 numbers, print sum of N numbers using FOR loop, and generate Fibonacci series of N numbers.

a. Course Name: Programming in Python with Full Stack Development

b. Course Code: 303105257

c. Course: B. Tech

d. Semester: 4

e. Prerequisite: Basic knowledge of Programming and Web Applications

f. Rationale: This course provides a broad introduction to Python programming and development of web applications. It covers using Python as a scripting language for automating tasks and data processing, and building and deploying web applications using popular Python frameworks such as Django and Flask.

g. Course Learning Objectives:

CLOBJ	Objective
CLOBJ 1	To explain the fundamental concepts of Python programming including variables, data types, conditional statements, loops, and built-in data structures such as lists, sets, tuples, and dictionaries.
CLOBJ 2	To describe the use of functions, object-oriented programming concepts (class, object, abstraction, encapsulation, polymorphism, inheritance), exception handling, and file handling in Python.
CLOBJ 3	To apply Python modules, packages, and popular libraries (NumPy, Pandas, Matplotlib, Pygame), PyCharm IDE with Git integration, PyTests, and Python database connectivity with SQLite and MongoDB.
CLOBJ 4	To build and deploy web applications using the Flask framework including routing, templates, static/media files, form handling, database connectivity (SQLite3, MySQL), authentication, and email integration.
CLOBJ 5	To develop web applications using the Django framework including project structure, apps, admin console, views, URL mapping, template system, models, form processing, and deployment.
CLOBJ 6	To design and implement RESTful APIs using HTTP methods (GET, POST, PUT, DELETE), apply best practices for API versioning and error handling, and consume APIs using tools such as cURL, Postman, and the Python requests library.

h. Course Learning Outcomes:

CLO	Outcome
CLO 1	Understand the fundamental concepts of Python programming and web development.
CLO 2	Create and manipulate data using a variety of databases, including SQL (SQLite, MySQL) and NoSQL (MongoDB).
CLO 3	Build and deploy web applications using a popular Python web framework such as Django or Flask.
CLO 4	Design and implement APIs (application programming interfaces) that enable different applications to communicate with each other.
CLO 5	Test and debug web applications and deploy them to production environments.

i. Teaching & Examination Scheme:

Teaching Scheme				Internal Evaluation			ESE		Total
L	**T**	**P**	**C**	**MSE**	**CE**	**P**	**Theory**	**P**	
3	0	-	3	40	20	-	60	-	100

L- Lectures; T- Tutorial; P- Practical; C- Credit; MSE- Mid-Semester Evaluation, CE- Continuous Evaluation, ESE- End Semester Examination

j. Course Content:

Sr No.	Content	Weightage	Teaching Hours
1	**Introduction to Python Programming** Introduction to Python and basic programming concepts, variables, data types, conditional statements and loops. Lists, Sets, Tuples, Dictionaries: Working with strings, lists, sets, tuples and dictionaries, including common operations and built-in functions.	15%	6
2	**Functions, OOP Concepts and Exception Handling** Functions: Defining and using functions, including the use of arguments and return values. OOP Concepts: Object, class, abstraction, encapsulation, polymorphism, inheritance. Exceptions and File Handling: Handling exceptions and working with files.	10%	5
3	**Modules, Packages and Python Libraries** Working with modules and packages in Python. Introduction to popular Python libraries: data analysis (NumPy, Pandas, Matplotlib), game development (Pygame). PyCharm IDE:	10%	5

	Git integration with PyCharm IDE, PyTests. Python connectivity with databases: SQLite, MongoDB CRUD operations.			
4	**Flask Framework** Introduction to Flask and web development with Python, installation in Virtual Environment. Creation, Routing, App Settings, URL Building, HTTP methods, Templates, working with Static and Media Files. Sending form data to template. Flask App with database connectivity (SQLite3, MySQL). Handling exceptions and errors, flash messages, working with mails. Authenticating and authorizing users with Flask-Login. Deploying a Flask application to a web server.	25%	10	
5	**Django Framework** Introduction to Django framework, Django project installation in Virtual Environment. Phases in Django Project Creation, creation of apps and their structure, working with Admin Console. Creating Views, URL Mapping, Template System, working with Models. Form Processing, static and media files, Django App Deployment.	25%	10	
6	**RESTful APIs** Introduction to RESTful APIs and the REST architectural style. Understanding the HTTP protocol and its role in RESTful APIs. Designing and implementing RESTful APIs using common HTTP methods such as GET, POST, PUT, and DELETE. Using URLs and resource representations to identify and transfer data. Implementing best practices such as HTTP status codes, versioning, and error handling. Consuming RESTful APIs using cURL, Postman, and the requests library in Python. Building scalable and secure RESTful APIs using Flask or FastAPI.	15%	6	
Total			**100%**	**42**

k. Text Book and Reference Books:

1.	Fluent Python, 2nd Edition by Luciano Ramalho (TextBook)
2.	Learn Python 3 the Hard Way by Zed Shaw
3.	“Django for Beginners: Build websites with Python and Django” by William S. Vincent
4.	“Learning Django Web Development” by Samuli Natri
5.	“Flask Web Development with Python” by Miguel Grinberg
6.	“Mastering Flask” by Jack Stouffer
7.	“Building RESTful Python Web Services” by Gastón C. Hillar
8.	“Building Web APIs with FastAPI” by Samuel Colvin

l. Course Learning Outcomes:

After Learning the Course, the students shall be able to:
• Understand the fundamental concepts of web development. • Create and manipulate data using a variety of databases, including SQL and NoSQL. • Build and deploy web applications using a popular Python web framework such as Django or Flask. • Design and implement APIs that enable different applications to communicate with each other. • Test and debug web applications and deploy them to production environments.

a. **Course Name: Probability, Statistics and Numerical Method**

b. **Course Code:** 03019104BS01

c. **Semester:** 4

d. **Course:** BTech

e. **Prerequisite:** Basic knowledge of algebra, calculus, and elementary differential equations.

f. **Rationale:** This course equips students with essential mathematical tools for engineering problem-solving, covering probability theory, statistical analysis, and numerical methods for solving equations, interpolation, integration, and ordinary differential equations.

g. **Course Learning Objectives**

CLOBJ	Objective
CLOBJ 1	Understand the fundamental concepts of probability and statistical methods for analyzing uncertain events and data.
CLOBJ 2	Apply measures of central tendency, dispersion, correlation, and regression to interpret and analyze real-life datasets.
CLOBJ 3	Use probability distributions and statistical techniques for prediction, estimation, and decision-making in engineering and scientific applications.
CLOBJ 4	Develop numerical solutions for algebraic and transcendental equations using appropriate numerical methods.
CLOBJ 5	Apply numerical techniques for interpolation, differentiation, integration, and solving ordinary differential equations.

h. **Course Learning Outcomes:**

CO	Outcome
CO 1	Explain fundamental concepts of probability, random variables, probability distributions, and basic statistical measures.
CO 2	Perform statistical analysis and hypothesis testing for data-driven decision-making.
CO 3	Solve interpolation, numerical integration techniques, systems of linear equations, and ordinary differential equations using appropriate numerical methods.
CO 4	Analyse the accuracy, convergence, and limitations of numerical techniques for given problems.

i. **Teaching & Examination Scheme:**

Teaching Scheme			Credit	Internal Assessment			External Assessment		Total
Lecture	**Tutorial**	**Lab**		**T**	**CE**	**P**	**T**	**P**	
3	—	—	**3**	20	20	0	60	0	**100**

L - Lectures; T - Tutorial; P - Practical; C.E. - Continuous Evaluation

Continuous Evaluation: Consists of Assignments / Seminars / Presentations / Quizzes / Surprise Tests (Summative/MCQ) etc.

j. **Course Content:**

Sr. No.	Topic	Weightage	Teaching Hrs.
1	**Basic Probability and Probability Distribution:** Introduction to probability, axioms, and events, Conditional probability and Bayes' theorem, Random variables (discrete and continuous), Probability mass and density functions, Mathematical expectation and variance, Standard distributions: Binomial, Poisson, and Normal.	25%	11
2	**Basic Statistics and Testing of Hypothesis:** Measure of central tendency: Ungrouped and Grouped Data, Standard deviation, variance and standard error, Population and Sample, Large sample tests (one mean test, two mean test), Small sample test (one mean test, two mean test, paired t-test), Chi-square test for goodness of fit and independence of attributes.	30%	13
3	**Numerical Methods I (Interpolation and Integration):** Finite difference, Relationship between operators, Error Analysis, Interpolation using Newton's forward and backward formulae, Lagrange's formulae for unequal interval. Numerical Integration: Trapezoidal rule, Simpson's 1/3rd rule, and Simpson's 3/8th rule.	20%	10
	Numerical Methods II (Linear Systems and ODEs): Solution of system of linear		

4	equations: Gauss Jacobi and Gauss Seidel methods. Numerical solution of Ordinary Differential Equations: Euler and Modified Euler's methods, Runge-Kutta method of second order and fourth order.	25%	11
	Total	**100%**	**45**

k. Text Books and Reference Books:

Sr No.	**Book**
T1	**Text Book:** Erwin Kreyszig, "Advanced Engineering Mathematics" (8th Edition), Wiley India Edition.
T2	**Text Book:** S. S. Shastry, "Introductory Methods of Numerical Analysis", Prentice Hall India Learning Private Limited.
R1	**Reference:** Dr. B. S. Grewal, "Higher Engineering Mathematics", Khanna Publishers, New Delhi.
R2	**Reference:** Murray Spiegel, "Advanced Mathematics for Engineering & Science: Schaum's Outline Series", Tata McGraw Hill Publication.
R3	**Reference:** Merel C. Potter, J. L. Goldberg, "Advanced Engineering Mathematics" (3rd Edition), Oxford India Publication.