Comprehensive Exam - Data Structure and Algorithm

Note: All questions are compulsory. Total:-50 Marks

1. Prove or disprove the following:
   (a) $n! = O(n^n)$ [1-marks]
   (b) If $f(n) = O(g(n))$ and $g(n) = O(h(n))$ then, $h(n) = \Omega(f(n))$ [1-marks]
   (c) $f(n) = \log_2 n$ is $O(n^\alpha)$ for any $\alpha > 0$ [2-marks]

   Note: All O's are (big-O) notation.

2. With the help of mathematical induction show that when n is an exact power of 2, the recurrence function:

$$T(n) = \begin{cases} 2, & \text{if } n = 2 \\ 2T(n/2) + n, & \text{if } n = 2^i \text{ for } i > 0 \end{cases}$$

   solution comes out to be $T(n) = n \log n$ [2-marks]

3. Given an array A with distinct element and sorted in decreasing order, a Quick-Sort is applied to it, show the running time of Quick-short is $\Theta(n^2)$. What will be running time of Quick-Sort when all elements of array A have same value. [2-marks]

4. If a queue Q is given, how many stacks are needed to replicate the queue Q, Implement it, and also give the running time of the implementation. [2-Marks]

5. Mr. B. C. Dull claims to have developed a new data structure for priority queues that supports the operations Insert, Maximum, and Extract-Max - all in $O(1)$ worst- case time. Prove that he is mistaken. [2-marks]

6. Give asymptotic upper and lower bounds for $T(n)$ in each of the following recurrences. Assume that $T(n)$ is constant for sufficiently small n. Make your bounds as tight as possible, and justify your answers.

   (a) $T(n) = 4T(n/2) + n^2\sqrt{n}$ [2-marks]

   (b) $T(n) = T(n/2) + T(n/4) + T(n/8) + n.$ [2-marks]

7. A Minimum Bottleneck Spanning Tree of an undirected graph $G(V, E)$ is a spanning tree whose maximum weight edge is minimized. Explain how an MST is always a minimum bottleneck spanning tree, but the converse may not be true. [3-marks]

8. Given two BST's how do you check if they represent the same set of the elements? [3-marks]

9. Show a simple way to compute the $\sqrt{n}$ smallest element among n elements in $O(n)$ deterministic time, without using the Median of Medians algorithm. [4-marks]

10. Given a connected graph G, with n vertices and m edges, assume that the cost of all edges are distinct. A particular edge e of G is specified. Give an algorithm with running time $O(m + n)$ to decide whether e is contained in a minimum spanning tree of G. [4-marks]

11. Bob algorithm enthusiastic wants to modify Strassen's algorithm to multiply $n \times n$ matrices in which n is not an exact power of 2? You as Bob friend derive an algorithm, and also show that the resulting algorithm runs in time $O(n^{\ln n})$. [4-marks]

12. Prove that, For any two nodes s and t in a directed graph G, their strong components are either identical or disjoint. [4-marks]

13. Show how to construct the min-heap in $O(n)$ time. In other words, show how an arbitrary array can be transformed to make sure that it satisfies the min-heap property in $O(n)$ time. [4-marks]

14. Consider an undirected graph $G = (V, E)$ with a weight function W providing non-negative real valued weights, such that the weights of all the edges are different. Prove that, G has a unique Minimum Spanning Tree. [4-marks]

15. Topological short on a directed acyclic graph $G = (V, E)$ can also be performed by repeatedly find a vertex of in-degree 0 (no incoming edges), output it, remove it and all of its outgoing edges from the graph. Explain how to put this concept into action so that it runs in time $O(V + E)$. What happens if G has cycles in this algorithm? [4-marks]