National Institute of Technology Tiruchirappalli
Department of Computer Science and Engineering
Design and Analysis of Algorithms (CSPC42)
Final Assessment

Date: May 09, 2023

Duration: 3 Hours | II-B.Tech CSE-A, V Semester | Max marks: 40

1. Find the Time complexity of the following recurrences **(4 M)(CO1)**
   (a) $T(n) = 2T(n/2) + nlogn$ (b) $T(n) = 16T(n/4) + n$

2. Suppose the symbols a, b, c, d, e occur with frequencies 1/2, 1/4, 1/8, 1/16, 1/32, respectively. (a) What is the Huffman encoding of each alphabet? (b) If this encoding is applied to a file consisting of $1,000,000$ characters with the given frequencies, what is the length of the encoded file in bits? **(4M)(CO2)**

3. Consider an undirected graph $G(V, E)$, where every edge $e \in E$ has a cost c_e. Assume that all edge costs are positive and distinct. Let T be a minimum spanning tree of G and P the shortest path from the vertex v_i to the vertex v_j. Suppose that the cost of every edge e of G increases by one and becomes $c_e + 1$. Call this new graph as G'. Then justify your answers with the examples for the following questions.
   (a) Is T will be the minimum spanning tree of new graph G'.
   (b) Is P will be the shortest path from vertex v_i to vertex v_j in Graph G'.
   (4M)(CO2)

4. Suppose we have a sequence of matrices $[A_1, A_2, A_3, A_4]$, where A_1 has dimensions 2x3, A_2 has dimensions 3x6, A_3 has dimensions 6x4, and A_4 has dimensions 4x5. What is the optimal order and the minimum number of scalar multiplications required to compute the product $A_1A_2A_3A_4$? Show the table using dynamic programming. Write the recurrence relation. Discuss its time complexity. How many total numbers of recursive calls and unique recursive calls are involved in the given example? **(4M)(CO1, CO2)**

5. Consider a variation of the $0-1$ Knapsack problem where we have two knapsacks, with integer capacities W_1 and W_2. As usual, we are given n items with positive values and positive integer weights. We want to pick subsets S_1 and S_2 with maximum total value (i.e., $\sum_{i \in S_1} v_i + \sum_{i \in S_2} v_i$) such that the total weights of S_1 and S_2 are at most W_1 and W_2 respectively. Assume that every item fits in either knapsack (i.e., $w_i \leq min\{W_1, W_2\}$ for every item i). Consider the following two algorithmic approaches.
   Algo. 1 : Use the $0-1$ Knapsack problem algorithm to pick a max-value feasible solution S_1 for the first knapsack, and then run it again on the remaining items to pick a max-value feasible solution S_2 for the second knapsack.
   Algo. 2 : Use the $0-1$ Knapsack problem algorithm to pick a max-value feasible solution for a knapsack with capacity $W_1 + W_2$ and then split the chosen items into two sets $S_1 + S_2$ that have size at most W_1 and W_2 respectively. Which of the following statements are true, and justify your answers with examples?
   (4M)(CO2)

   (a) Algorithm (1) is guaranteed to produce an optimal feasible solution to the original problem provided $W_1 == W_2$.

(b) Algorithm (1) is guaranteed to produce an optimal feasible solution to the original problem but algorithm (2) is not.

(c) Algorithm (2) is guaranteed to produce an optimal feasible solution to the original problem but algorithm (1) is not.

(d) Neither algorithm is guaranteed to produce an optimal feasible solution to the original problem.

6. For the problem in Question 5, design a dynamic programming solution (recurrence relation and DP algorithm). **(4M)(CO2)**

7. A subsequence is palindromic if it is the same whether read left to right or right to left. For instance, the sequence
$A, C, G, T, G, T, C, A, A, A, A, T, C, G$
has many palindromic subsequences, including A, C, G, C, A and A, A, A, A (on the other hand, the subsequence A, C, T is not palindromic). Devise an algorithm that takes a sequence x[1...n] and returns the (length of the) longest palindromic subsequence. Its running time should be $O(n^2)$. **(4M)(CO1, CO2)**

8. Determine the cost and structure of an optimal binary search tree for a set of $n = 7$ keys with the following probabilities: **(4M)(CO2)**

i	0	1	2	3	4	5	6	7
p_i		0.04	0.06	0.08	0.02	0.10	0.12	0.14
q_j	0.06	0.06	0.06	0.06	0.05	0.05	0.05	0.05

9. What are the differences between Las Vegas and Monte Carlo algorithms? Give examples for both algorithms and explain. **(2M)(CO3)**

10. For problem P, if we are given an input I and a possible answer A, and we find a way to verify whether or not A really is a valid answer to P given I, then what kind of problem is P? Explain **(2M)(CO4)**

11. How would you use the idea of reducibility to prove that problem A is NP-Complete, if you know that: (i) Problem B is NP-Complete (ii) Problem C is NP (iii) Problem D is P. Write down which of these 3 pieces of information might be useful in showing that A is NP-Complete **(2M)(CO4)**

12. Given the current state of knowledge, which of the following statements cannot be true? Justify your answer. **(2M)(CO4)**

(a) Some NP-complete problems are polynomial-time solvable, and some NP-complete problems are not polynomial-time solvable.

(b) There is an NP-complete problem that is polynomial-time solvable.

Best Wishes