14/6/16 AN

Reg. No. | | | | | | | | | | | | |

Question Paper Code : 27403

5 Years M.Sc. DEGREE EXAMINATION, MAY/JUNE 2016

Fifth Semester

Computer Technology

XCS 355/10677 SW 503 – DESIGN AND ANALYSIS OF ALGORITHMS

(Common to 5 Year M.Sc. Software Engineering and 5 Year M.Sc. – Information Technology)

(Regulations 2003/2010)

Time : Three Hours **Maximum : 100 Marks**

Answer ALL questions.

PART – A (10 × 2 = 20 Marks)

1. Using the step count method analyze the time complexity when two m x n matrices are added.
2. Give the control abstraction for divide and conquer technique.
3. What do you mean by feasible solution and optimal solution ?
4. Define principle of optimality.
5. Differentiate Binary tree from Binary Search Tree.
6. What is the use of Spanning tree ? Give examples.
7. Write any two applications where branch and bound technique can be applied.
8. What is meant by Graph colouring ?
9. What is non-deterministic algorithm ?
10. What is the need for approximation algorithm ?

PART – B (5 × 16 = 80 Marks)

11. (a) Explain how the minimum and maximum number is identified in a set of n numbers by using divide and conquer method. **(16)**

OR

(b) Describe how quicksort algorithm sorts the following sequences of keys. **(16)**

50, 10, 25, 30, 15, 70, 35, 55

12. (a) Solve the knapsack problem using greedy technique. **(16)**

OR

(b) What is multistage graph ? Explain with an example. Write the procedure for finding the minimum cost path using backward approach. **(16)**

13. (a) Explain the Breadth First Search algorithm. Compare it with Depth First Search. **(16)**

OR

(b) (i) Write a note on spanning trees. **(10)**

(ii) What do you mean by biconnected graph ? How is it used in DFS of a graph ? **(6)**

14. (a) (i) Define Hamiltonian cycle. **(4)**

(ii) Discuss the algorithm for graph colouring. **(12)**

OR

(b) Explain the backtracking algorithm for solving the 8 queens problem. **(16)**

15. (a) What is NP hard problem ? Discuss in detail NP hard scheduling problem. **(16)**

OR

(b) (i) Write a brief note on NP-Hard Code generation problem. **(8)**

(ii) Explain the need for common subexpressions in code generation. **(8)**