AMAL JYOTHI COLLEGE OF ENGINEERING
KANJIRAPPALLY

MCA-INTEGRATED

MCAINT2022-27-S6 : 20INMCA308-Design & Analysis of Algorithms-Assignment 1

QP Code: 20INMCA308/2020/A/5 **Max.Marks :6**

Q.No	Questions	Marks	CO	BL	PI
1	I. Solve the recurrence relation using Iteration Method a. T(n)= 2T(n/2) +n b. T(n)= c+(n-1) c. T(n)= T(n/2) +1 d. T(n)= 8T(n/2) +n^2 (T (1) =1) e. T(n)=T(n-1) +n II. Solve the recurrence relation using Recurrence Tree Method a. T(n)= 1 n=1 1. T(n/2) +n n>1 b. T(n)= n=1 1. 2T(n/2) +n n>1 c. T(n)=T(n-1) +n d. T(n) = T(n/10) + T(9n/10) + n e. T(n) = T(n/5) + T(4n/5) + n III. Apply the Master's Theorem to determine the time complexity of the recurrence a. T(n)= 3T(n/2) +n^2 b. T(n)= 4T(n/2) +n^2 c. T(n)= 9T(n/3) +n d. T(n)= 2T(n/2) +cn e. T(n)= 16T (n/4) + n	3	CO1	L6	1.1.1,1.1.2,1.3.1, 2.1.3,2.2.1,2.2.5, 2.3.1
2	I. Given a sorted array **arr[] = {2, 5, 8, 12, 16, 23, 38, 56, 72, 91}** and a target value **23**, implement a Binary Search algorithm to determine whether the target exists in the array. II. Given a sorted array **a[] = {2, 3, 7, 7, 11, 15, 25},** implement a Binary Search algorithm to find the target element **11**. III. Given an unsorted array **a[] = {38, 27, 43, 3, 9, 82, 10},** implement the Merge Sort algorithm to sort the array in ascending order. Perform a step-by-step dry run, explaining	3	CO2	L6	1.1.1,1.1.2, 1.3.1,1.4.1,2.1.1, 2.1.2,2.2.1,2.2.3, 2.2.5

Q.No	Questions	Marks	CO	BL	PI

and visualizing how the array is recursively divided and merged at each stage of the Merge Sort algorithm.

IV. Formulate the Merge Sort algorithm for the given array **a[] = {12, 11, 13, 5, 6, 7}** and sort it in ascending order. Count and display the number of comparisons made during the sorting process.

V. Given an unsorted array a[] = {29, 10, 14, 37, 13, 20}, implement the Quick Sort algorithm to sort the array in ascending order, and demonstrate a step-by-step dry run of the algorithm, showing the partitioning process at each step.

VI. Given the unsorted array a[] = {44, 33, 55, 22, 88, 77, 11, 99}, implement the Quick Sort algorithm. Demonstrate the sorting process, showing the choice of pivot and partitioning at each recursion step.

VII. Analyse and implement a divide and conquer algorithm to find the maximum and minimum values in the array [3, 1, 4, 1, 5, 9, 2].

VIII. Design a recursive divide and conquer approach to find the minimum and maximum elements in a given array of integers [12, 7, 5, 10, 8]

IX. A thief enters a store with a knapsack that can carry a maximum weight of **50 kg**. There are **three items** available, each with a specific weight and value. The thief can take **fractional** parts of an item. Find the maximum value the thief can carry using the **Greedy Algorithm**.

X. Item	**XI. Weight**	**XII. Value**
XIII. 1	XIV. 10	XV. 60
XVI. 2	XVII. 20	XVIII. 100
XIX. 3	XX. 30	XXI. 120

XXII. A hiker has a backpack with a **capacity of 60 kg** and wants to carry the most valuable items. There are **four items**, each with a given weight and value. The hiker can take **fractional** parts of any item. Find the **maximum value** that can be carried using the **Greedy Algorithm**.

Item	**Weight**	**Value**
1	20	100
2	10	60
3	30	120
4	40	240

CO1: Implement design principles and analyze the asymptotic performance of algorithms.

CO2: Derive and solve recurrences describing the performance of divide-and-conquer algorithms and greedy algorithms

CO(s) contribution for PO/PSO Attainment from Assignment 1

Question	COs Mark & Mapped PO(s)/PSO(s)[Strength] *3.Substantial, 2.Moderate, 1.Slight*	Total Marks per CO	40% per CO(s)	CO contribution to calculate PO/PSO attainment(%)

1 a)	CO1[**3**]=>**PO1** (3), **PO2** (2), **PO3** (3), **PO4** (2), **PO5** (1), **PO7** (1), **PO8** (1),**PSO1** (2), **PSO2** (3)	CO1=>3
		CO2=>3
			CO1=>1.2
			CO2=>1.2	

2 a)	CO2[**3**]=>**PO1** (2), **PO2** (3), **PO3** (3), **PO4** (2), **PO5** (1), **PO8** (1),**PSO1** (3), **PSO2** (2), **PSO3** (1)			

Rubrics used for the assessment- MCAINT2022-27-S6 : 20INMCA308-Design & Analysis of Algorithms-Assignment 1

Bloom's Level wise Marks Distribution

Blooms Taxonomy Level		**Percentage**
L6	Creating	100

Course Outcome wise Marks Distribution

COs	**Percentage**
CO1	50
CO2	50