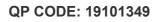
|--|





Reg No	:	
Name	:	

Time: 3 Hours

R

**B.Sc/BCA DEGREE (CBCS) EXAMINATION, MAY 2019** 

**Fourth Semester** 

Core Course - CS4CRT09 - DESIGN AND ANALYSIS OF ALGORITHMS

(Common for B.Sc Information Technology Model III, Bachelor of Computer Application)

2017 Admission onwards

3965FC9E

Maximum Marks: 80

Part A

Answer any ten questions.

Each question carries 2 marks.

- List out algorithm techniques. 1.
- 2. What is best-case complexity?
- 3. List any four examples of problems using Divide and Conquer.
- 4. State the average case and worst case complexity of quicksort.
- 5. Write the complexity of;

a) Selection sort b) Mergesort

- What is knapsack problem? 6.
- Explain the method of Kruskal's algorithm. 7.
- State Principle of Optimality. 8.
- Define the single source shortest path problem. 9.
- Give the time complexity and space complexity of TSP. 10.
- 11. Define a planar graph.
- 12. What is a state space tree?

(10×2=20)

## Part B

Answer any six questions.

Each question carries 5 marks.

- 13. Explain the Performance Analysis.
- 14. Compare time complexity and space complexity.
- 15. Illustrate the binary search algorithm with an example.
- 16. State the greedy method. Differentiate between the subset paradigm and ordering paradigm.
- 17. Explain in detail anyone of the problem solved using Greedy Method.
- 18. Explain all pair shortest path with algorithm.



- 19. Write the algorithm for 0/1 knapsack problem with example.
- 20. What are the basic differences in representing the directed and undirected graph?
- 21. Explain Hamiltanion circuit with suitable example.

(6×5=30)

## Part C

Answer any two questions.

## Each question carries **15** marks.

- 22. What is an algorithm? Explain the different characteristics of algorithm. Explain the different areas of algorithm study.
- 23. Write an algorithm for Merge Sort and Derive its time complexity.
- 24. Explain Prim's algorithm with an example.
- 25. Explain the graph coloring problem and draw the state space tree for m=3 colors and n=4 vertices graph.

(2×15=30)