M.Sc DEGREE EXAMINATION,APRIL 2019
I Year II Semester
Design and Analysis of Algorithms

Time : 3 Hours **Max.marks :75**

Section A $(10 \times 2 = 20)$ Marks

Answer any **TEN** questions

1. Define the terms Time and Space Complexity.
2. Define "Big-Oh".
3. What is a Strassen's Matrix? Give example.
4. State the control abstraction of Divide and Conquer Strategy.
5. What are Bi-connected Components? Give example.
6. Define Graph. Mention the ways using which a graph can be represented.
7. What are implicit and explicit constraints?
8. Give a note on Hamiltonian Cycle?
9. When a problem is said to be NP – Complete?
10. State the Graph Colouring problem.
11. Differentiate between Las Vegas and Monte Carlo Algorithms.
12. All NP – Complete problems are NP – Hard but some NP – Hard problems are not NP – Complete – True/False.

Section B $(5 \times 5 = 25)$ Marks

Answer any **FIVE** questions

13. Describe about the concept of Repeated Element Testing with Random Number Generation algorithm.
14. Explain the Greedy Knapsack algorithm with all three cases with example.
15. Give the Breadth First Search algorithm with an example graph.
16. Write the algorithm of Graph Colouring.
17. How will you implement Comparison Trees for problems based on Selection.
18. State the problem of Job Sequencing with deadlines. Write the algorithm.
19. Discuss in detail about editing a String using the concept of Dynamic Programming.

Section C $(3 \times 10 = 30)$ Marks

Answer any **THREE** questions

20. Discuss in detail about the Merge Sort algorithm with an example. Derive time complexity of the algorithm.
21. Write the Quick Sort algorithm. Explain step by step how the quick sort algorithm sorts the following set of elements in ascending order.
    27, 97, -9, 45, 21, 11, 37, 2
22. Write the following algorithms:
    (a) Floyd's Algorithm (b) Bellman Ford Algorithm
23. State and solve N – Queens' problem using Backtracking with an example. (N=4)
24. How will you derive the lower bounds for sorting algorithms using comparison trees.

M.Sc DEGREE EXAMINATION,APRIL 2019
I Year II Semester
Design and Analysis of Algorithms

Time : 3 Hours **Max.marks :75**

Section A $(10 \times 2 = 20)$ Marks

Answer any **TEN** questions

1. Define the terms Time and Space Complexity.
2. Define "Big-Oh".
3. What is a Strassen's Matrix? Give example.
4. State the control abstraction of Divide and Conquer Strategy.
5. What are Bi-connected Components? Give example.
6. Define Graph. Mention the ways using which a graph can be represented.
7. What are implicit and explicit constraints?
8. Give a note on Hamiltonian Cycle?
9. When a problem is said to be NP – Complete?
10. State the Graph Colouring problem.
11. Differentiate between Las Vegas and Monte Carlo Algorithms.
12. All NP – Complete problems are NP – Hard but some NP – Hard problems are not NP – Complete – True/False.

Section B $(5 \times 5 = 25)$ Marks

Answer any **FIVE** questions

13. Describe about the concept of Repeated Element Testing with Random Number Generation algorithm.
14. Explain the Greedy Knapsack algorithm with all three cases with example.
15. Give the Breadth First Search algorithm with an example graph.
16. Write the algorithm of Graph Colouring.
17. How will you implement Comparison Trees for problems based on Selection.
18. State the problem of Job Sequencing with deadlines. Write the algorithm.
19. Discuss in detail about editing a String using the concept of Dynamic Programming.

Section C $(3 \times 10 = 30)$ Marks

Answer any **THREE** questions

20. Discuss in detail about the Merge Sort algorithm with an example. Derive time complexity of the algorithm.
21. Write the Quick Sort algorithm. Explain step by step how the quick sort algorithm sorts the following set of elements in ascending order.
    27, 97, -9, 45, 21, 11, 37, 2
22. Write the following algorithms:
    (a) Floyd's Algorithm (b) Bellman Ford Algorithm
23. State and solve N – Queens' problem using Backtracking with an example. (N=4)
24. How will you derive the lower bounds for sorting algorithms using comparison trees.