Reg. No. | | | | | | | | | | | | | |

B.E. / B.TECH. DEGREE EXAMINATIONS, MAY 2023

Fourth Semester

CS18405 – DESIGN AND ANALYSIS OF ALGORITHM

(Computer Science and Engineering)

(Regulation 2018/2018A)

TIME: 3 HOURS **MAX. MARKS: 100**

COURSE OUTCOMES	STATEMENT	RBT LEVEL
CO 1	The students will be able to analyze the running time of algorithms using asymptotic Analysis.	2
CO 2	The student will be able to describe the divide-and-conquer techniques and analyze the running time of algorithms synthesizing those paradigms.	2
CO 3	The students will be able to describe the dynamic programming and greedy paradigms and analyze the running time of the algorithm using those techniques.	3
CO 4	The students will be able to employ linear programming and computational geometry methods to solve engineering problems.	4
CO 5	The student will be able to describe the non-deterministic polynomial algorithms.	5

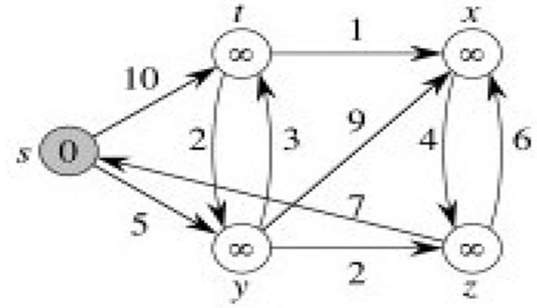
PART- A (10 x 2 = 20 Marks)

(Answer all Questions)

		CO	RBT LEVEL
1.	Explain the role of algorithms in computing with example.	1	3
2.	Identify the parameters to measure the efficiency of an algorithm.	1	3
3.	State the difference between divide-and-conquer and dynamic programming strategies.	2	2
4.	State the regularity condition used in Master's theorem.	2	2
5.	How many numbers of binary search trees can be formed using 'n' keys?	3	2
6.	State the limitation of using Floyd-Warshall Algorithm for finding all pairs shortest path in a graph.	3	2
7.	State the rules for converting linear programs into standard form.	4	2
8.	How cross products are used for determining whether two lines segments intersect?	4	2
9.	Examine the polynomial time reducibility and infer its significance in NP-completeness.	5	4
10.	Differentiate feasible solution and optimal solution with suitable example.	5	3

PART- B (5 x 14 = 70 Marks)

		Marks	CO	RBT LEVEL
11. (a)	Illustrate the Asymptotic Notations and its purpose in Algorithm analysis with suitable diagrams and examples.	(14)	1	2
	(OR)			
(b)	Analyze Insertion sort with the proper algorithm. Sort the elements 56,71,88,15,23,45,12,33,44 using insertion sort. Derive its Complexity.	(14)	1	2
12. (a)	Given a one-dimensional array that may contain both positive and negative integer, find the sum of contiguous sub-array of numbers which has the largest sum using maximum-sub-array-sum algorithm. Given the arrays: {-3, -5, 1, -2, -1, 11, -15, 6}. Find the maximum sub-array sum. Derive the complexity of the algorithm.	(14)	2	3
	(OR)			
(b)	Write an algorithm to find the closest pair of points using divide and conquer and explain it with an example. Derive the worst case and average case time complexity.	(14)	2	3
13. (a)	Explain the algorithm to find the longest common subsequence for any 2 sequences. Consider the strings X={A,B,C,B,D,A,B} and Y={B,D,C,A,B,A}. Calculate the length of the longest common subsequence. Find the longest common subsequence for the above using dynamic programming strategy .	(14)	3	3
	(OR)			
(b)	Explain how greedy approach is used in Dijkstra's algorithm for finding the single-source shortest paths for the given graph.	(14)	3	3



14. (a)	Solve the problem using simplex method in linear programming problem. Maximize $2x_1-3x_2+3x_3$ Subject to $X_1+X_2-X_3 \leq 7$ $-X_1-X_2+X_3 \leq -7$ $X_1-2X_2+2X_3 \leq 4$ $X_1,X_2,X_3 \geq 0$	(14)	4	3
	(OR)			
(b)	Examine Graham's scan and Jarvi's march algorithms for the construction of convex hull.	(14)	4	3

Q. Code:182388

15. (a)	**(i)**	Prove that Circuit satisfiability problem is NP-complete.	**(7)**	**5**	**4**
	(ii)	Prove that Formula satisfiability problem is NP-complete.	**(7)**	**5**	**4**
		(OR)			
(b)	**(i)**	Explain Clique Problem. Prove that Clique problem is NP-complete.	**(7)**	**5**	**4**
	(ii)	Explain Vertex cover problem. Prove that Vertex problem is NP-complete.	**(7)**	**5**	**4**

PART- C (1 x 10 = 10 Marks)

(Q.No.16 is compulsory)

		Marks	CO	RBT LEVEL
16.	Explain optimal binary search tree algorithm in detail. Derive its time complexity. Given the values of pi's and qi's in Table1, construct the optimal binary search tree using the above keys whose expected search cost is minimum.	**(10)**	**3**	**5**

i	0	1	2	3	4	5
P_i	-----	0.15	0.10	0.05	0.10	0.20
q_i	0.05	0.10	0.05	0.05	0.05	0.10

****