

No. of Printed Pages : 4

**BACHELOR OF COMPUTER APPLICATIONS
(BCA) (Revised)**

Term-End Examination, 2019

BCS-042 : INTRODUCTION TO ALGORITHM DESIGN

Time : 2 Hours

Maximum Marks : 50

Note : Question no. 1 is compulsory. Answer any three questions from the rest.

1. (a) Let $f(n)$ and $g(n)$ are two positive functions, using basic definition of Big Oh ("O") and Theta (θ), prove/disprove the following : [5]

(i) $\max\{f(n), g(n)\} = \theta(f(n) + g(n))$

(ii) $2^n = O(2^{n+1})$

(b) Solve the following Recurrence using Recursion tree method : [5]

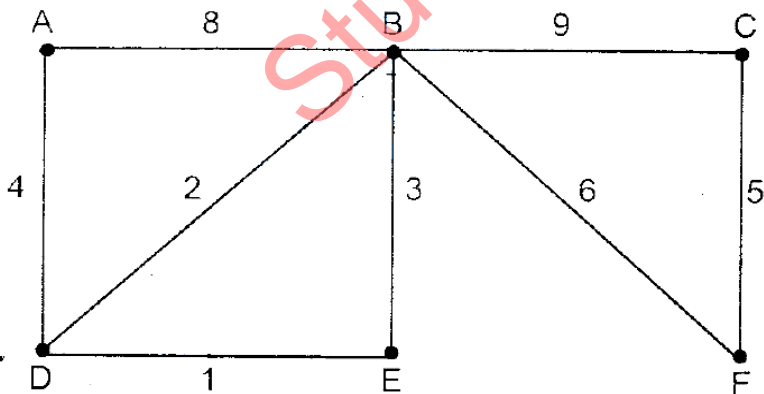
$$T(n) = 2T(n-1) + 1$$

Find tight solution of the Recurrence.

(c) Explain general algorithm to solve any problem using Greedy techniques. Write any two characteristics of Greedy Algorithm. [5]

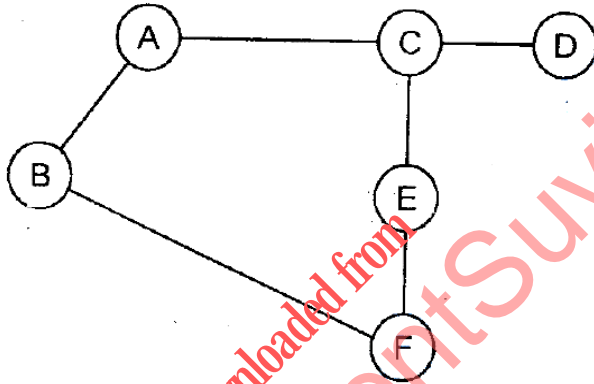
(d) Write Algorithm to solve Knapsack (fractional) problem using Greedy Method. Find the running time of the algorithm also. [5]

2. (a) Define minimum spanning tree. Apply Kruskal's Algorithm to find minimum cost spanning tree for the following graph : [6]



(b) Write bubble sort algorithm and find its time complexity in worst case. [4]

3. (a) For the following graph write DFS (sequence of traversal) from the node A : [4]

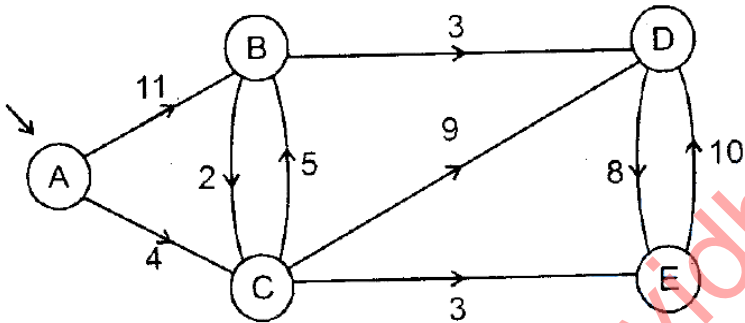


- (b) Apply master method to solve the following recurrence relation : [6]

(i)
$$T(n) = 2T\left(\frac{n}{2}\right) + n$$

(ii)
$$T(n) = 3T\left(\frac{n}{4}\right) + n \log n$$

4. Explain Dijkstra's Algorithm to find a single source shortest path in a given graph. Apply Dijkstra's Algorithm and find the shortest path from source vertex 'A' to rest of the vertices : [10]



5. Differentiate between the following with respect to method of solving a problem and time complexity : [10]

- (a) Depth-First-Search (DFS) Vs. Breath-First-Search (BFS)
- (b) Bellman-Ford Algorithm Vs Dijkstra's Algorithm for single source shortest path.

----- x -----