30

ISLAMIC UNIVERSITY OF TECHNOLOGY (IUT)

ORGANISATION OF ISLAMIC COOPERATION (OIC)

Department of Computer Science and Engineering (CSE)

MID SEMESTER EXAMINATION SUMMER SEMESTER, 2022-2023

DURATION: 1 HOUR 30 MINUTES FULL MARKS: 50

CSE 4809: Algorithm Engineering

Programmable calculators are not allowed. Do not write anything on the question paper.

Answer **all 3 (three)** questions. Figures in the right margin indicate full marks of questions with corresponding COs and POs in parentheses.

---

1. a) Answer the following questions in brief: 2×4 (CO2) (PO1)

   i. Why do we look for asymptotic bound of any algorithm?

   ii. Mention the characteristic(s) of algorithms that have a complexity of $\boldsymbol{lg(n)}$.

   iii. How does divide and conquer technique help merge sort algorithm?

   iv. Why does quick sort algorithm just have division (i.e. partition) but does not have any merging?

   b) Write down the rationale of master method for its all three cases for determining the complexity of recursive algorithms. 5 (CO1) (PO1)

   c) Given a regular recursion $T(n) = aT(n/b) + f(n)$
   If $f(n) = O(n^{log_b a - \epsilon})$, prove that $T(n) = (n^{log_b a})$. 5 (CO1) (PO1)

2. a) Answer the following question in a single sentence: 1×5 (CO2) (PO1)

   i. How does dynamic programming save computation of a combinatorial optimization problem?

   ii. Can dynamic programming algorithm be used in a path finding problem where the problem is to find the list of paths within a ratio of the optimal paths?

   iii. What is Catalan number? How is it related to optimal parenthesis problem?

   iv. What is optimal substructure equation?

   v. Why longest simple path finding problem does not have an optimal substructure property?

   b) Write down the optimal substructure equation for Dynamic Time Warping (DTW) after briefly describing the optimization problem the algorithm attempts to solve. Comment on the complexity of the DTW algorithm. Briefly describe two ways to improve the time complexity of the algorithm. 6 (CO2) (PO1)

   c) Given a grid of $(m \times n)$ dimension containing cells filled with reward (positive or negative), the reward of an area will be sum of rewards of all the cells withing the area. 5 (CO2) (PO2)

   If a problem is defined as to find the maximum possible reward from a minimum square:

   i. Can you use Dynamic Programming algorithm for the problem defined above? If yes, what will be the optimal substructure for the problem?

   ii. If Dynamic Programming cannot be used or the benefit of using the algorithm is hindered for some reason, explain why it might be the case.

3. a) Write an algorithm to find the median of a data array in linear time. Prove the linear complexity of the algorithm in the worst case. 3 + 4 (CO4) (PO1)

b) What is quasi-polynomial time algorithm? Outline the solution for 0-1 Knapsack problem and justify its complexity as quasi-polynomial. Mention at least one more algorithm that is also quasi-polynomial. 5 (CO1) (PO1)

c) Prove that the complexity of quicksort algorithm is $(nlgn)$. 4 (CO1) (PO1)