**ST JOSEPH’S UNIVERSITY, BENGALURU -27**

Registration Number:

Date & session:

**M.Sc (Computer Science) – I SEMESTER**

**SEMESTER EXAMINATION: OCTOBER 2023**

**(Examination conducted in November /December 2023)**

**CS 7321 – DESIGN AND ANALYSIS OF ALGORITHMS**

**(For current batch students only)**

**Time: 2 Hours Max Marks: 50**

**This paper contains TWO printed pages and THREE parts**

**PART-A**

**Answer all the questions (2\*5=10)**

1.Define minimal cost spanning tree.

2. Explain time and space complexity.

3. What are non-deterministic algorithms?

4. What is the control abstraction for greedy method?

5. Define the following terms

 a. Live Node

 b. Dead Node

**PART B**

**Answer any five of the following questions (4\*5=20)**

6. Solve the following graph to calculate the minimum spanning tree using Kruskal’s Algorithm.



7. Compare and contrast between Merge Sort and Quick Sort.

8. Illustrate the eight queens’ problem with the help of a relevant example.

9. Explain Prim’s algorithm for minimal spanning tree with an example.

10. Simulate Quick sort algorithm for the following example 25,36,12,4,5,16,58,54,24,16,9,65,78.

11. Construct the shortest path for the following graph using Dijkstra’s algorithm.



12. Apply Warshall’s Algorithm to find the transitive closure of the digraph defined by

the following adjacency matrix



**PART C**

**Answer any two of the following questions (2\*10=20)**

13.a. What is Branch and Bound theory?

 b. Explain how the Traveling salesman problem can be solved using the Branch and Bound

 technique. Elucidate your answer with a clear example. [3+7]

14. a. Design an algorithm to find the maximum and minimum of a list of numbers using Divide

 and Conquer technique.

 b. Explain the three types of Asymptotic Notations with relevant examples

15. a. Mention the constraints of n queen problem.

b. Describe Greedy knapsack problem. Find the optimal solution of the Knapsack instance n= 7, M=15, (p1, p2,……p7) = (10,5,15,7,6,18,3) and (w1,w2,...w7)=(2,3,5,7,1,4,1). [3+7]

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*