United International University

Department of Computer Science and Engineering

CSE 2217: Data Structure and Algorithms II

Mid Examination: Summer 2026, Duration: 1 Hour 30 Minutes, Total Marks: 30

Answer all questions. Marks are indicated on the right side of each question.

Any examinee found adopting unfair means will be expelled from the trimester/program as per UIU disciplinary rules.

---

1. a) An algorithm takes approximately n operations for any input data set of size n. Represent the time complexity of the algorithm using appropriate asymptotic notation with a proper explanation. [1]

b) The best case time complexity, $T_{best} = log_2 100$ and worst case complexity, $T_{worst} = \frac{n^2}{100} + 5log_2 n^3$.
Represent the best and worst case time complexities using asymptotic notation. [1]

c) Prove that $2n^3 + 4n^2 + 5n = \Omega(n^3)$ using the definition of Big-Omega notation. [2]

2. a) If every element in an array is negative, what is the time complexity of finding the maximum sum subarray? Explain your answer briefly. [2]

b) Given the array A = [6, -3, 5, -8], use the Divide and Conquer approach to find the maximum sum subarray. Show the step-by-step computation and state the maximum sum and the corresponding subarray. [3]

c) A modified version of Merge Sort, called Pairwise-Merge Sort, splits the array into 4 equal parts and recursively sorts each part. Instead of merging all 4 sorted parts together in a single pass, it merges them pairwise: Part 1 is merged with Part 2, and Part 3 is merged with Part 4 (each merge takes O(n) time), and then the two resulting sorted halves are merged together in one final O(n) merge.

i. Write a recurrence relation for the Pairwise-Merge Sort algorithm. [1]

ii. Solve this recurrence relation using any method of your choice. [2]

iii. What is the time complexity of standard Merge Sort (splitting into 2 halves, one merge per level)? Does the time complexity of Pairwise-Merge Sort differ from it? Explain briefly. [2]

3. a) A relief team has a truck with a capacity of 30 kg and can take fractions of supplies. Five supplies have weights 12, 10, 8, 6, 5 kg and corresponding values 72, 70, 56, 48, 35.

i. Apply the greedy approach to determine the maximum value the truck can carry. Show the selection order and justify the greedy criterion. [4]

ii. A student argues that the item with the highest value should always be selected first. Determine whether this strategy produces the optimal solution for this instance and explain your reasoning. [2]

b) In Huffman Coding, explain why the two symbols with the lowest frequencies are combined first. What property of the resulting binary tree makes this greedy choice suitable for minimizing the total encoded length? [2]

4. a) A 0/1 Knapsack Dynamic Programming solution depends on three parameters: the number of items, the knapsack capacity, and the maximum number of items that can be selected. What will be the size of the table and its space complexity? [2]

b) A ride-sharing driver has **7 liters** of fuel available before ending the shift. Four ride requests are currently available. Each ride requires a certain amount of fuel to complete and provides a fixed fare. Each ride request can be accepted at most once. The driver wants to choose the ride requests that maximize the total fare earned without using more than 7 liters of fuel.

Ride Request	Fuel Required (liters)	Fare ($)
A	2	5
B	3	8
C	4	9
D	2	6

i. Formulate the scenario as a Dynamic Programming (DP) problem and write the recurrence relation for it. [2]

ii. Using the bottom-up Dynamic Programming approach, determine which ride requests should be accepted and state the maximum total fare. Show all necessary calculations. [4]