CSE 5311 Design and Analysis of Algorithms

Sample Questions
The quiz will have questions for approx. 70 minutes

1. Choose the MOST appropriate answer and circle the letter corresponding to your answer. (**15 Minutes**)

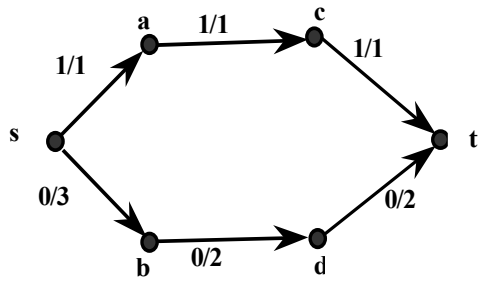
The penalty for each INCORRECT answer is 0.25 point.

i) Input comprises a sorted list of *n* integers with many duplications such that the number of distinct integers in the sequence is O (*log n*), the time complexity to find an element in the list is given by,

A. O(*n*) B. O(*log n*) C. O(*log log n*) D. Θ (1)

ii) The residual capacity of the augmenting path for the network shown below is

A. 1 B. 2 C. 3 D. 7



iii) Which of the following is not $O(n^2)$

A. $n + 10000n$ B. $n^{1.9999}$ C. $10^5 n + 2^6 n$ D. $n^3/\sqrt{n}$

iv) The augmenting path in a flow network
- A. determines the edge connectivity of a network
- B. comprises edges which can admit positive flow
- C. converts a tree network into a cyclic chain
- D. is used only for cyclic chain networks

v) We employ dynamic programming approach when
- A. It gives optimal solution
- B. The solution has optimal substructure
- C. It is faster than Greedy technique

D. The given problem can be reduced to the 3-SAT problem.

vi) A problem is said to be NP-Complete
A. If it is as 'hard' as any problem in NP
B. A non-polynomial time algorithm has been discovered
C. A polynomial time algorithm can exist but needs a parallel computer
D. There is Greedy solution to the problem

vii) Which of the following problems is known to have a polynomial time solution
A. Longest simple path problem for a given graph
B. The 3-colorability problem in graphs
C. The Eulerian cycle in a graph
D. The Hamiltonian Cycle in a graph

viii) The distance matrix of a graph with vertices P,Q, R and S is given by

$$\begin{bmatrix} 0 & 1 & \infty & \infty \\ \infty & 0 & 2 & 4 \\ 3 & \infty & 0 & 1 \\ 1 & \infty & \infty & 0 \end{bmatrix}$$

The shortest path from Q to S consists of edges
A. QR and RS B. QS C. PQ and RS D. there is no path

ix) Which of the following basic algorithms can be used to most efficiently determine the presence of a cycle in a given graph
A. minimum cost spanning tree algorithm
B. Ford-Fulkerson's algorithm
C. breadth-first search algorithm
D. depth-first search algorithm

x) Working modulo q= 11, how many spurious hits does the Rabin-Karp matcher encounter in the text T = 31415926 when looking for the pattern P = 26

A. None B. One C. Two D. Three

2. State whether the following statements are TRUE or FALSE

(**15 Minutes**)

Penalty for each incorrect answer is 0.5 point

i) If $f(n) = O\ g(n)$ and $g(n) = O\ (h(n))$ then $f(n) = \Omega\ (h(n))$

ii) A binary search tree on n integer keys in the range from 1 to n^2 can be constructed in $O(n)$ worst-case time.

iii) Kruskal's algorithm of finding a minimum spanning tree of a weighted undirected graph is an example of a dynamic programming algorithm.

iv) The transitive closure of a directed graph $G = (V, E)$ can be computed in $O(v^2)$ time by the Floyd-Warshall algorithm, which repeatedly squares the adjacency matrix of G.

v) The (log n) th smallest number of n unsorted numbers can be determined in $O(n)$ worst-case time.

vi) The worst-case running time randomized quicksort on an array of length n is $\Omega\ (n^2)$.

vii) In the bitonic merge-sorting network, there are n/2 stages with $\log n$ switches per stage.

viii) The recurrence for mergesort algorithm can be written as $T(n)=2T(n/2)+\Theta\ (n)$

ix) The time complexity of the Heapify algorithm is $O(n)$.

x) The asymptotically tight upper bound for $T(n) = T(n-2) + 1$ is $O\ (\log n)$.

3. Consider a set S of $n \geq 2$ distinct numbers given in unsorted order and, x and y are two distinct numbers in the set S. Write an $O(n)$ time algorithm to determine $x, y \in S$ such that $|x-y| \geq |w-z|$ for all $w,z \in S$.
(**15 Minutes**)

4. Examine the procedure DOES_SOMETHING below and answer the following questions. **(15 Minutes)**

a. What is the function of the procedure DOES_SOMETHING?
b. What is the time complexity of the procedure?
c. If A[*l* .. r] = [24, 30, 09, 46, 15, 19, 29, 86,78], what is the output?

Procedure **DOES_SOMETHING(A,*l*,r)**
Input : Array A(*l* .. r)

```
1.    x ← A[l]; i ← l; j ←r;
2.    while i < j do
3.        while A[i] ≤ x and i ≤ r do i ← i +1;
4.        while A[j] > x and j ≥ l do j ← j -1;
5.        if i< j then
6.            exchange A[i] ↔ A[j];
7.    q ← j;
8.    exchange A[l] ↔ A[q];
```

5. Show how to compute the length of an LCS (Longest Common Sequence) using only min (*m,n*) entries in an array plus O(1) additional space. (**15 Minutes**)

6. A graph is bipartite if all its vertices can be partitioned into two disjoint subsets *X* and *Y* so that every edge connects a vertex in *X* with a vertex in *Y*. Design an algorithm to check whether a given graph is bipartite. (**15 Minutes**)

7. What is an optimal Huffman code for the following set of frequencies, based on the first 8 Fibonacci numbers?
a:1 b:1 c:2 d:3 e:5 f:8 g:13 h:21
Can you generalize your answer to find the optimal code when the frequencies are the first *n* Fibonacci numbers?

(**15 Minutes**)