2- Office Copies

Roll Number:____________________

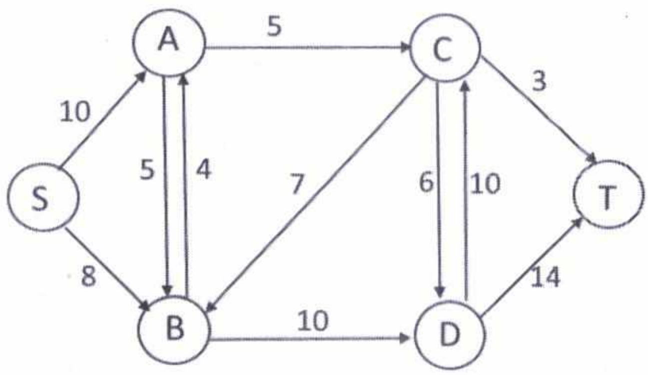
Thapar Institute of Engineering & Technology, Patiala

Department of Computer Science and Engineering

AUXILIARY EXAMINATION

B. E. (CSE & COE)	Course Code: UCS415 Course Name: Design and Analysis of Algorithms
Date: August 17, 2022	Time: 5:30 PM – 7:30 PM
Duration: 2 Hours, M. Marks: 50	Name of Faculty: Dr. Yashwant Singh Patel

Note: Attempt all questions in a proper sequence with justification.
Assume missing data, if any, suitably.

Q. 1(a)	Solve the following recurrence relation using recursive tree method. $$T(n) = \left\{\begin{matrix} T\left(\frac{n}{3}\right) + T\left(\frac{2n}{3}\right) + n & n > 1 \\ 1 & n = 1 \end{matrix}\right\}$$	(5)
Q. 1(b)	Explain the difference between deterministic and non-deterministic algorithm? Also, explain the relationship between P, NP, NP-complete and NP hard class problems with example of each class.	(1+4)
Q. 2(a)	Execute the Ford-Fulkerson algorithm to find the maximum flow for a graph (shown in **Fig. 1**) from source **S** to destination **T**. Show all the intermediate steps of algorithm. Also explain: (a) Augmented path (b) Residual graph (c) Minimum cut (d) Residual capacity.  **Fig. 1**	(8)
Q. 2(b)	Discuss the time complexity of the Hierholzer's algorithm to determine Euler circuit ?	(2)
Q. 3	Use dynamic programming to fully parenthesize the product of five matrices, i.e. **A[4 × 10], B[10 × 3], C[3 × 12], D[12 × 20], E[20 × 7]** such that the number of scalar multiplications gets minimized. Show each and every step.	(10)
Q. 4(a)	Discuss the recursive definition/equation employed in dynamic programming solution for 0/1 knapsack problem. Utilize it to maximize the profit for a knapsack having a capacity of **W = 7** using four items. The respective weight and profit values of the four items are **w[ ]: (1, 3, 4, 5)** and **p[ ]: (1, 4, 5, 7)**.	(1+5)
Q. 4(b)	Consider a modification to the activity-selection problem in which each activity a_i has, in addition to a start and finish time, a value v_i. The objective is no longer to maximize the number of activities scheduled, but instead to maximize the total value of the activities scheduled. That is, we wish to choose a set A of compatible activities such that $\sum_{a_k \in A} v_k$ is maximized. Discuss whether a greedy algorithm is possible. If yes, present such an algorithm, and if no, argue why not.	(4)

Q. 5(a)	For string matching, working modulo **q = 13**, how many spurious hits does the Rabin-Karp matcher encounter in the Text **T** = 3141592653589793 When looking for the pattern **P =26**? Explain your answer with proper implementation of Rabin-Karp algorithm.	(7)
Q. 5(b)	What is backtracking ? Explain 8 queen's problem and how can we solve it using backtracking ?	(3)