



Semester 4th Syllabus

1. Probability & Statistics

Topic	Key Focus Areas	Preparation Method	Expected Output
Measures of Central Tendency	Mean (Arithmetic, Weighted), Median, Mode – Definitions, Properties, Merits, Demerits	- Memorize all formulas.- Solve numerical problems for each type of data (raw, discrete, continuous).- Analyze solved examples to grasp patterns.	Correct application of formulas and concepts.
Measures of Dispersion	Range, Quartiles, Mean Deviation, S.D., Variance, Coefficient of Variation	- Practice problems involving data sets.- Understand coefficient calculations.- Relate dispersion measures to real-world applications.	Ability to calculate spread/variability in data.
Probability Theory	Events (Union, Intersection, Independent), Probability Theorems (Addition, Multiplication), Conditional & Inverse Probability	- Use Venn diagrams for visual understanding.- Solve theoretical and numerical questions.- Memorize theorem derivations.	Clear understanding of event relationships.
Random Variables	Discrete & Continuous Variables, PDF, CDF, Joint PDF/CDF, Marginal & Conditional Distributions	- Differentiate between discrete and continuous cases.- Practice graphical interpretations (PDF/CDF).- Solve problems involving distributions.	Accurate computation and application of concepts.
Distributions	Binomial, Poisson, Normal, Uniform –	- Memorize key properties and	Application of distributions in

	PMF/PDF, Properties, Applications	formulas.- Solve numerical problems involving distributions.- Visualize distributions with graphs.	real-world scenarios.
Correlation & Regression	Types of Correlation, Karl Pearson's Coefficient, Regression Lines & Coefficients	- Solve numerical problems to calculate coefficients.- Practice plotting regression lines and interpreting results.	Proficient in correlation and regression analysis.

2. Operating System

Topic	Key Focus Areas	Preparation Method	Expected Output
Linux Kernel	FOSS, OS Services, System Calls, Kernel Data Structures, Microkernel	- Understand kernel architecture using diagrams.- Practice theoretical questions related to system calls and OS services.- Compare kernel types.	Clear understanding of kernel design principles.
File Systems	Inodes, Super Blocks, File Sharing, Secondary Storage Management	- Focus on understanding inodes and their role in file systems.- Study file and directory organization.- Practice writing short notes on concepts like Super Blocks.	Grasp of internal file organization concepts.
Process Management	Process vs Thread, State Transitions, Scheduling Algorithms, Synchronization Issues	- Practice Gantt charts for scheduling problems.- Solve problems on synchronization and deadlock.- Memorize process state transitions.	Proficiency in solving scheduling problems.
Memory Management	Logical vs Physical Addressing, Paging, Segmentation, Virtual Memory	- Practice diagrammatic representations.- Memorize differences between addressing schemes.- Solve	Clear conceptual understanding of memory handling.

		problems on paging and segmentation.	
Inter-Process Communication	Signals, Pipes, Semaphores, Monitors, Deadlocks, Dining Philosophers Problem	- Solve standard problems (e.g., dining philosophers, producer-consumer).- Focus on deadlock detection and prevention techniques.- Understand semaphore implementation.	Clear understanding of concurrency control.

3. Functional Programming

Topic	Key Focus Areas	Preparation Method	Expected Output
Python Programming Basics	Types, Operators, Loops, Lambda Functions	- Write programs for conditional/loop-based problems.- Explore lambda function use cases.- Debug programs to enhance logic-building skills.	Proficiency in Python basics.
Data Structures in Python	Lists, Dictionaries, Sets, Tuples	- Practice CRUD operations on data structures.- Solve problems requiring advanced use of dictionaries and lists.	Strong command of Python data structures.
Object-Oriented Programming	Classes, Inheritance, Polymorphism	- Create small projects to apply OOP principles.- Practice solving real-world problems using Python's OOP concepts.	Competency in designing OOP-based programs.
Working with Databases	SQLite Database, CRUD Operations, SQL Queries	- Practice queries for creating, updating, and retrieving data.- Integrate Python with SQLite.- Use examples to implement database operations.	Mastery of database interaction with Python.

4. Software Engineering

Topic	Key Focus Areas	Preparation Method	Expected Output
SDLC Models	Agile, Waterfall, SCRUM	- Create diagrams of different SDLC models.- Understand use cases for each model.- Practice theoretical questions on model comparisons.	Clear understanding of SDLC variations.
UML Diagrams	Requirement Modeling, Behavioral/Structural Diagrams	- Practice drawing use case, class, and activity diagrams.- Analyze examples to understand diagram relationships.	Competency in UML design.
Testing Techniques	Unit, Integration, Regression, Black Box, White Box	- Practice writing test cases.- Understand manual and automation techniques.- Solve examples focusing on error detection in given code.	Proficiency in software testing methods.

5. Front End Technologies

Topic	Key Focus Areas	Preparation Method	Expected Output
JavaScript and jQuery	DOM Manipulation, Event Handling	- Build mini-projects focusing on DOM manipulation.- Solve coding challenges related to event-driven programming.	Mastery in DOM and JavaScript basics.
Angular	Components, Routing, Services, CRUD Operations	- Build small projects with Angular.- Practice CRUD operations using MongoDB and Angular.- Create a dynamic UI with bindings and observables.	Proficiency in Angular development.
React	JSX, State Management, CRUD Operations	- Develop React apps with CRUD operations.- Use hooks to manage state.- Practice data binding and creating reusable components.	Strong React programming skills.