ROLL NO. _____

Code: AC64/AT64 Subject: DESIGN & ANALYSIS OF ALGORITHMS

AMIETE – CS/IT

Time: 3 Hours **DECEMBER 2013** **Max. Marks: 100**

PLEASE WRITE YOUR ROLL NO. AT THE SPACE PROVIDED ON EACH PAGE IMMEDIATELY AFTER RECEIVING THE QUESTION PAPER.

NOTE: There are 9 Questions in all.

- **Question 1 is compulsory and carries 20 marks. Answer to Q.1 must be written in the space provided for it in the answer book supplied and nowhere else.**
- **The answer sheet for the Q.1 will be collected by the invigilator after 45 minutes of the commencement of the examination.**
- **Out of the remaining EIGHT Questions answer any FIVE Questions. Each question carries 16 marks.**
- **Any required data not explicitly given, may be suitably assumed and stated.**

Q.1 Choose the correct or the best alternative in the following: (2×10)

a. If every node u in G is adjacent to every other node v in G, A graph is said to be

(A) isolated **(B)** complete
(C) finite **(D)** strongly connected

b. The worst case occur in linear search algorithm when

(A) Item is somewhere in the middle of the array
(B) Item is not in the array at all
(C) Item is the last element in the array
(D) Item is the last element in the array or is not there at all

c. The complexity of binary search algorithm is

(A) O(n) **(B)** O(log n)
(C) $O(n^2)$ **(D)** O(n log n)

d. An algorithm that calls itself directly or indirectly is known as

(A) sub algorithm **(B)** recursion
(C) polish notation **(D)** traversal algorithm

e. Back-Tracking and Branch-and-bound based solutions use _____

(A) Spanning Tree **(B)** Decision Tree
(C) Binary Tree **(D)** State-space Tree

f. Travelling Salesman problem belongs to complexity class _____

(A) P **(B)** NP
(C) P & NP **(D)** None of these

g. Fractional knapsack problem is solvable by

(A) Dynamic programming **(B)** Greedy strategy
(C) Branch & Bound **(D)** None of these

h. Which of the following is shortest path algorithm?

(A) Kruskal's algorithm **(B)** Dijkstra's algorithm
(C) Gaussian elimination **(D)** None of these

i. Which of the following problem(s) can be solved by exhaustive-search algorithms?

(A) Travelling salesman **(B)** Knapsack
(C) Assignment **(D)** All of these

j. Following sequence of operation is performed on a stack. Push(1), Push(2), Pop, Push(1), Push(2), Pop, Pop, Pop, Push(2), Pop. The sequences of popped out values are

(A) 2,2,1,2,2 **(B)** 2,2,1,1,2
(C) 2,1,2,2,1 **(D)** 2,1,2,2,2

Answer any FIVE Questions out of EIGHT Questions.
Each question carries 16 marks.

Q.2 a. What is time and space complexity of algorithms? Provide the complexity of various searching and sorting techniques. **(8)**

b. Write an algorithm for computing n!, also find its complexity. **(8)**

Q.3 a. Write a recursive algorithm for solving Tower of Hanoi problem. What is the basic operation in the Tower of Hanoi problem and give the recurrence relation for the number of moves? **(10)**

b. What is the computational complexity of the Fibonacci sequence and how is it calculated? **(6)**

Q.4 a. Write an algorithm used to partition an array for quicksort. Explain and calculate its complexity. **(8)**

b. Explain an optimal order algorithm for multiplying n matrices. **(8)**

Q.5 a. Write and explain heapsort algorithm and show that its worst case performance is O(n log n). **(10)**

b. Solve the system by Gaussian Elimination **(6)**

$$2x_1 - x_2 + x_3 = 1$$
$$4x_1 + x_2 - x_3 = 5$$
$$x_1 + x_2 + x_3 = 0$$

Q.6 a. Describe in detail both Prim's and Kruskal's algorithms for finding a minimum cost spanning tree of an undirected graph with edges labelled with positive costs and explain why they are correct. Compare the relative merits of the two algorithms. **(8)**

b. Explain how the knapsack problem is solved with approximation algorithm.**(8)**

Q.7 a. When does a directed acyclic graph (DAG) yield a unique topological sort? Explain with the help of an example. Write an algorithm for finding topological sort of a DAG. **(8)**

b. Explain Breadth-First Search (BFS) algorithm using a suitable example. How it differs from DFS? Also explain which is preferable and when. **(8)**

Q.8 a. Write a note on the challenges of numerical algorithms. **(8)**

b. Write Comparison–Counting-Sort algorithm. Will this algorithm work correctly for arrays with equal values? **(8)**

Q.9 a. Apply the branch-and-bound technique in solving the travelling Salesman Problem. **(8)**

b. Give a template for a generic backtracking algorithm. What are the additional features required in branch-and-bound when compared to backtracking? **(8)**