



Name : .....

Roll No. : .....

Invigilator's Signature : .....

**CS/B.TECH(CSE)(N)/SEM-5/CS-501/2012-13**

**2012**

**DESIGN AND ANALYSIS OF ALGORITHM**

Time Allotted : 3 Hours

Full Marks : 70

*The figures in the margin indicate full marks.*

*Candidates are required to give their answers in their own words as far as practicable.*

**GROUP – A**

**( Multiple Choice Type Questions )**

1. Choose the correct alternatives for the following :  $10 \times 1 = 10$ 
  - i) The Big O Notation of the expression  $f(n) = n \log_2 n + n^2 + e^{\log_2 n}$  is
    - a)  $O(n \log_2 n)$
    - b)  $O(n^2)$
    - c)  $O(n)$
    - d)  $O(e^{\log_2 n})$ .
  - ii) Traveling Salesman Problem is
    - a) NP Hard
    - b) NP
    - c) NP Complete
    - d) none of these.
  - iii)  $o(g(n))$  is [ Read as small oh of  $g(n)$  ] is
    - a) asymptotically loose
    - b) asymptotically tight
    - c) same as Big Oh
    - d) None of these.
  - iv) Complexity the recurrence relation  $T(n) = 8T\left(\frac{n}{2}\right) + n^2$  is
    - a)  $O(n)$
    - b)  $O(n^2)$
    - c)  $O(\log_2 n)$
    - d)  $O(n^3)$ .



- v) Kruskal's Algorithm is an example of
- a) Dynamic Programming
  - b) Greedy Method
  - c) Both (a) and (b)
  - d) None of these.
- vi) Complexity of Tower of Hanoi problem is
- a)  $O(n)$
  - b)  $O(n^2)$
  - c)  $O(2^n)$
  - d) None of these.
- vii) Binary Search algorithm can't be applied to
- a) Sorted linked lists
  - b) Sorted binary trees
  - c) Sorted linear array
  - d) Sorted integer array.
- viii) The technique of Pruning is used in
- a) Branch and Bound
  - b) Backtracking
  - c) Divide and Conquer
  - d) Dynamic Programming.
- ix) The tight bound for building a max heap is
- a)  $O(n)$
  - b)  $O(\log_2 n)$
  - c)  $O(n \cdot \log_2 n)$
  - d) None of these.
- x) The worst case running time of a quick sort algorithm is
- a)  $O(n^2)$
  - b)  $O(n \cdot \log_2 n)$
  - c)  $O(n)$
  - d)  $O(\log_2 n)$ .

**GROUP - B**

**( Short Answer Type Questions )**

Answer any *three* of the following.  $3 \times 5 = 15$

2. Find the best and worst case time complexity of quick sort.
3. State Master's theorem and find out the time complexity for the recurrence  $T(n) = T(2n/3) + 1$ .



4. Find the optimal solution using greedy criteria for a knapsack having capacity 100 kg for the following list of items having values and weights as shown in the table.

Item	Value	Weight
$I_1$	10	15
$I_2$	20	25
$I_3$	30	35
$I_4$	40	45
$I_5$	50	55

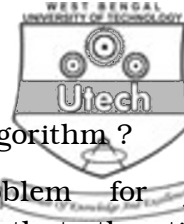
5. Compare and contrast BFS vs DFS.  
 6. Use the recursion tree to give an asymptotically tight solution to the recursion  $T(n) = T(n - a) + T(a) + cn$  where  $a \geq 1$  and  $c > 0$  are constant.

**GROUP - C**

**( Long Answer Type Questions )**

Answer any *three* of the following.  $3 \times 15 = 45$

7. Suppose we have a recurrence relation  $T(n) = aT\left(\frac{n}{b}\right) + f(n)$ , show that the following are true.
- If  $af\left(\frac{n}{b}\right) = kf(n)$  for some constant  $k < 1$ , then  
 $T(n) = O(f(n))$ . 5
  - If  $af\left(\frac{n}{b}\right) = kf(n)$  for some constant  $k > 1$ , then  
 $T(n) = O(n \log_b^a)$ . 5
  - If  $af\left(\frac{n}{b}\right) = kf(n)$  for some constant  $k = 1$ , then  
 $T(n) = O(n \log_b^a)$ . 5
8. a) Discuss the Bellman-Ford's algorithm for single-source shortest path problem. 7  
 b) Prove that the time-complexity of the algorithm is  $\Theta(V^2E)$ . 3  
 c) What is union-find algorithm ? Explain with an example. 5



9. a) What are the characteristics of greedy algorithm ? 3
- b) Discuss the activity selection problem for job sequencing with an example. Prove that the time complexity of the algorithm is  $O(n \log n)$ . 5 + 3
- c) Differentiate between greedy method and dynamic programming. 4
10. a) Explain the max-flow min-cut theorem with an example. 6
- b) Compare and contrast BFS and DFS. State the 0/1 knapsack problem. 3 + 2
- c) Describe the Clique Decision Problem (CDP). Prove that the CDP is NP complete. 2 + 2
11. Write short notes on any *three* of the following : 3 × 5
- a) Asymptotic Notations
  - b) Strassen's Matrix Multiplication
  - c) Approximation Algorithms
  - d) Knuth-Morris-Pratt Algorithm
  - e) Recursion Trees.

