[This question paper contains 4 printed pages.]

**Your Roll No.....**

**L**

| | | |
|---|---|---|
| **Sr. No. of Question Paper** | : | **1806** |
| Unique Paper Code | : | 2342012401 |
| Name of the Paper | : | Design and Analysis of Algorithms |
| Name of the Course | : | **B.Sc. (H) Computer Science** |
| Semester | : | IV |

Duration : 3 Hours

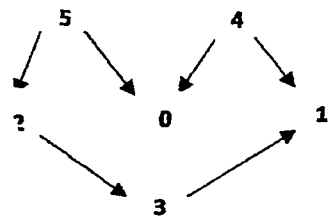
Maximum Marks : 90

**Instructions for Candidates**

1. Write your Roll No. on the top immediately on receipt of this question paper.
2. **Section A** is Compulsory.
3. Attempt any **four** questions from **Section B**.
4. Parts of a question must be answered together.
5. Write all the intermediate steps, wherever required to get full credits.

## Section A

Q1. (a) List all the topological orderings of the Directed Acyclic Graph (DAG) G given below.




3

(b) Consider the following procedure SUM-ARRAY which computes the sum of n numbers in array A[1:n].

```
SUM-ARRAY (A, n)
1  sum = 0
2  for i = 1 to n
3      sum = sum + A[i]
4  return sum
```

State a loop invariant for this procedure, and use its initialization, maintenance, and termination properties to show that the SUM-ARRAY procedure returns the sum of the numbers in A[1:n]. 3

(c) Quicksort partitions an array into two subarrays based on a pivot element. The running time depends on how balanced the partition is. Suppose at each step, the pivot divides the array into two parts in the ratio 1:3. Formulate the recurrence relation for the running time of Quicksort in this case. Solve the recurrence and determine the time complexity. 3

(d) Consider a hash table of size 11 with the hash function h(k) = k mod 11. Insert the keys 21, 32, 43, 54, 65, and 76 into the table. Handle any collisions using linear probing. 3

(e) A matrix multiplication technique reduces the number of recursive multiplications by dividing matrices into submatrices and combining results using fewer multiplications than the conventional method. Identify the algorithm and compute the product of the following matrices using the same algorithm.

Given two matrices:

$$A = \begin{bmatrix} 1 & 7 \\ 3 & 5 \end{bmatrix} \qquad B = \begin{bmatrix} 6 & 4 \\ 2 & 8 \end{bmatrix}$$ 4

(f) Consider the interval scheduling problem wherein we are given a resource and set of requests each having a start time and finish time. The goal is to maximize the number of requests scheduled. Show that the greedy strategy of selecting the request with the fewest incompatible requests does not always yield an optimal solution for this problem. 4

(g) Let $S \subset V$ be a subset of vertices in a connected, weighted, undirected graph, and let V−S denote the remaining vertices. Suppose an edge (u,v) has the minimum weight among all edges connecting S and V−S. Determine whether this edge should be included in a Minimum Spanning Tree (MST). Justify your answer. 5

(h) A delivery company has packages with weights {2, 5, 7, 3, 9} and truck capacity 15 kg. Design a greedy algorithm to maximize the number of packages loaded. Explain its time complexity. 5

## Section B

Q2. (a) Which sorting technique is most efficient if the range of input data is not significantly greater than the number of elements to be sorted and runs in $\Theta(n+k)$ time, where n is the size of the array and k is the range of the non-negative key values. Sort the array A= {5, 9, 0, 1, 5, 4, 2, 0, 6, 9} using the same algorithm. 7

(b) Consider the following two algorithms used for sorting a list of numbers:

- Algorithm A preserves the relative order of equal elements and follows a divide-and-conquer approach, using extra memory.
- Algorithm B uses a tree-based structure, requires minimal extra space, but does not preserve relative order.

i. Identify Algorithm A and Algorithm B.

ii. Identify the property that ensures preservation of the relative order of equal elements.

iii. State the time complexity of Algorithm B.
iv. Analyse the running time complexity of the algorithm A by formulating and solving a recurrence relation for the same. 8

Q3. (a) The Quick sort algorithm produces highly unbalanced partitions for already sorted inputs. Analyze this situation by formulating and solving a recurrence relation to determine its time complexity. Suggest an improvement to the pivot selection strategy to reduce the likelihood of this behaviour. Justify your answer. 7

(b) A city's emergency response system is modelled as a graph, where locations such as hospitals, fire stations, and intersections are represented as vertices, and roads connecting them are represented as edges. All roads are assumed to have equal travel time. Suggest an algorithm for the control centre to determine the minimum number of roads required to reach every location from a central emergency hub A in the following graph. Find out the path determined by the algorithm from A to all other locations. 8
The graph is defined as follows:
Vertices: {A, B, C, D, E, F, G}
Edges: {A–B, A–C, B–D, B–E, C–F, E–F, D–G, F–G}

Q4. (a) The subset-sum problem involves a given set $X = \{x_1, x_2, x_3 \ldots\ldots x_n\}$ and a positive integer target sum S. Solve the problem for the set X = {2, 3, 7, 8, 10}, S = 11 and determine its running time. 7

(b) Consider a graph with n vertices and m edges, where all edge weights are non-negative. Analyze the time complexity of Dijkstra's algorithm when implemented using a priority queue (min-heap). Explain the role of the priority queue in improving the efficiency of the algorithm. 8

Q5. (a) A digital media company schedules video programs on a streaming platform. Each program has a start time, end time, and associated revenue, and only one program can be shown at a time. Which algorithm will maximize the total revenue? Also, determine the optimal set of programs that maximizes the revenue. Explain why a greedy approach may not yield the optimal solution.

| Program | Start | End | Revenue |
|---|---|---|---|
| P1 | 1 | 3 | 5 |
| P2 | 2 | 5 | 6 |
| P3 | 4 | 6 | 5 |
| P4 | 6 | 7 | 4 |
| P5 | 5 | 8 | 11 |
| P6 | 7 | 9 | 2 |

7

(b) Write a recursive algorithm to find the $i^{th}$ Fibonacci number. How can the algorithm be improved using dynamic programming? Compute the $6^{th}$ Fibonacci number using the dynamic programming. 8

Q6. (a) "Bubble sort introduces a flag variable which is set to true if no exchange is made for an entire pass over the array. It reduces the number of passes." Explain with an example, how this variable helps reducing the number of passes. 7

(b) A traveller has a backpack with a maximum capacity of 10 kg. There are 5 items available: 8

| Item | Weight (kg) | Value |
|---|---|---|
| 1 | 2 | 6 |
| 2 | 3 | 10 |
| 3 | 5 | 12 |
| 4 | 4 | 8 |
| 5 | 1 | 3 |

Each item can either be taken completely or left. Find the maximum total value the traveller can carry without exceeding the capacity and list the selected items. Design a dynamic programming table to solve the problem and state its time complexity.

Q7. (a) Write the best-case and worst-case time complexities of insertion and deletion operations in hashing when collisions are resolved using the chaining method. Explain it with an example. 7

(b) What is a bipartite graph? Describe how Breadth-First Search (BFS) can be used to check whether a given graph is bipartite. State its time complexity. 8