Learn Research
Innovate

UNITED INSTITUTE OF TECHNOLOGY

(An Autonomous Institution)
(Approved by AICTE | Affiliated to Anna University |
Accredited by NAAC with A+ Grade | Certified by ISO 9001:2015)
Periyanaickenpalayam, Coimbatore – 641020

DEPARTMENT OF INFORMATION TECHNOLOGY

QUESTION BANK

II YEAR

ODD SEMESTER

ACADEMIC YEAR 2025 – 2026

INDEX

CLASS ADVISOR **HOD** **ACADEMIC DIRECTOR** **PRINCIPAL**

24ECPC306 -DIGITAL PRINCIPLES AND COMPUTER ORGANIZATION

UNIT I

UNIT I COMBINATIONAL LOGIC

Combinational Circuits – Karnaugh map -5 variable Karnaugh map, don't care conditions - Analysis and Design Procedures – Binary Adder-Subtractor – Decimal Adder - Magnitude Comparator – Decoder – Encoder – Multiplexers - De-multiplexers.

Q.No	Question	CO	BTL	Marks
	PART A			
1.	List the four possible elementary operations simple binary addition consists of.	1	RE	2
2.	Which combinational circuit is otherwise known as data selector ?why?	1	RE	2
3.	Outline a full adder by using two half adders and OR gate.	1	UN	2
4.	Compare the function of decoder and encoder .	1	UN	2
5.	Give disadvantage of half adder and full adder.	1	RE	2
6.	Draw the logic diagram of a 2 –bit multiplier?	1	RE	2
7.	The output Y of a 2-bit comparator is logic1whenever the 2-bit input A Is greater than the 2-bit input B. What is the number of combinations for which the output is logic 1?	1	RE	2
8.	Draw the logic circuit of a 2-bit comparator.	1	RE	2
9.	List out the applications of multiplexer.	1	RE	2
10.	In what way encoder differs from decoder?	1	RE	2
11	Write down the Borrow and Subtractor equation of full subtractor.	1	RE	2
12	Compare Sequential circuit and Combinational circuit	1	UN	2
13.	Outline the combinational circuit with three inputs and one output. The output is 1 when the binary value of the inputs is less than three. The output is zero otherwise	1	UN	2
14.	Recall how many selection inputs, data inputs and outputs for 1*32	1	RE	2

	Demultiplexer ?			
15.	Outline the following function using suitable multiplexer F= Σm (0,2,5,7).	1	RE	2
	PART B			
1.	Simplify below boolean function expression considering dont care conditions using 4 variable K map F(A,B,C,D) $=\Sigma(1,2,4,5,8,9,10,12)+d(3,6,7,13,14)$.	1	AN	16
2.	Develop a combinational circuit with three inputs, x, y and z and three outputs, A, B and C. When the binary input is 0, 1, 2 or 3 the binary output is one greater than the input.When the binary input is 4,5,6 or 7,The binary output is one less than the input.	1	AP	16
3.	Develop a 4-bit adder and subtractor circuit and explain.(8 mark each)	1	AP	16
4.	Construct 2-bit magnitude comparator with three outputs: A>B, A<B and A=B.	1	AP	16
5	Build BCD adder and explain how it differ from 4 bit binary adder	1	AP	16
6	Build three to eight line decoder circuit using inverters and AND gates. Also, present the truth table for the same	1	AP	16
7	(a) Develop the Boolean expression F (A, B, C) = Σm (0, 2, 5, 6) using 4:1 multiplexer. (b) Develop the Boolean expression F (A, B, C, D) = Σm (0,1,3,4,8,9,15) using suitable multiplexer.	1 1	AP AP	8 8
8	What is a K–Map? Simplify the Boolean function F (a,b,c,d,e) = Σ_m(1, 2, 4, 5, 6, 8, 9, 12, 13, 14,16,17,20,23,28,30,32) using K-Map.	1	AN	16

UNIT II

SYNCHRONOUS SEQUENTIAL LOGIC

. Introduction to Sequential Circuits – Flip-Flops – operation and excitation tables, Triggering of FF,Analysis and design of clocked sequential circuits – Design – Moore/Mealy models, state minimization, state assignment, circuit implementation - Registers – Counters.

Q.No	Question	CO	BTL	Marks
	PART A			
1.	Outline the difference between asynchronous sequential circuit and synchronous sequential circuit.	2	UN	2
2.	How does ripple counter differ from synchronous counter?	2	RE	2
3.	How do you eliminate the race around condition in a JK flip-flop?	2	RE	2
4.	Compare latches and flip-flops.	2	UN	2
5.	Explain about D-Latch with truth table.	2	UN	2
6.	List the different types of shift registers.	2	RE	2
7.	Define the terms :state table and state assignment.	2	RE	2
8.	Differentiate Mealy and Moore state machines.	2	UN	2
9.	Compare edge triggering and level triggering in sequential circuits	2	UN	2
10	Find minimum number of flip-flops needed to design a counter of Modulus 60.	2	RE	2
11	Construct a NAND based logic diagram of Master Slave JK FF.	2	AP	2
12	Illustrate the logic diagram of a clocked SR Flipflop.	2	UN	2
13	Show the operation of T FF.	2	RE	2
14	Define Counters and its types.	2	RE	2
15	Develop a Mealy machine to detect the sequence **"10"** and output **1** when detected. Otherwise, output **0**.	2	AP	2

PART B				
1.	Explain SR FF using NOR gate.	2	EV	16
2.	Explain in detail about 4 bit Johnson counter.	2	EV	16
3.	Which flip flop is called as data flip flop? explain the operation of same with its circuit diagram, characerstic table and excitation table.	2	EV	16
4.	Outline the design of a BCD Ripple counter using JK Flip flops with state diagram and logic diagram	2	UN	16
5	Develop a Mod-5 synchronous counter using JK flip flop.	2	AP	16
6	Summarize the design procedure for asynchronous sequential circuit.	2	UN	16
7	What are registers? Construct a 4 bit register using D FF and explain the operation of register.	2	AP	16
8	Explain Mealy and Moore Models with the help of block diagram	2	UN	16

UNIT III

COMPUTER FUNDAMENTALS

Functional Units of a Digital Computer: Von Neumann Architecture – Operation and Operands of Computer Hardware Instruction – Instruction Set Architecture(ISA):Memory Location, Address and Operation – Instruction and Instruction Sequencing – Addressing Modes, Encoding of Machine Instruction – Interaction between Assembly and High Level Language.

Q.No	QuestionQuestion	CO	BTL	Marks
PART A	**PART A**			
1	What is the role of control unit in the operation of digital computer?	3	RE	2
2	Define computer architecture.	3	RE	2
3	Name the functional units of computer.	3	RE	2
4	What is Von Neumann bottleneck?	3	RE	2
5	Outline instruction cycle with diagram..	3	UN	2
6	What is the role of PC?	3	RE	2
7	What is operation code?	3	UN	2
8	Interpret instruction set architecture (ISA)	3	UN	2
9	What are the data transfer instructions?	3	RE	2
10	Compare direct and indirect addressing modes.	3	UN	2
11	List the types of addressing modes.	3	RE	2
12	What is meant by an addressing mode? Mention most important of them?	3	RE	2
13	Distinguish between auto increment and auto decrement addressing mode.	3	UN	2
14	Define compiler.	3	RE	2
15	Compare machine level, assembly level and high level languages.	3	UN	2

PART B				
1.	A) Explain about fundamentals units in digital computer. B) Explain about instruction cycle.	3	UN UN	8 8
2.	Explain Von Neumann architecture with neat sketch and features.	3	UN	16
3	Explain in detail about the various operands of the computer hardware.	3	UN	16
4.	Illustrate Byte addressability (4) Big endian assignment (6) Little endian assignment (6)	3	UN	16
5	Mention four types of operations required to be performed by instruction in a computer. classify types of instruction formats? Give an example for each.	3	UN	16
6	Define addressing mode. Explain the basic addressing mode with an example for each.	3	EV	16
7	A) Explain about encoding in assembly language and types of instructions.(8) B) Discuss the interconnection between assembly language and high level language.(8)	3	EV	16
8	Explain the data transfer operations between memory and the processor.	3	EV	16

UNIT IV

PROCESSOR

Instruction Execution–Building a Data Path–Designing a Control Unit–Hard wired control Microprogrammed Control – Pipelining – Data Hazard – Control Hazards.

Q.No	QuestionQuestion	CO	BTL	Marks
PART A	**PART A**			
1.	List the operations involved in instruction cycle.	4	RE	2
2.	Outline the data path segment for arithmetic-logic instructions.	4	UN	2
3.	What is the ideal speed-up expected in a pipelined Architecture with ‘n’ stages? Justify your answer.	4	RE	2
4.	What do you mean by pipelining? List its types?	4	RE	2
5.	Compare static and dynamic techniques.	4	UN	2
6.	What is branch hazard?	4	RE	2
7.	What is meant by speculative execution?	4	RE	2
8.	Compare data hazards and control hazards.	4	UN	2
9.	Differentiate: Hardwired control and micro programmed control	4	UN	2
10.	What is program counter?	4	RE	2
11.	When do data hazard occur in pipelining?	4	RE	2
12.	What is a data path?	4	RE	2
13.	What is exception?	4	RE	2
14.	What is a branch prediction buffer?	4	RE	2
15.	What is the need for speculation?	4	RE	2
PART B				
1.	Outline the difference between hardwired control and micro programmed control.	4	UN	16

2.	What is a hazard? Analyze hazard free realization for the following Boolean functions F(A, B, C, D) = Σm (1, 5, 6, 7) using AND–OR gate network.	4	AN	16
3.	Explain why branch prediction algorithm needed? Differentiate between the static and dynamic techniques.	4	UN	16
4.	Illustrate pipeline hazards. Outline the types of pipeline hazards.	4	UN	16
5.	Construct a simple MIPS data path with control unit and explain the execution of ALU instruction.	4	AP	16
6.	Outline the methods for avoiding the control hazards.	4	UN	16
7.	Illustrate about data hazards. Explain with suitable techniques, how these hazards can be mitigated?	4	UN	16
8.	Explain how instruction is being fetched and executed through the data path in the processor?	4	UN	16

UNIT V

MEMORY AND I/O

Memory Concepts and Hierarchy – Memory Management – Cache Memories: Mapping and Replacement Techniques – Virtual Memory – DMA – I/O – Accessing I/O: Parallel and Serial Interface – Interrupt I/O – Interconnection Standards: USB, SATA.

Q.No	Question	CO	BTL	Marks
	PART A			
1.	What is a direct-mapped cache?	5	RE	2
2.	What is hit time?	5	RE	2
3.	Which signal is used to notify the processor that the transfer is completed? Define.	5	RE	2
4.	Mention the modes of DMA transfer.	5	RE	2
5.	Outline of interrupt driven I/O.	5	UN	2
6.	What is memory mapped I/O?	5	RE	2
7.	Define supervisor / kernel / executive state.	5	RE	2
8.	List the advantages of virtual memory?	5	RE	2
9.	Explain the concept of memory hierarchy in a computer system.	5	UN	2
10.	Compare main memory and cache memory based on access time and cost.	5	UN	2
11.	Illustrate the role of virtual memory with the help of a block diagram.	5	UN	2
12.	Summarize the need for Direct Memory Access (DMA) in data transfer.	5	UN	2
13.	Outline the differences between parallel and serial I/O interfaces.	5	UN	2
14.	Classify different cache mapping techniques.	5	UN	2
15.	Infer the importance of interconnection standards like USB and SATA.	5	UN	2
	PART B			

1.	Present an outline of virtual address, physical address, address translation, segmentation, page table, swap space and page fault.	5	UN	16
2.	Explain interconnection standards.	5	EV	16
3.	Construct a direct memory access with a diagram and mention its advantage.	5	AP	16
4.	Describe the various mechanisms for accessing I/O devices.	5	AN	16
5.	Explain how memory mapping techniques are useful for finding the memory blocks in cache?	5	EV	16
6.	How virtual addresses are translated into physical addresses? Explain it with the help of virtual memory organization and page translation.	5	EV	16
7.	Examine the role of I/O interfaces in system bottlenecks.	5	AN	16
8.	Analyze how cache memory improves performance with respect to memory hierarchy.	5	AN	16

24CSPC301-FOUNDATIONS OF DATA SCIENCE

UNIT I

FUNDAMENTALS OF DATA SCIENCE AND DATA ACQUISITION

Data Science: Benefits and uses – facets of data - Data Science Process: Overview – Defining research goals – Retrieving data – Data preparation - Exploratory Data analysis – build the model– presenting findings and building applications - Data Mining - Data Warehousing – Basic Statistical descriptions of Data.

Q.No	Question	CO	BTL	Marks
	PART A			
1.	What is data science?	1	RE	2
2.	List all stages in the data science lifecycle.	1	RE	2
3.	Name two popular programming languages used in Data Science.	1	RE	2
4.	Define data mining.	1	RE	2
5.	What is data warehousing?	1	RE	2
6.	What is structured data?	1	RE	2
7.	What is unstructured data?	1	RE	2
8.	What is meant by “building applications” in data science?	1	RE	2
9.	What does data retrieval mean?	1	RE	2
10.	Why is data preparation important before analysis?	1	UN	2
11.	Define one real-world application of data science.	1	UN	2
12.	Why is data science important in modern industries?	1	UN	2
13.	Why is Python considered a superior tool for data science compared to other tools?	1	UN	2
14.	Identify two applications of data science in education.	1	UN	2
15.	What is the primary goal of data science?	1	UN	2

PART B				
1	Illustrate the Data Science Life Cycle with a neat diagram and explain each phase.	1	UN	16
2	Explain the various benefits and uses of Data Science in modern industries with suitable examples.	1	UN	16
3	Differentiate between Data Mining and Data Warehousing with suitable examples.	1	UN	16
4	Explain the importance of data retrieval and preparation in the early stages of data science.	1	UN	16
5	Describe the process of building a model in the Data Science Life Cycle.	1	UN	16
6	Discuss how exploratory data analysis (EDA) helps in understanding data before modeling.	1	UN	16
7	Apply the Data Science process to analyze student performance data in an academic institution.	1	AP	16
8	Demonstrate how you would retrieve and prepare data from a public dataset for analysis.	1	AP	16

UNIT II

DESCRIBING DATA AND RELATIONSHIPS

Types of Data - Types of Variables -Describing Data with Tables and Graphs –Describing Data with Averages - Describing Variability - Normal Distributions and Standard (z) Scores. Correlation –correlation coefficient for quantitative data –computational formula for correlation coefficient – Populations, Samples and Probability – Sampling distribution of the mean

Q. No	Question	CO	BTL	Marks
	PART A			
1.	Define qualitative data with an example.	2	RE	2
2.	What is a frequency table?	2	RE	2
3.	What is range in statistics with example data?	2	RE	2
4.	What is standard deviation?	2	RE	2
5.	What is a normal distribution?	2	RE	2
6.	What is a population in statistics?	2	RE	2
7.	What is an event in probability?	2	RE	2
8.	What is mode in a dataset?	2	RE	2
9.	What is mean in descriptive statistics?	2	RE	2
10.	Define the concept of sample space with a coin toss experiment.	2	UN	2
11.	Why is standard deviation considered a better measure of dispersion than range?	2	UN	2
12.	Why is median preferred over mean for skewed data?	2	UN	2
12.	How do estimates of location help in summarizing data?	2	UN	2
13.	How is the mean affected by outliers?	2	UN	2
14.	What is the importance of probability in daily life scenarios?	2	UN	2

15.	Why is correlation coefficient important in data analysis?	2	UN	2
	PART B			
1.	Explain the differences between qualitative and quantitative data with suitable examples.	2	UN	16
2.	Discuss the concept of probability and its role in data analysis.	2	UN	16
3.	Compute the mean, median, mode, range and standard deviation using descriptive statistics for the given data. data = [2, 4, 4, 4, 6, 8, 10]	2	AP	16
4.	Construct the sample space for tossing two dice and derive the probability distribution for the sum of the outcomes.	2	AP	16
5.	Compare and contrast the different measures of variability and discuss their importance.	2	AN	16
6.	Analyze the correlation between two quantitative variables and determine if a linear relationship exists.	2	AN	16
7.	Justify the importance of understanding population vs. sample in inferential statistics, especially in survey-based research.	2	EV	16
8.	Justify the use of descriptive statistics in summarizing large datasets using a ca study.	2	EV	16

UNIT III

DATA EXPLORATION AND FEATURE ENGINEERING

Data analytics: descriptive analysis, diagnostic analytics, predictive analytics, predictive analytics; Data pre-processing: handling missing values — imputation techniques, dealing with outliers; Exploratory Data Analysis(EDA); Feature Engineering: One-hot encoding, label encoding, creating new features, dimensionality reduction techniques.

Q.No	QuestionQuestion	CO	BTL	Marks
PART A	**PART A**			
1.	Define descriptive analytics.	3	RE	2
2.	What is diagnostic analytics?	3	RE	2
3.	Mention any two differences between predictive and diagnostic analytics.	3	RE	2
4.	What is the goal of predictive analytics?	3	RE	2
5.	Define data pre-processing.	3	RE	2
6.	List two common causes of missing data.	3	RE	2
7.	What is mean imputation?	3	RE	2
8.	What is the purpose of one-hot encoding in data preprocessing?	3	UN	2
9.	Differentiate between one-hot encoding and label encoding with a brief example.	3	UN	2
10.	What is feature engineering and why is it important in data preparation?	3	UN	2
11.	Mention any two techniques used for dimensionality reduction.	3	UN	2
12.	Define the role of Exploratory Data Analysis (EDA) in the data science process.	3	UN	2
13.	List two common techniques used to handle outliers in a dataset.	3	UN	2
14.	Define label encoding and provide a simple example.	3	UN	2

15.	How is prescriptive analytics different from predictive analytics?	3	UN	2
	PART B			
1.	Describe the importance of exploratory data analysis (EDA) in data science.	3	UN	16
2.	Explain the various types of data analytics and their role in decision-making processes.	3	UN	16
3.	Apply one-hot encoding to an example dataset and explain the results.	3	AP	16
4.	Given the following dataset structure, identify and explain which type(s) of data analytics would be most appropriate to apply. Justify your reasoning with examples. **Dataset Description:** You are provided with data from a local retail store that contains the following fields: • Transaction_ID • Date of purchase • Customer age and gender • Item purchased • Item category (e.g., groceries, clothing, electronics) • Quantity purchased • Unit price • Payment mode (cash, card, UPI) • Customer feedback (1–5 rating)	3	AP	16
5.	Analyze the limitations of descriptive analytics in real-world decision-making.	3	AN	16
6.	Analyze the effect of outliers on model accuracy with and without outlier handling.	3	AN	16
7.	Assess the importance of feature creation in boosting model performance.	3	EV	16
8.	Justify the need for dimensionality reduction in high-dimensional datasets.	3	EV	16

UNIT IV

PYTHON LIBRARIES FOR DATA WRANGLING

Basics of Numpy arrays –aggregations –computations on arrays –comparisons, masks, boolean logic – fancy indexing – structured arrays – Data manipulation with Pandas – data indexing and selection – operating on data – missing data – Hierarchical indexing – combining datasets – aggregation and grouping – pivot tables.

Q.No	QuestionQuestion	CO	BTL	Marks
PART A	**PART A**			
1.	Define an array in NumPy.	4	RE	2
2.	How do you download and import the NumPy library in Python?	4	RE	2
3.	What function is used to create a 1D NumPy array?	4	RE	2
4.	Mention any two mathematical operations available in NumPy.	4	RE	2
5.	What is the use of pandas in Python?	4	RE	2
6.	Define a DataFrame in pandas.	4	RE	2
7.	What is the purpose of the info() and describe() method in a pandas DataFrame?	4	RE	2
8.	What is the role of NumPy in scientific computing?	4	UN	2
9.	What happens when two arrays of different shapes are added?	4	UN	2
10.	How does a DataFrame differ from a NumPy array?	4	UN	2
11.	How to select a single column from a pandas DataFrame?	4	UN	2
12.	What is the use of read_csv() method in pandas?	4	UN	2
13.	What is the use of reset_index() method in pandas?	4	UN	2
14.	How drop() works in pandas DataFrame?	4	UN	2
15.	What does combining tables mean in pandas and why is it useful?	4	UN	2

PART B				
1.	Explain the structure and purpose of NumPy arrays. How do they differ from Python lists?	4	UN	16
2.	Describe the basic operations that can be performed on a Pandas DataFrame with suitable examples.	4	UN	16
3.	Apply NumPy to create a one-dimensional and two-dimensional array. Demonstrate at least five array operations.	4	AP	16
4.	Demonstrate how to merge and join multiple tables using Pandas DataFrame with appropriate examples.	4	AP	16
5.	Break down the steps to identify and handle missing values using functions like isnull(), dropna() and fillna() in Pandas.	4	AN	16
6.	Analyze the impact of handling different file formats efficiently in data preprocessing using Pandas.	4	AN	16
7.	Evaluate the scalability of Pandas when working with big data. How can limitations be mitigated through optimization techniques or alternative tools?	4	EV	16
8.	Critically assess the importance of combining multiple tables in a relational format using Pandas. How does this benefit real-world applications like healthcare or e-commerce?	4	EV	16

UNIT V

DATA VISUALIZATION

Importing Matplotlib – Line plots – Scatter plots – visualizing errors – density and contour plots – Histograms – legends – colors – subplots – text and annotation – customization – three dimensional plotting - Geographic Data with Basemap - Visualization with Seaborn. Case Study : PowerBI

Q.No	Question	CO	BTL	Marks
	PART A			
1.	How do you download and import the Matplotlib library in Python?	5	RE	2
2.	Define a line plot.	5	RE	2
3.	What is scatter plots?	5	RE	2
4.	Which function is used to create histograms in Matplotlib?	5	RE	2
5.	What is legends in data visualization?	5	RE	2
6.	What are subplots in Matplotlib?	5	RE	2
7.	Name the library used for geographic data visualization in Python.	5	RE	2
8.	What is the role of Seaborn in data visualization?	5	RE	2
9.	What is the difference between a line plot and a scatter plot?	5	UN	2
10.	How legends enhance the readability of a chart?	5	UN	2
11.	How customization can improve data visualization in Matplotlib?	5	UN	2
12.	What is the importance of colors in communicating data insights?	5	UN	2
13.	What is the benefit of using 3D plots over 2D plots?	5	UN	2
14.	Why are error bars important in visualizing experimental data?	5	UN	2
15.	Why is annotation useful in a graph or chart?	5	UN	2
	PART B			

1.	Explain the purpose of line plots in data visualization and how they help interpret trends over time.	5	UN	16
2.	Describe how histograms can help in understanding data distribution.	5	UN	16
3.	Using the given data points, construct a scatter plot to compare two variables, x and y and explain the observed pattern or relationship. Given: x = [10, 20, 30, 40, 50] y = [15, 25, 35, 30, 45]	5	AP	16
4.	Using Matplotlib, develop a subplot layout to display monthly temperature trends for the years 2020, 2021 and 2022 in a single figure. Explain how subplots help in comparing trends across years effectively.	5	AP	16
5.	Analyze how the choice of colors in data visualizations can either mislead or enhance data interpretation. Support your answer with relevant examples, such as how using similar shades for different categories or inappropriate color gradients can lead to confusion or misinterpretation of the data.	5	AN	16
6.	Analyze the use of Basemap in representing demographic and geographic trends. Support your analysis with a relevant example, such as visualizing population density across different regions of India.	5	AN	16
7.	Evaluate the use of scatter plots in determining correlation strength and direction.	5	EV	16
8.	Evaluate the effectiveness of using annotations in communicating insights from plots.	5	EV	16

24ITPC301 - DATA STRUCTURES AND ALGORITHMS

UNIT I

ABSTRACT DATA TYPES

Abstract Data Types (ADTs) – ADTs and classes – Introduction to analysis of algorithms – Asymptotic notations - Recursion – Analyzing recursive algorithms with examples such as Fibonacci numbers, Binary search and Towers of Hanoi.

Q.No	Question	CO	BTL	Marks
	PART A			
1	Define an Abstract Data Type (ADT).	1	RE	2
2	List two characteristics of ADTs.	1	RE	2
3	Why ADTs are useful in software design?	1	RE	2
4	What's the relationship between ADTs and classes?	1	RE	2
5	Name one way classes extend ADTs.	1	RE	2
6	Define analysis of algorithms.	1	RE	2
7	Why analyze algorithms using asymptotic growth?	1	RE	2
8	Name three asymptotic notations.	1	RE	2
9	What does Big O notation represent?	1	RE	2
10	Define recursion in algorithms.	1	RE	2
11	What are the two essential components of a recursive function?	1	RE	2
12	Write the recurrence for Towers of Hanoi.	1	RE	2
13	State the time complexity of recursive Fibonacci (simple form).	1	RE	2

14	Why binary search is logarithmic in complexity?	1	RE	2
15	How many moves are required in Towers of Hanoi and why?	1	UN	2
PART				
1	Explain how Abstract Data Types (ADTs) are used in software design. Illustrate with an example comparing an ADT and its class-based implementation.	1	UN	16
2	Compare and contrast ADTs with classes in object-oriented programming.	1	AP	16
3	Explain the importance of asymptotic notation in algorithm evaluation. Contrast Big-O, Big-Ω, and Big-Θ with examples.	1	UN	16
4	Using an example, apply asymptotic analysis to derive the time complexity of binary search.	1	AP	16
5	Given the Fibonacci sequence defined recursively, analyze its time complexity and discuss strategies to optimize it.	1	AN	16
6	Trace the recursive solution to the Towers of Hanoi for n disks. Derive the recurrence and solve it.	1	AP	16
7	Compare and analyze recursive vs. iterative approaches for the factorial and binary search functions, focusing on performance and stack usage.	1	AN	16
8	Design a divide-and-conquer algorithm for a problem of your choice (e.g., merge sort), derive its recurrence, and analyze its time complexity using the Master Theorem.	1	AN	16

UNIT II

LINEAR STRUCTURES

List ADT – Array-based implementations – Linked List Implementations – Singly Linked Lists – Circularly Linked Lists – Doubly Linked Lists – Stack ADT – Applications of Stack -Queue ADT – Applications of Queue.

Q.No	Question	CO	BTL	Marks
	PART A			
1.	Define Abstract Data Type (ADT).	2	RE	2
2.	How the List ADT can be implemented using arrays?	2	RE	2
3.	Define array-based implementation of a stack.	2	RE	2
4.	Define array-based implementation of a queue.	2	RE	2
5.	Define a linked list.	2	RE	2
6.	List the basic operations on a singly linked list.	2	RE	2
7.	Define a singly linked list structure.	2	RE	2
8.	Define circularly linked list.	2	RE	2
9.	Give two use cases of a circular linked list.	2	RE	2
10.	Define doubly linked list.	2	RE	2
11.	State two applications of a doubly linked list.	2	RE	2
12.	Define Stack ADT and its operations.	2	RE	2

13.	Mention two applications of a stack.	2	RE	2
14.	Define Queue ADT and its operations.	2	RE	2
15.	List application of a queue.	2	RE	2
PART B				
1.	Design and implement an Array-based List ADT supporting insertion, deletion, traversal, and search. Discuss its time and space complexity, and describe how boundary conditions (e.g., full/empty) are handled.	2	AP	16
2.	Critically analyze insertion and deletion operations in a Singly Linked List. Compare efficiency and complexity with array-based implementation, and provide pseudocode or diagrams illustrating node link adjustments.	2	AN	16
3.	Develop and explain an algorithm to insert and delete nodes in a Circularly Linked List. Evaluate its benefits and limitations versus both singly and doubly linked lists in terms of traversal efficiency and complexity.	2	AN	16
4.	Analyze the implementation of a Doubly Linked List. Show how backward and forward traversals work, and discuss scenarios (e.g., cache, undo functionality) where its bidirectional nature provides clear advantages.	2	AN	16
5.	Design both array-based and linked list implementations of the Stack ADT. Compare their memory usage, overflow/underflow handling, and operation runtime. Provide pseudocode for push/pop operations.	2	AP	16
6.	Explain three real-world applications of stacks: recursion, expression evaluation (infix to postfix), and undo/redo functionality. Illustrate each with examples or flow diagrams.	2	UN	16
7.	Implement a Queue ADT using both array (circular buffer) and linked list. Provide enqueue, dequeue, front, size, and isEmpty operations. Discuss how wrap-around is handled in the circular buffer implementation.	2	AP	16

8.	Explain the use of queues in breadth-first search, printer scheduling, CPU task scheduling, and network packet handling. Show how FIFO behavior supports each application and discuss performance considerations.	2	UN	16

UNIT III

SEARCHING AND SORTING

Linear Search – BinarySearch - Bubble Sort – Selection Sort – Insertion Sort – Merge Sort – Quick Sort – Analysis of sorting algorithms – Hashing – Hash Functions – Collision Handling - Rehashing - Double Hashing .

Q.No	QuestionQuestion	CO	BTL	Marks
	PART APART A			
1.	Define Linear Search.	3	RE	2
2.	State the time complexity of Linear Search in average and worst case.	3	RE	2
3.	Define Binary Search.	3	RE	2
4.	Why Binary Search is better than Linear Search on sorted data?	3	RE	2
5.	Define Bubble Sort.	3	RE	2
6.	Define Selection Sort.	3	RE	2
7.	Define Insertion Sort.	3	RE	2
8.	What is Merge Sort?	3	RE	2
9.	What is Quick Sort?	3	RE	2
10.	Compare Bubble Sort and Quick Sort in terms of efficiency.	3	UN	2
11.	State the worst-case time complexities for Merge Sort and Quick Sort.	3	RE	2
12.	Why Merge Sort guarantees better worst-case performance than Quick Sort?	3	RE	2

13.	What is a hash function?	3	RE	2
14.	Define collision handling in hashing.	3	RE	2
15.	Define double hashing.	3	RE	2
PART B				
1.	Using a given unsorted integer array, demonstrate linear search and binary search to find a target value. Compare and discuss the number of comparisons each performs.	3	AP	16
2.	Apply bubble sort, selection sort, and insertion sort to the list: [4, 2, 7, 1, 3]. Provide each pass and count swaps or shifts. Then recommend which is best for nearly sorted lists.	3	AP	16
3.	Explain merge sort and quick sort on an array of 8 elements: identify recursion tree height, total comparisons, and best/worst-case scenarios. Explain which is preferred for large, nearly sorted data.	3	UN	16
4.	Compare and analyze time and space complexities of all discussed sorting algorithms. Present a table summarizing best, average, worst cases, stability, and whether they are in-place.	3	AN	16
5.	Using a hash table of size 11, hash keys [54, 26, 93, 17, 31, 77, 44] using the division method. Show the table state after insertion, calculate load factor, and demonstrate search for key 44.	3	AP	16
6.	Explain collision handling by (a) separate chaining, (b) open addressing with linear probing, (c) quadratic probing, and (d) double hashing, using key sequence [20, 31, 42, 53, 64] in a table of size 7. Show final states.	3	UN	16
7.	Analyze rehashing: For open addressing hash table with $\lambda max = 0.75$ and $\lambda min = 0.25$, demonstrate insertions triggering resize. Show before-and-after states and analyze performance implications.	3	AN	16
8.	Compare double hashing and linear probing in terms of clustering and efficiency. Derive expected probe counts for successful/unsuccessful searches as $\lambda \rightarrow 0.75$, using theoretical formulas. Conclude which	3	AN	16

	technique is preferable.			

UNIT IV

TREE STRUCTURE

Tree ADT – Binary Tree ADT – Tree Traversals – Expression Trees - Binary Search trees: Insertion and Deletion – AVL Trees – Binary Heap:Properties - Operations: Insert- Findmin and Findmax - DeleteMin - Heap sort - SplayTrees - RedBlack Tree.

Q.No	Question	CO	BTL	Marks
	PART A			
1.	Define the Tree Abstract Data Type (ADT).	4	RE	2
2.	What distinguishes a binary tree ADT from a general tree ADT?	4	RE	2
3.	List the three standard recursive tree traversal methods.	4	RE	2
4.	Why inorder traversal of a binary search tree yields sorted output?	4	RE	2
5.	What is an expression tree?	4	RE	2
6.	Show how inorder traversal of an expression tree corresponds to the original infix expression.	4	UN	2
7.	What property must a binary search tree satisfy for insertion?	4	RE	2
8.	What happens during a deletion of a node with two children in a BST?	4	RE	2
9.	Define an AVL tree.	4	RE	2
10.	When and why rotations are used in AVL tree insertion?	4	RE	2

11.	State the two main properties of a binary heap.	4	RE	2
12.	What is the process of inserting a new element into a min-heap?	4	RE	2
13.	What is a splay tree?	4	RE	2
14.	Define the splay operation (e.g., zig-zig or zig-zag) and its purpose.	4	RE	2
15.	What extra information does each node in a red–black tree carry?	4	RE	2
PART B				
1.	Given a Tree ADT and its generic node structure, implement a C or Java function that converts a general Tree into its equivalent Binary Tree using the "left-child, right-sibling" technique. Demonstrate your method on a sample 4-node tree and explain how the representation changes.	4	AP	16
2.	For the postfix expression a b c + * d - e f + *, construct the corresponding expression tree. Then: a) Provide the infix and prefix expressions by performing inorder and preorder traversals. b) Analyze and explain why traversal order preserves operator precedence without parentheses.	4	AN	 8 8
3.	a) Starting from an empty BST, insert the keys 50, 30, 70, 20, 40, 60, 80, 35. b) Delete node 30 (which has two children) and detail the steps: i) Show the tree structure after deletion, ii) Identify successor or predecessor used, iii) Explain rotations (if any) needed to maintain BST properties.	4	AP	6 10
4.	a) Insert keys [10, 20, 30, 25] into an AVL tree one by one. b) After each insertion, compute balance factors of all affected nodes. c) When the tree becomes unbalanced, analyze the imbalance type (LL, LR, RL, RR) and demonstrate the correct rotation(s) to restore balance.	4	AN	5 5 6

5.	a) Build a min-heap by inserting keys [9, 4, 15, 6, 10, 2, 8] sequentially, explaining each "up-heap" operation. b) Then perform delete Min and show the resulting heap, detailing the "down-heap" process. c) Finally, apply heap sort on the resulting structure and list the sorted output.	4	AP	5 5 6
6.	Compare and analyze the time and space complexity of heap sort vs merge sort**.**	4	AN	16
7.	a) Starting with an empty splay tree, insert keys [10, 20, 30, 40, 50] using standard BST insertion followed by splaying each inserted node to the root. b) Demonstrate how the tree evolves, showing zig, zig-zig, or zig-zag rotations performed during each splay.	4	AP	8 8
8.	a) Insert the sequence [30, 10, 20, 40, 50, 25] into an initially empty Red–Black Tree, following insertion rules and fixing violations (coloring, rotations). b) After each insertion, analyze any red/red or black-depth violations, and document how you resolve them.	4	AN	8 8

UNIT V

GRAPH STRUCTURES

Graph ADT – Representations ofGraph – Graph Traversals–Topological Ordering – Greedy algorithms – Shortest paths:Dijikstra's algorithm – MinimumSpanning trees: Prim's algorithm, Kruskal's algorithm - Bi-connectivity -The Classes P and NP - An NP complete Problem.

Q.No	Question	CO	BTL	Marks
	PART A			
1.	Define Graph ADT.	5	RE	2
2.	List two common graph representations.	5	RE	2
3.	Define adjacency matrix vs adjacency list.	5	RE	2
4.	What are the two primary graph traversal methods?	5	RE	2
5.	Differentiate DFS and BFS.	5	UN	2
6.	Define topological sort.	5	RE	2
7.	Define a greedy algorithm.	5	RE	2
8.	Give an example in graph context.	5	UN	2
9.	State the purpose of Dijkstra's algorithm.	5	RE	2
10.	Define a Prim's algorithm.	5	RE	2
11.	Define Kruskal's algorithm.	5	RE	2

12.	Define biconnectivity in graphs.	5	RE	2
13.	Define P and NP.	5	RE	2
14.	Give two NP-complete examples.	5	UN	2
15.	List the NP-complete significance.	5	RE	2
PART B				
1.	Given an undirected graph representing a network of 8 cities (nodes) with weighted connections (edges), model its Graph ADT using adjacency lists. Then, write a function to insert a new weighted edge and another to remove an existing edge.	5	AP	16
2.	For the graph above, apply both BFS and DFS starting from city node 0. Demonstrate the exact traversal order for each and discuss which traversal would be better suited for finding the shortest number of hops versus detecting cycles. Cite examples.	5	AP	16
3.	You are given a DAG representing 6 tasks and their dependencies. Using Kahn's algorithm, produce a topological order. Then analyze and explain how topological ordering helps in scheduling tasks, and whether the ordering is unique or not in this scenario.	5	AN	16
4.	Run Dijkstra's algorithm on a weighted directed graph of 7 nodes with non-negative weights, starting from node A. Show each step's tentative distances, the priority queue states, and finalize the shortest distances.	5	AP	16
5.	For a connected undirected graph with 8 nodes and weighted edges, construct MSTs using both Prim's algorithm (starting from node 1) and Kruskal's algorithm. Compare both results (edges, order of selection, total weight) and analyze scenarios where they might differ.	5	AN	16
6.	Explain the concept of bi-connectivity in undirected graphs. Using Tarjan's algorithm, identify articulation points and biconnected components in a sample graph. Analyze why these points are critical	5	UN	16

	and how they affect network resilience.			
7.	Explain the complexity classes P and NP. Given the decision version of the Hamiltonian Path problem**,** demonstrate how you would verify a proposed solution (certificate) in polynomial time.	5	UN	16
8.	Prove that the Subset-Sum problem is NP-complete by: i) Showing it is in NP (via certificate verification). ii) Reducing a known NP-complete problem (e.g., 3-SAT or Partition) to Subset-Sum, detailing the reduction process.	5	AN	 8 8

24CSP303

OBJECT ORIENTED PROGRAMMING

UNIT I

BASICS OF OOP AND JAVA FUNDAMENTALS

Introduction to OOP– Java Fundamentals -Data Types, Variables, and Arrays Operators- Control Statements – Classes – Methods –Constructors- Garbage Collection.

Q.No	Question	CO	BTL	Marks
	PART A			
1.	Define Object Oriented Programming.	1	RE	2
2.	How can we create an instance of a class in java?	1	RE	2
3.	Define abstraction and encapsulation.	1	RE	2
4.	List the features of Object Oriented Programming	1	RE	2
5.	How to create an instance of a class in java	1	UN	2
6.	What are the three major sections of java source file?	1	RE	2
7.	List out the source file declaration rules.	1	RE	2
8.	What are public static void main(String args[]) and System.out.println()?	1	UN	2
9.	What is bytecode?	1	RE	2
10.	Define static variable and static method.	1	RE	2

11.	What is a variable? How to declare variable in java?	1	RE	2
12.	List out the operator in java.	1	RE	2
13.	Write a java program using control flow statements.	1	UN	2
14.	What are constructors?	1	RE	2
15.	What is Garbage collection?	1	RE	2
	PART B			
1.	Explain the various object oriented concepts with an example.	1	Und	16
2.	i) Compare and contrast between break and continue ii) Write a java program to display following number pattern? 1 1 2 1 2 3 1 2 3 4 1 2 3 4 5	1	UN	8 8
3	Classify the following: i) Control flow statements in Java ii) Data types in Java	1	UN	8 8

4.	Explain how two numbers can be swapped with and without temporary variables. Write a java program for each.	1	AN	16
5.	Explain what a constructor is. Describe its types with suitable examples.	1	UN	16
6.	Make use of a simple Java program to implement basic Calculator operations.	1	AP	16
7.	+0Develop a program to perform the following functions using classes, objects, constructors and destructors where essential. Get as input the marks of 5 students in 5 subjects. a. Calculate the total and average. b. Print the formatted result on the screen.	1	AP	16
8.	Explain the arrays and its types in detail with example program.	1	UN	16

UNIT II

INHERITANCE AND EXCEPTION HANDLING

Inheritance –Packages and Interfaces - Exception Handling Fundamentals – Java's Built-in Exceptions-Creating new Exception subclasses

Q.No	Question	CO	BTL	Marks
	PART A			
1.	Define an interface.	2	RE	2
2.	What is the use of Inheritance and what are its advantages?	2	RE	2
3.	Tell how interfaces can be extended.	2	RE	2
4.	What is object cloning?	2	RE	2
5.	What are the properties of nested interface?	2	RE	2
6.	How can you access the super class version of an overridden method?	2	UN	2
7.	Whether a subclass can access all members of super class? How?	2	RE	2
8.	Define Exception.	2	RE	2
9.	Give the advantages of using exception handling mechanism	2	RE	2
10.	Difference Between exception and error.	2	UN	2

11.	What is Exception handling?	2	RE	2
12.	What is ArrayIndexOut Of Bounds Exception?	2	RE	2
13.	Difference Between exception and error.	2	UN	2

14.	What are the various segments of an exception handling mechanism?	2	UN	2
15.	Tell about Package in Java.	2	RE	2
	PART B			
1.	Define Inheritance? With diagrammatic illustration and java programs illustrate the different types of inheritance	2	AP	16
2.	Define multiple inheritance and how to perform multiple inheritance in Java.	2	AP	16
3.	What is object cloning? Why it is needed? Explain how objects are cloned?	2	UN	16
4.	Explain in detail about Java's Built-in Exceptions. Explain any three Exceptions.	2	AN	16
5.	Analyze a Java program that uses custom exceptions to handle errors. Explain how the program manages errors and compare it with handling using built-in exceptions.	2	AN	16

6.	Describe the following concepts with example i. Try-catch-throw paradigm. ii. Exception specification.	2	AP	 8 8
7.	Discuss about the java error handling mechanism? What is the difference between 'unchecked exceptions' and 'checked exceptions'?	2	CR	16

8.	Write a Java program using multiple catch blocks and finally block ? Explain the output.	2	UN	16

UNIT III

POLYMORPHISM AND MULTITHREADING IN JAVA

Polymorphism- Abstract classes and methods-Overloading-Overriding-final methods and classes –Multithreaded programming –The Thread class and the Runnable Interface-Creating multiple threads-Synchronization-Autoboxing and Annotations (Metadata).

Q.No	Question	CO	BTL	Marks
	PART A			
1.	Define polymorphism in Java.	3	RE	2
2.	What is method overloading? Give an example.	3	RE	2
3.	Define abstract class.	3	RE	2
4.	What is method overriding in Java?	3	RE	2
5.	"Thread is a lightweight process" - Discuss	3	UN	2
6.	What do you mean by synchronization?	3	RE	2
7.	What are the three ways by which the thread can enter in waiting stage?	3	RE	2
8.	Mention two ways to create threads in Java.	3	UN	2
9.	When to use final variable and method?	3	RE	2
10.	Differentiate between process and thread.	3	RE	2

11.	What is mean by thread constructor?	3	RE	2
12.	Why is synchronization important in Java multithreading?	3	RE	2
13.	Name any four thread constructor.	3	UN	2
14.	Define autoboxing with a small example.	3	RE	2
15.	What is annotation in Java?	3	RE	2
PART B				
1.	Classify polymorphism in Java with examples for both method overloading and overriding.	3	AN	16
2.	Analyze method overloading and method overriding. Explain differences with syntax and examples	3	AN	16
3	Describe the life cycle of thread and various thread methods.	3	UN	16
4.	Explain Abstract classes with an example program. Also describe the properties of abstract classes	3	AN	16
5.	How do you create multiple threads with an example	3	UN	16
6.	What is synchronization in multithreading? Explain with an example program showing how to handle race conditions	3	AP	16
7.	With illustrations explain multithreading, interrupting threads, thread states and thread properties.	3	AP	16
8.	Explain the concept of autoboxing and unboxing. Write a Java program to demonstrate both.	3	UN	16

UNIT IV

GENERIC TYPES, METHODS, AND COLLECTIONS FOR ADTS AND ALGORITHMS

Introduction to Generics and Collection Types- Restrictions, Inheritance, and Reflection in Generics. Generic Collections and Iteration – Working with Generic collections. Algorithms for Collections – Sorting and Searching. User defined Algorithms.

Q.No	Question	CO	BTL	Marks
	PART A			
1.	List out motivation needed in generic programming.	4	RE	2
2.	Define Java I/O basics.	4	RE	2
3.	Write the Restrictions and Limitations of generic type.	4	UN	2
4.	Illustrate why generic programming is required?	4	UN	2
5.	Show an example of a generic class.	4	UN	2
6.	How can generic be used with inheritance in several ways? What are they?	4	RE	2
7.	List any two challenges of generic programming in virtual machine.	4	RE	2
8.	How does inheritance work with generics?	4	RE	2
9.	What is the difference between List<Object> and List<?>?	4	RE	2
10.	Give an example of generic method declaration.	4	RE	2

11.	Mention two commonly used generic collection interfaces.	4	RE	2
12.	Write the syntax of declaring a generic class.	4	UN	2
13.	What is the use of reflection in generics?	4	RE	2
14.	Name any two generic collection types in Java.	4	RE	2
15.	What is the purpose of the Collections.sort() method?	4	RE	2
PART B				
1.	(i) Define generic programming and explain with an example about generic classes . (8) (ii) Write in detail about generic functions with an example. (8)	4	RE	16
2.	Write a program to count the total no. of chars, words, lines, alphabets, digits, white spaces in a given file.	4	RE	16
3.	Analyze how Generics Improve Type Safety with an example.	4	AN	16
4.	Explain the generic collection types available in Java. Give examples of List, Set, and Map.	4	UN	16
5.	Discuss the generic collection types available in Java. How can iteration be performed over generic collections with examples?	4	AP	16
6.	Write a user-defined generic algorithm for finding the maximum value in a list. Explain the concept with implementation and usage.	4	RE	16
7.	Discuss sorting and searching algorithms available in the Collections framework. How are they used with generics?	4	AN	16
8.	Write a user-defined generic algorithm in Java to find the maximum of three elements. Explain with syntax and output	4	RE	16

UNIT V

FILES AND STREAMS IN JAVA

Files and streams –Byte Stream-I/O Stream, File I/O Stream, Byte Array I/O Stream- Character Stream-Error Handling in File I/O . File Reader and Writer, CharArrayReader and Writer-Serialization

Q.No	Question	CO	BTL	Marks
	PART A			
1.	Define Java I/O basics.	5	RE	2
2.	List the byte stream classes.	5	RE	2
3.	What is relative file name?	5	RE	2
4.	What are streams?	5	RE	2
5.	Express a Java code to check if the command line argument is file or not.	5	UN	2
6.	What is absolute file name?	5	RE	2
7.	How does FileWriter handle existing files?	5	RE	2
8.	What is the purpose of the FileReader and FileWriter classes in Java?	5	UN	2
9.	Write the use of seek method.	5	RE	2
10.	What is serialization in Java?	5	RE	2
11.	How is error handling done during File I/O operations?	5	RE	2

12.	Write the syntax for creating a FileInputStream object.	5	RE	2
13.	Define ByteArrayOutputStream	5	UN	2
14.	Mention the use of CharArray Reader class.	5	UN	2
15.	Summarize the use of Input Stream Reader and Output Stream Writer?	5	UN	2
	PART B			
1.	Write a program to count the total no. of chars, words, lines, alphabets, digits, white spaces in a given file.	5	RE	16
2.	Write a program that copies the content of one file to another file	5	UN	16
3.	Describe in detail about FileInputStream and FileOutputStream with an example.	5	UN	16
4.	(i)Summarize in detail about String and String constructor. (ii)Discuss in detail about String handling functions.	5	UN	8 8
5.	Summarize the concept of streams and stream classes and their classification	5	UN	16
6.	While reading a file how would you check whether you have reached the end of the file.	5	AP	16
7.	Illustrate in brief about i. Reading from a file. ii. Writing in a file.	5	UN	16
8.	Explain FileReader, FileWriter, CharArrayReader, CharArrayWriter, and Serialization in Java.	5	RE	16

24MABS301
DISCRETE MATHEMATICS

UNIT I

LOGIC AND PROOFS

Propositional logic – Propositional equivalences - Predicates and quantifiers – Nested quantifiers – Rules of inference - Introduction to proofs.

Q.No	Question	CO	BTL	Marks
	PART A			
1.	State the truth table of "If tigers have wings then the earth travels round the sun".	1	UN	2
2.	Give the converse and contra positive of the implication "If it is raining, then I get wet".	1	UN	2
3.	Show that P→Q and ¬P∨Q are equivalent.	1	UN	2
4.	Construct the truth table for the compound proposition (p →q)→(q→p).	1	UN	2
5.	Symbolize the statement "Good food is not cheap".	1	UN	2
6.	Show that ¬ (P→Q) and P∧¬Q are equivalent.	1	UN	2
7.	Verify (R→S) ∨¬(R→S) is a tautology.	1	UN	2
8.	Verify P∨Q→P is a tautology.	1	UN	2
9.	Construct the truth table for the compound proposition $(p \to q) \to (q \to p)$.	1	UN	2
10.	State Demorgan's laws of logic.	1	RE	2
11.	Write the following sentence in a symbolic form "Everyone who is healthy can do all kinds of work".	1	UN	2
12.	What are the negations of the statements $\forall x(x^2 > x)$ *and* $\exists x(x^2 = 2)$?	1	RE	2
13.	Express the following statement using predicators and quantifiers, "All men are mortal"	1	UN	2
14.	When a set of formulae is consistent and inconsistent?	1	RE	2
15.	Determine whether the conclusion C follows logically from the premises H_1 and H_2 or not. H_1 : P→Q, H_2 : P, C : Q	1	UN	2
	PART B			
1	(i) Construct the truth table for the following statement $\neg(p \vee (q \wedge r)) \leftrightarrow ((p \vee q) \wedge (p \vee r))$.	1	UN	8
	(ii) Show that $(\neg P \wedge (\neg Q \wedge R)) \vee (Q \wedge R) \vee (P \wedge R) \Leftrightarrow R$.	1	UN	8

2	(i) Show that $p \vee (q \wedge r)$ *and* $(p \vee q) \wedge (p \vee r)$ are logically equivalent.	1	AP	8
	(ii) Without constructing the truth table find the PDNF and PCNF of $(\neg P \rightarrow R) \wedge (Q \leftrightarrow P)$.	1	AP	8
3	(i) Without constructing the truth table find the PDNF and PCNF of $P \rightarrow (Q \wedge R) \wedge (\neg P \rightarrow (\neg Q \wedge \neg R))$	1	AP	8
	(ii) Find the PDNF of the statement, $(q \vee (p \wedge r)) \wedge \neg((p \vee r) \wedge q)$.	1	AP	8
4	(i) Find the disjunctive normal form of $p \rightarrow ((p \rightarrow q) \wedge \neg (\neg q \vee \neg p))$.	1	UN	8
	(ii)Without using truth tables, show that $Q \vee (P \wedge \neg Q) \vee (\neg P \wedge \neg Q)$ is a tautology.	1	AP	8
5	(i) Using indirect method of proof, derive $P \rightarrow \neg S$ from $P \rightarrow (Q \vee R)$, $Q \rightarrow \neg P$, $S \rightarrow \neg R$, P.	1	AP	8
	(ii) Use indirect method of proof, $(\forall x)(P(x) \vee Q(x)) \Rightarrow (\forall x)P(x) \vee (\exists x)Q(x)$.	1	AP	8
6	(i)Show that $R \rightarrow S$ can be derived from the premises $P \rightarrow (Q \rightarrow S)$, $\neg R \vee P$ and Q.	1	AP	8
	(ii)Show that $R \vee S$ follows logically from the premises $C \vee D$, $(C \vee D) \rightarrow \neg H$, $\neg H \rightarrow (A \wedge \neg B)$ and $(A \wedge \neg B) \rightarrow (R \vee S)$.	1	AP	8
7	(i) Use the indirect method to prove that the conclusion $\exists z Q(z)$ follows from the premises $\forall x(P(x) \rightarrow Q(x))$ *and* $\exists y P(y)$.	1	AP	8
	(ii)Show that S is valid inference from the premises $P \rightarrow \neg Q, Q \vee R, \neg S \rightarrow P$ and $\neg R$.	1	AP	8
8	(i)Show that the premises $R \rightarrow \neg Q, R \vee S, S \rightarrow \neg Q, P \rightarrow Q, P$ are inconsistent.	1	AP	8
	(ii)Show that d can be derived from the premises $(a \rightarrow b) \wedge (a \rightarrow)$, $\neg(b \wedge c), d \vee a$	1	AP	8

UNIT II

COMBINATORICS

Mathematical induction – Strong induction and well ordering – The basics of counting – Thepigeonhole principle – Permutations and combinations – Recurrence relations – Solving linear recurrence relations – Inclusion and exclusion principle and its applications.

Q.No	Question	CO	BTL	Marks
	PART A			
1.	State the principle of Mathematical induction.	2	RE	2
2.	State the principle of Strong induction.	2	RE	2
3.	How many ways are there to select five players from a 10 member tennis team to make a trip to a match at another school?	2	UN	2
4.	What is the number of arrangements of all the six letters in the word PEPPER ?	2	UN	2
5.	In how many ways can the letters of the word MISSISSIPPI be arranged?	2	UN	2
6.	Find the number of permutations of the letters of the word MATHEMATICS.	2	UN	2
7.	State the Pigeonhole principle.	2	RE	2
8.	If 9 colours are used to paint 100 houses, show that at least 12 houses will be of the same colour.	2	UN	2
9.	If we select any group of 1000 students on campus . show that atleast 3 of them must have same birthday .	2	UN	2
10.	Define recurrence relation .	2	RE	2
11.	Find the recurrence relation for the Fibonacci sequence	2	UN	2
12.	Find the first four terms of the sequence defined by the recurrence relation and initial condition$a_n = a_{n-1}^2$, $a_1 = 2$.	2	UN	2
13.	Find the recurrence relation satisfying the equation $y_n = A(3)^n + B(-4)^n$	2	UN	2
14.	Write an explicit formula for a_n if $a_n = 3a_{n-1}$ and $a_1 = 2$.	2	UN	2
15.	Write the Principle of Inclusion and Exclusion.	2	RE	2
	PART B			
1	(i) Using mathematical induction to show that $1^2 + 2^2 + 3^2 + \cdots \ldots + n^2 = \frac{n(n+1)(2n+1)}{6}$whenever n is a positive integer.	2	AP	8

	(ii) Using mathematical induction prove that $1^2 + 3^2 + 5^2 \ldots\ldots + (2n-1)^2 = \frac{n(2n-1)(2n+1)}{3}$ for all $n \geq 1$.	2	AP	8
2	(i) Use mathematical induction to show that $1+2+3+....+n = \frac{n(n+1)}{2}$.	2	AP	8
	(ii) In how many arrangements of the letters of the word PHOTOGRAPH are there with exactly 5 letters between the two H's?	2	UN	8
3	(i) Prove by mathematical induction that $6^{n+2} + 7^{2n+1}$ is divisible by 43 for each positive integer 'n'.	2	AP	8
	(ii) Use mathematical induction to prove the inequality $n < 2^n$ for all positive integer n.	2	AP	8
4	(i) A box contains 6 white balls and 5 red balls. Find the number of ways four balls can be drawn from the box if (1) They can be any colour (2) Two must be white and two red (3) They must all be the same colour.	2	UN	8
	(ii) Find the number of distinct permutations that can be formed from all the letters of each word RADAR (b) UNUSUAL	2	UN	8
5	(i) In a survey of 300 students, 64 had taken a Mathematics course, 94 had taken a English course, 58 had taken a Computer course, 28 had taken both a Mathematics and Computer course,26 had taken a both a English and Mathematics course,22 had taken both a English and Mathematics course, 22 had taken both a English and a Computer course, 14 had taken all three courses. How many students were surveyed who had taken none of three courses?	2	UN	8
	(ii)How many people must you have to guarantee that atleast 9 of them will have birthdays in the same day of the week.	2	UN	8
6	(i) Solve the recurrence relation $a_n = 8a_{n-1} - 16a_{n-2}$ for $n \geq 2$, $a_0 = 16$, $a_1 = 80$.	2	AP	8
	(ii) A survey of 100 students with respect to their choice of the ice cream flavours Vanilla, Chocolate and Straberry shows that 50 students like Vanilla, 43 like Chocolate, 28 like Strawberry, 13 like Vanilla and Chocolate, 11 like Chocolate and Strawberry, 12 like Strawberry and Vanilla, and 5 like all of them. Find the number of students who like (i) Vanilla only (ii) Chocolate only (iii) Strawberry only (iv) number of students who do not like any of these.	2	AP	8
7	(i)Solve the recurrence relation $a_{n+2} = 4a_{n+1} - 4a_n$, $n \geq 0, a_0=1, a_1 = 3$.	2	AP	8
	(ii) In a survey of 100 students, it was found that 40 studied Mathematics, 64 studied Physics, 35 studied Chemistry, 1 studied all the three subjects, 25 studied Mathematics and Physics, 3 studied Mathematics and Chemistry, 20 studied Physics and Chemistry. Use the principle of inclusion and exclusion, find the number of students who studied Chemistry only and the number who studied none of these subjects?	2	AP	8
8	(i) Solve the recurrence relation $a_n = 6a_{n-1} - 9a_{n-2}, n \geq 2, a_0 = 2, \quad a_1 = 3.$	2	AP	8
	(ii) Show that the sequence $\{a_n\}$ is a solution of the recurrence relation $a_n = a_{n-1} + 2a_{n-2} + 2n - 9$ if $a_n = 3(-1)^n + 2^n - n + 2$.	2	AP	8

UNIT III

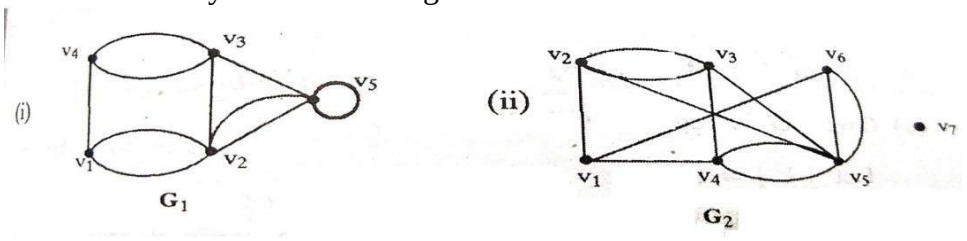
GRAPHS

Graphs and graph models – Graph terminology and special types of graphs – Matrix representation of graphs and graph isomorphism – Connectivity – Euler and Hamilton paths

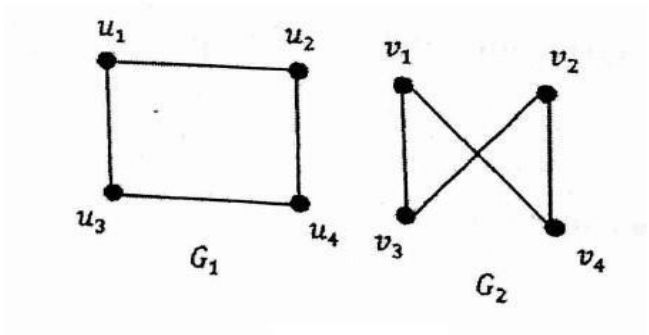
Q.No	Question	CO	BTL	Marks
	PART A			
1.	Define a complete graph with example.	3	RE	2
2.	What is meant by simple graph? Give an example.	3	RE	2
3.	Define a regular graph.	3	RE	2
4.	State the handshaking theorem.	3	RE	2
5.	Define Pseudo graph.	3	RE	2
6.	Define degree of a vertex in a graph.	3	RE	2
7.	Draw the graph represented by the given adjacency matrix $\begin{bmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{bmatrix}$	3	UN	2
8.	Show that the simple graphs with the following adjacency matrices are isomorphic. $\begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 1 \\ 1 & 1 & 0 \end{bmatrix}$, $\begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix}$.	3	UN	2
9.	Define isomorphism between graphs.	3	RE	2
10.	Define path.	3	RE	2
11.	State the necessary and sufficient conditions for the existence of an Eulerian path in connected graph.	3	RE	2
12.	Draw a complete bipartite graph of $K_{2,3}$ and $K_{3,3}$	3	RE	2
13.	Give an example of an Euler graph.	3	RE	2
14.	Give an example of a non-Eulerian graph which is Hamiltonian.	3	RE	2
15.	Give an example of a graph which is Eulerian but not Hamiltonian.	3	RE	2

PART B

		3	AP	8
1	(i) In any graph G, prove that the total number of odd-degree vertices is even.	3	AP	8
	(ii) Show that the maximum number of edges in a simple graph with n vertices is $\frac{n\,(n-1)}{2}$.	3	AP	8
2	(i) Find the number of vertices, number of edges and the degree of each vertex. Verify the handshaking theorem.			

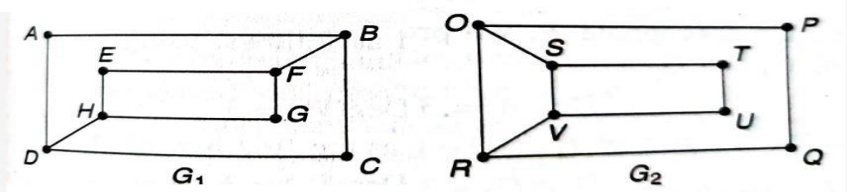


		3	UN	8
	(ii) Draw the graph with 5 vertices A,B,C,D,E such that deg(A) = 3 , B is an odd vertex , deg(C) = 2 and D and E are adjacent.			
3	(i) Determine whether the following graphs G1 and G2 are isomorphic.	3	AP	8



	(ii) State and prove Hand shaking theorem .	3	AP	8
4	(i) Draw the graph with the adjacency matrix $\begin{bmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix}$ with respect to the ordering of A, B, C,D.	3	UN	8
	(ii) Draw the directed graph corresponding to the adjacency matrix and also find the indegree and outdegree for each of the vertices. $\begin{bmatrix} 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{bmatrix}$	3	UN	8

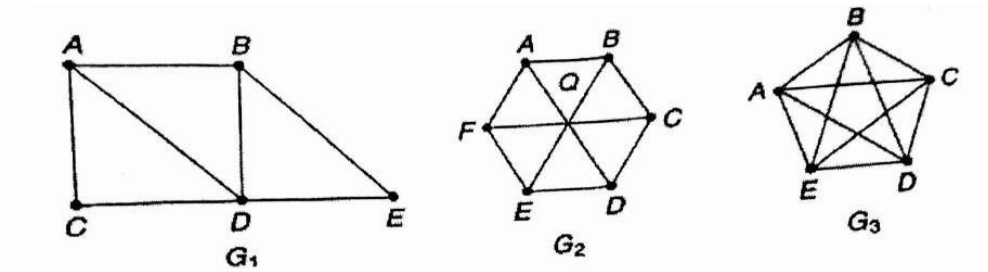
5	(i)Prove that an undirected graph has an even number of vertices of odd degree.	3	AP	8
	(ii)Examine whether the following pair of graphs are isomorphic or not.	3	AP	8



6	(i)Prove that a Simple graph with at least two vertices has at least two vertices of same degree.	3	AP	8
	(ii)The adjacency matrices of two pairs of graph as given below. Examine the isomorphism of G and H by finding a permutation matrix .	3	AP	8

$$A_G = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & 1 \\ 1 & 1 & 0 \end{pmatrix} \quad , \quad A_H = \begin{pmatrix} 0 & 1 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix}$$

7	(i)If G is a simple graph with n vertices with minimum degree $(G) \geq \frac{n}{2}$, show that G is connected.	3	AP	8
	(ii) Prove that a connected graph 'G' is an Euler graph if only if all the vertices of ' G ' are of even Degree.	3	AP	8
8	(i) Prove that maximum number of edges in a simple disconnected graph G with 'n' vertices and 'k'components is $\frac{(n-k)(n-k+1)}{2}$.	3	AP	8
	(ii) Find an Euler path or an Euler circuit, if it exists in each of the three graphs below. If it does not exist, explain why?	3	AP	8



UNIT IV

ALGEBRAIC STRUCTURES

Algebraic systems – Semi groups and monoids - Groups – Subgroups – Homomorphism's – Normal subgroup and cosets – Lagrange's theorem.

Q.No	Question	CO	BTL	Marks
	PART A			
1.	State any two properties of a group.	4	RE	2
2.	Define monoid.	4	RE	2
3.	Define a semi-group.	4	RE	2
4.	Prove that identity element is unique in a group.	4	RE	2
5.	Find the idempotent elements of G={1, -1, i, -i}under the binary operation multiplication.	4	UN	2
6.	Show that if every element in a group is its own inverse, then the group must be abelian.	4	UN	2
7.	Prove that if G is abelian group then for all $a,b \in G$ $(a*b)^2 = a^2 * b^2$.	4	UN	2
8.	Show that every cyclic group is abelian.	4	UN	2
9.	When is a group (G,*) called abelian?	4	RE	2
10.	Given $G = \{1, -1, i, -i\}$ is a group under usual multiplication and $H = \{1, -1\}$ is a subset of G. Find the index of H in G.	4	UN	2
11.	Find the left cosets of {[0], [3]} in the group $(Z_6, +_6)$	4	UN	2
12.	If 'a' is a generator of a cyclic group G then show that a^{-1} is also a generator of G.	4	UN	2
13.	Prove that every subgroup of an abelian group is a normal subgroup.	4	UN	2
14.	Show that $(Z_5, +_5)$ is a cyclic group.	4	UN	2
15.	Find all the cosets of the sub group $H = \{1, -1\}$ in $G = \{1, -1, i, -i\}$ with the operation multiplication.	4	UN	2
	PART B			
1	(i) If H_1 and H_2 are subgroups of a group (G,*) prove that $H_1 \cap H_2$ is a subgroup of (G,*).	4	AP	8

	(ii) Let (G, *) be a group, then prove that (i) For each $a \in G, (a^{-1})^{-1} = a$ (ii) For all, $a, b \in G, (a * b)^{-1} = b^{-1} * a^{-1}$ for all $a, b \in G$	4	AP	8
2	(i) Let $Z_5^* = \{[1], [2], [3], [4]\}$ be the non-zero elements of Z_5. Prove that $(Z_5^*, .5)$ is an abelian group.	4	AP	8
	(ii) Show that the set of all non-zero real numbers is an abelian group under the operation * defined by $a * b = \frac{ab}{2}$.	4	AP	8
3	(i) Show that $(Z_m, +_m)$ is an abelian group.	4	AP	8
	(ii)If H and K are subgroup of G, prove that $H \cup K$ is a subgroup of G if and only if either $H \subseteq K$ or $K \subseteq H$	4	AP	8
4	(i) If * is a binary operation on the set R of real numbers defined by a*b = a+b+2ab , a) Find (R, *) is a semigroup b) Find the identity element if it exist . c) Which element has inverse and what are they?	4	UN	8
	(ii) Prove that the necessary and sufficient condition for non-empty subset H of a group {G , *} to be subgroup is $a,b \in H \Rightarrow a * b^{-1} \in H$.	4	AP	8
5	(i)Show that every subgroup of a cyclic group is cyclic.	4	AP	8
	(ii) Show that the intersection of two normal subgroup of a group (G,*) is a normal subgroup of (G,*).	4	AP	8
6	(i) State and prove Lagrange's theorem.	4	AP	8
	(ii) Show that Kernal of a group homomorphism is a normal subgroup of the group.	4	AP	8
7	(i) If $(G,*)$ *and* (H,Δ) are two groups and $g:(G,*) \to (H,\Delta)$ is a group homomorphism then prove that kernel of g is normal subgroup of (G,*).	4	AP	8
	(ii) If <H,* > is a subgroup of <G,* > then show that <H,* > is a normal subgroup iff $a * h * a^{-1} = H, \forall a \in G$.	4	AP	8
8	State and prove the fundamental theorem on homomorphism of groups.	4	AP	16

UNIT V

LATTICES AND BOOLEAN ALGEBRA

Partial ordering – Posets – Lattices as posets – Properties of lattices - Lattices as algebraic systems– Sub lattices – Some special lattices – Boolean algebra – SubBoolean Algebra – Boolean Homomorphism.

Q.No	Question	CO	BTL	Marks
	PART A			
1.	Define a partially ordered set.	5	RE	2
2.	Draw a Hasse diagram of $D_{20} = \{1,2,4,5,10,20\}$.	5	UN	2
3.	Draw a Hasse diagram of $D_{12} = \{1,2,3,4,6,12\}$.	5	UN	2
4.	Draw the Hasse – diagram of the set of partitions of 5.	5	UN	2
5.	If A = {2,3} $\subseteq$ X = {2,3,6,12,24,36} and the relation $\leq$ is such that x $\leq$ y is x divides y, find the least element and greatest element for A.	5	UN	2
6.	Define Lattice.	5	RE	2
7.	State the distributive inequalities in Lattice.	5	RE	2
8.	In a distributive lattice prove that complement of an element, if it exists, is unique.	5	RE	2
9.	Give an example of a lattice which is a modular but not a distributive.	5	RE	2
10.	Define Sub lattices.	5	UN	2
11.	Define Boolean algebra.	5	RE	2
12.	Prove that $(a')' = a$ for all $a \in B$ where B is a Boolean Algebra.	5	UN	2
13.	If B is a Boolean algebra then prove that for $a \in B$, $a + 1 = 1$and $a.0 = 0$.	5	UN	2
14.	State De Morgan's law in Boolean Algebra.	5	RE	2
15.	In any Boolean algebra, show that a = b if and only if $a\bar{b} + \bar{a}b = 0$.	5	UN	2
	PART B			
1	(i) Let D_{24} be the set of divisors of 24 and the relation $\leq$ is $a \leq b$if $a \mid b$. Draw the Hasse diagram for the poset $(D_{24}, \mid)$.	5	UN	8
	(ii) Draw the Hasse – diagram of the Lattice L of all subsets of {a, b, c} under intersection and union.	5	UN	8

2	(i) Let $(L, \leq)$ be a lattice. For any $a, b, c \in L$ the following inequalities hold. (i) $a \oplus (b * c) \leq (a \oplus b) * (a \oplus c)$ (ii) $a * (b \oplus c) \geq (a * b) \oplus (a * c)$.	5	AP	8
	(ii) State and Prove De Morgan's law in Lattice.	5	AP	8
3	(i) Prove that every chain is a distributive lattice.	5	AP	8
	(ii) In a distributive lattice (L, *, $\oplus$) if for any $a, b, c \in L$, $a * b = a * c$ and $a \oplus b = a \oplus c$ then $b = c$.	5	AP	8
4	(i)Prove that every distributive lattice is modular. Is the converse true? Justify your claim.	5	AN	8
	(ii) In a distributive lattice prove that $a * b = a * c$ *and* $a \oplus b = a \oplus c$ *imply* $b = c$.	5	AP	8
5	(i) Let $(L, \leq)$ be a lattice. For any $a, b, c \in L$ the following properties called isotonicity hold. If $b \leq c$ then (i) $a * b \leq a * c$ (ii) $a \oplus b \leq a \oplus c$.	5	AP	8
	(ii)In a lattice show that $a \leq b \Rightarrow a * b = a$.	5	AP	8
6	(i)Show that if L is a distributive lattice then for all $a, b, c \in L$, $(a * b) \oplus (b * c) \oplus (c * a) = (a \oplus b) * (b \oplus c) * (c \oplus a)$	5	AP	8
	(ii) Show that in a distributive and complemented lattice $a \leq b \Leftrightarrow a * b' = 0 \Leftrightarrow a' \oplus b = 1 \Leftrightarrow b' \leq a'$	5	AP	8
7	(i) In a Boolean Algebra show that $ab' + a'b = 0$ if and only if $a = b$.	5	AP	8
	(ii) In a Boolean Algebra, prove that the following statements are equivalent. (i) $a + b = b$ (ii) $a.b = a$ (iii) $a' + b = 1$ (iv) $a.b' = 0$.	5	AP	8
8	(i) State and Prove De Morgan's law in Boolean Algebra.	5	AP	8
	(ii)In Boolean algebra, prove that $(a \wedge b)' = a' \vee b'$ for all a,b∈L.	5	AP	8

----- END ----