## Code: 20CS3403, 20IT3403, 20AM3403, 20DS3403

## II B.Tech - II Semester – Regular / Supplementary Examinations MAY - 2024

## DESIGN AND ANALYSIS OF ALGORITHMS (Common for CSE, IT, AIML, DS)

Duration:	3	hours
-----------	---	-------

Note: 1. This paper contains questions from 5 units of Syllabus. Each unit carries 14 marks and have an internal choice of Questions.

2. All parts of Question must be answered in one place.

BL – Blooms Level

CO – Course Outcome

Max. Marks: 70

Image:				BL	СО	Max.	
1a)Discuss the basic principles of brute forceL2CO16 M1a)Discuss the basic principles of brute forceL2CO16 M1technique and provide examples of problemswhere it is appropriate.11b)Define asymptotic notations (Big O, Omega, Theta). Provide examples of algorithms and express their time complexities using these notations.L2CO18 MOR2a)Define exhaustive search as a specific application of the brute force technique. Highlight its strengths and limitations. Provide a real-world example where exhaustive search might be employed.L3CO28 M				DL		Marks	
technique and provide examples of problems where it is appropriate. Image: technique and provide examples of problems where it is appropriate.   b) Define asymptotic notations (Big O, Omega, Theta). Provide examples of algorithms and express their time complexities using these notations. Image: L2 CO1 8 M   2 a) Define exhaustive search as a specific Highlight its strengths and limitations. L3 CO2 8 M   Provide a real-world example where exhaustive search might be employed. Image: L3 CO2 8 M		UNIT-I					
where it is appropriate. where it is appropriate.   b) Define asymptotic notations (Big O, Omega, L2 CO1 8 M Theta). Provide examples of algorithms and express their time complexities using these notations. L2 CO1 8 M	1	a)	Discuss the basic principles of brute force	L2	CO1	6 M	
b) Define asymptotic notations (Big O, Omega, Theta). Provide examples of algorithms and express their time complexities using these notations. L2 CO1 8 M   OR   2 a) Define exhaustive search as a specific application of the brute force technique. Highlight its strengths and limitations. L3 CO2 8 M   Provide a real-world example where exhaustive search might be employed. L3 CO2 8 M			technique and provide examples of problems				
Theta). Provide examples of algorithms and express their time complexities using these notations. Image: Complexities of algorithms and express their time complexities using these notations.   2 a) Define exhaustive search as a specific the brute force technique. Highlight its strengths and limitations. Provide a real-world example where exhaustive search might be employed. L3 CO2 8 M			where it is appropriate.				
express their time complexities using these notations. Image: Complexities using these notations.   2 a) Define exhaustive search as a specific technique.   2 a) Define exhaustive search as a specific technique.   Highlight its strengths and limitations. Image: Complexities technique technique technique.   Provide a real-world example where exhaustive search might be employed. Image: Complexities technique te		b)	Define asymptotic notations (Big O, Omega,	L2	CO1	8 M	
Image: Instant of the image: Image			Theta). Provide examples of algorithms and				
OR   2 a) Define exhaustive search as a specific L3 CO2 8 M application of the brute force technique. Highlight its strengths and limitations. Highlight its strengths and limitations. Provide a real-world example where exhaustive search might be employed. L3 CO2 8 M			express their time complexities using these				
2 a) Define exhaustive search as a specific L3 CO2 8 M application of the brute force technique. Highlight its strengths and limitations. Provide a real-world example where exhaustive search might be employed. CO2 8 M			notations.				
application of the brute force technique. Highlight its strengths and limitations. Provide a real-world example where exhaustive search might be employed.	OR						
Highlight its strengths and limitations.Provide a real-world example whereexhaustive search might be employed.	2	a)	Define exhaustive search as a specific	L3	CO2	8 M	
Provide a real-world example where exhaustive search might be employed.			application of the brute force technique.				
exhaustive search might be employed.			Highlight its strengths and limitations.				
			Provide a real-world example where				
			exhaustive search might be employed.				
b) Explain the significance of time complexity L2 CO1 6 M		b)	Explain the significance of time complexity	L2	CO1	6 M	
with an appropriate example.			with an appropriate example.				

UNIT-II							
	\ \		10	GOO	0.14		
3	a)	Illustrate the Quick Sort algorithm with an	L3	CO2	8 M		
		appropriate example.					
	b)	In the Quick Sort algorithm, the choice of	L3	CO2	6 M		
		the pivot element determines the					
		performance of the algorithm. Justify this					
		statement with an appropriate example.					
		OR		<b>I I</b>			
4	a)	Define Binary Search and explain how it	L3	CO3	8 M		
		works on a sorted array? Discuss its time					
		complexity and scenarios where it					
		outperforms other searching algorithms.					
	b)	Write an algorithm to find the maximum and	L3	CO3	6 M		
		minimum element in a single pass. Derive					
		its time complexity.					
UNIT-III							
5	a)	Find Huffman code for each symbol in	L3	CO3	7 M		
		following text :					
		ABCCDEBABFFBACBEBDFAAAABC					
		DEEDCCBFEBFCAE					
	b)	Apply the Dijkstra's algorithm on the	L3	CO3	7 M		
		following network to obtain the shortest path					
		from the node A to all the other nodes.					

		OR				
6	a)	Consider the knapsack problem where weights and profits are given as follows: Weights: {3, 4, 6, 5} Profits: {2, 3, 1, 4} Assuming that the total weight of the knapsack is 8 kg, write an algorithm that will apply the greedy method to maximize the profit earned. Illustrate your algorithm using the above example.	L3	CO3	8 M	
	b)	Illustrate the Kruskal's algorithm to obtain a minimum spanning tree for the following graph. 1100 + 100 +	L3	CO3	6 M	
	UNIT-IV					
7	a)	Explain what is Dynamic Programming with a suitable example?	L3	CO2 CO3	7 M	

	b)	Write a recursive algorithm for finding the	L3	CO2	7 M		
		n <sup>th</sup> term in the Fibonacci series. Derive its		CO3			
		time complexity. Explain how dynamic					
		programming can be used to reduce the time					
		complexity of the recursive algorithm.					
		OR	L	1 1			
8	a)	Explain the dynamic programming principle	L4	CO2	7 M		
		behind the 0/1 Knapsack problem with an		CO3			
		example.					
	b)	How does the dynamic programming	L4	CO2	7 M		
		principle work in Optimal Binary Search		CO3			
		trees? Illustrate with an example.					
		UNIT-V					
9	a)	Write a backtracking algorithm for the n-	L3	CO4	7 M		
		Queens problem and illustrate the same for					
		n=4.					
	b)	Explain the Branch-and-Bound technique	L3	CO4	7 M		
		with an appropriate example.					
OR							
10	a)	Apply backtracking technique to solve the	L3	CO3	6 M		
		following instance of the sum of subsets					
		problem S={3, 8, 9, 10, 12, 13} and m=32					
		(where S is set of elements and m is sum of					
		elements)					
	b)	What are the different complexity classes?	L4	CO4	8 M		
		Explain each with an example.					