ROLL NO. ________________

AMIETE – CS/IT (Current & New Scheme)

Time: 3 Hours **JUNE 2016** **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. For the Euclid's algorithm indicate its basic operation
(A) Addition of two numbers **(B)** Multiplication of two integers
(C) Comparison of two numbers **(D)** Modulo division

b. Two main measures for the efficiency of an algorithm are
(A) Processor and memory **(B)** Complexity and capacity
(C) Time and space **(D)** Data and space

c. Find the number of comparisons made by the sentinel version of sequential search in the worst case.
(A) n **(B)** (n+1)
(C) n/2 **(D)** (n–1)

d. In a binary tree, certain null entries are replaced by special pointers which point to nodes higher in the tree for efficiency. These special pointers are called
(A) Leaf **(B)** Branch
(C) Path **(D)** Thread

e. 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

f. The upper bound on the time complexity of the nondeterministic sorting algorithm is
(A) O (n) **(B)** O (n lg n)
(C) O (1) **(D)** O (lg n)

g. What would be the depth of tree whose level is 9?
(A) 10 **(B)** 8
(C) 9 **(D)** 11

h. A node of a directed graph G having no out-degree and a positive in-degree is called

(A) Source node **(B)** Sink node
(C) Sibling node **(D)** Null node

i. The asymptotic notation for defining the average time complexity is

(A) Equivalence **(B)** Symmetric
(C) Reflexive **(D)** Transitive

j. The following are the statements regarding the NP problems. Choose the right option from the following options:

I. All NP-complete problems are not NP-hard.
II. Some NP-hard problems are not known to be NP-complete.

(A) Both (I) and (II) are true **(B)** Both (I) and (II) are false
(C) Only (I) is true **(D)** Only (II) is true

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

Q.2 a. What is the smallest value of n such that an algorithm whose running time is $100n^2$ runs faster than an algorithm whose running time is 2^n on the same machine? **(8)**

b. int f(int x) {
if (x < 1) return 1;
else return f(x-1) + g(x); }
int g(int x) {
if (x < 2) return 2;
else return f(x-1) + g(x/2); }
What best describes the growth of f(x) as a function of x? Give f(x). **(8)**

Q.3 a. Justify whether or not the function $\lceil \lg n \rceil!$ and $\lceil \lg\lg n \rceil$ is polynomially bounded? What is the asymptotic behaviour of these functions w.r.t. the polynomial functions? **(8)**

b. Consider the analysis of linear time selection algorithm for groups of k. Let $x \geq \lceil k/2 \rceil (\lceil 1/2 \lceil n/k \rceil \rceil - 2)$. Solve the recurrence by substitution, $T(n) \leq T(\lceil n/k \rceil) + T(n-x) + O(n)$ and mention worst case time complexity for groups of 5,3. **(8)**

Q.4 a. While comparing implementations of insertion sort and merge sort on the same machine, for inputs of size n, insertion sort runs in $8n^2$ steps, while merge sort runs in ($64\ n\ lg\ n$) steps. For which threshold value of n does insertion sort beats merge sort? Given the complexity, if a binary search is used to improve the worst-case running time of insertion sort. **(8)**

b. What is the largest k such that if you can multiply 3×3 matrices using k multiplications (not assuming commutativity of multiplication), then you can multiply n×n matrices in time $O(n^{\lg 7})$? What would be running time of this algorithm? **(8)**

Q.5 a. A table with row and column labels WHITE, GRAY, and BLACK. Indicate for each cell (i, j) whether, at any point during a DFS of a directed graph, there can be an edge from a vertex of *color i* to a vertex of *color j*. For each possible edge, indicate what edge types it can be. Also make the table for the DFS of an undirected graph. **(8)**

b. Another way to perform topological sorting on a directed acyclic graph G = (V,E) is to repeatedly find a vertex of in-degree 0 (zero), output it, and remove it and all of its outgoing edges from the graph. Explain how to implement this idea so that it runs in time O(V+E). **(8)**

Q.6 a. Consider the following variation of the BOTTOM-UP HEAP CONSTRUCTION procedure. **(8)**

```
BOTTOM-UP-HEAP'(A)
A.heap-size = 1
for i = 2 to A.length
MAX-HEAP-INSERT(A, A[i])
```

```
MAX-HEAP-INSERT(A, key)
A.heap-size = A.heap-size + 1
A[A.heap-size] = −∞
HEAP-INCREASE-KEY(A, A.heap-size, key)
```

```
HEAP-INCREASE-KEY(A, i, key)
if key < A[i]
    error "new key is smaller than current key"
A[i] = key
while i > 1 and A[PARENT(i)] < A[i]
    exchange A[i] with A[PARENT(i)]
    i = PARENT(i)
```

Does the procedures BOTTOM-UP HEAP CONSTRUCTION and BOTTOM-UP-HEAP' always create the same heap for the list (1, 2, 3, 4, 5, 6)? Give the worst case complexity of BOTTOM-UP-HEAP' to build n-element heap.

b. In context to AVL trees, draw diagrams of the single L-rotation and of the double RL-rotation in their general form. **(8)**

Q.7 a. For the bottom-up dynamic programming algorithm for the knapsack problem, prove that, (i) its time efficiency is θ (nW) (ii) its space efficiency is θ (nW) (iii) the time needed to find the composition of an optimal subset from a filled dynamic programming table is O (n). **(8)**

b. Indicate whether the following statements are true or false: **(8)**

(i) If e is a minimum-weight edge in a connected weighted graph, it must be among edges of at least one minimum spanning tree of the graph.

(ii) If e is a minimum-weight edge in a connected weighted graph, it must be among edges of each minimum spanning tree of the graph.

(iii) If edge weights of a connected weighted graph are all distinct, the graph must have exactly one minimum spanning tree.

(iv) If edge weights of a connected weighted graph are not all distinct, the graph must have more than one minimum spanning tree.

Q.8 a. Find the probability of all *n* keys being hashed to the same cell of a hash table of size *m* if the hash function distributes keys evenly among all the cells of the table. **(8)**

b. In which context does the NP algorithms is polynomial? Give example of the running time of P and NP algorithms. **(8)**

Q.9 a. In context to n-queen problem, give the possible placement of queen if the queen in the table is placed at T[0,3] and draw the table. **(8)**

b. TSP: How will you show that there exists a lower bound for the travelling salesman problem? **(8)**