14/06/2022

Data Structures and Algorithms (CSE102)
Mid-Semester Examination, Summer Semester, 2022
Indraprastha Institute of Information Technology, Delhi

Duration: 2 Hrs **Full Marks: 25**

Answer all questions.

1. (a) Write an algorithm for the merge procedure in the mergesort algorithm that only requires $\mathcal{O}(1)$ extra space. Recall, the merge algorithm discussed in the class requires $\mathcal{O}(n)$ extra space. What is the time complexity of your algorithm? Answers with time complexity greater than $\mathcal{O}(n)$ are accepted. (3)

   (b) Compute the overall time complexity of the mergesort algorithm that uses your merge algorithm in part A. (2)

2. Let's say we have two sorted arrays of integers A and B. Write an algorithm that takes an integer x as input and finds two integers x_1 and x_2 such that x_1 belongs to A, x_2 belongs to B, and the sum of x_1 and x_2 is x. If no such x_1 and x_2 exist, the algorithm prints an error. The time complexity of your algorithm should be less than $\mathcal{O}(n^2)$. (4)

3. Write a recursive algorithm that takes n integers and an integer r as input and prints all $\binom{n}{r}$ combinations. For example, if the input numbers are 1, 2, 3, 4 and the value of r is 2, then all the following six combinations will be printed.
   1 2
   1 3
   1 4
   2 3
   2 4
   3 4
   Note that $\binom{n}{r}$ is all possible ways of selecting r items out of n numbers in which the order of selection doesn't matter. For example, 1 2 is the same as 2 1 because the order doesn't matter. (5)

4. (a) There are n balls in a row. Each ball is colored *red*, *white*, or *blue*. You are required to sort these balls so that all the red balls precede all the white balls, which in turn precede all the blue balls. The only operations you are allowed to perform are "*check*", which returns the color of a ball, and "*swap*" which swaps the positions of two balls. Give an $\mathcal{O}(n)$ time algorithm for this problem. Justify your answer. (3)

   (b) Consider k different colours $c_1, \ldots, c_k$, where k is a constant. Now, assume that there are n balls in a row where each ball is coloured $c_1, c_2, \ldots, orc_k$. Then sort these balls such that balls coloured c_i precede balls coloured c_j for all $i < j$. Generalize your algorithm for part(a) to give an algorithm for this problem. What is the complexity of your generalized algorithm? (2)

5. Create a binary search tree with the nodes $G, D, I, B, F, J, H, A, C, E$. Assume the ordering $A < B < C < \ < Z$. Suppose the node D is deleted. Construct the new binary search tree. Generate the output of the Post-order traversal of the tree before and after the deletion. (You should clearly state any assumptions that you may make). (3)

6. Write an algorithm to reverse a doubly-linked list. You can assume that the list is not empty and the head of the linked list is the input to the algorithm. You are not allowed to use an array in your algorithm. The time complexity of your algorithm should not be more than O(n). (3)

- - End - -