CS225 Homework 9

Graph and Arrays

For this assignment you will present algorithms and worse case complexity analysis for the problems listed below.

Deliverables: One MS Word, PDF, or equivalent file that contains the algorithms and complexity analysis for the problems given below. File naming convention (required!): *yourusername*HW4.docx (or appropriate file extension for your format). Deliver via Canvas.

Problem Description:

Problem 1: -- The input is an array of n integers, a[n], having index values from 0 – (n-1). For the jth index value, find the number of integers larger than the value at a[j]. Repeat for all index values 0 <= j <= (n-1). In other words, for each element in the array, find the number of values that are larger further down the array. Return the results in an array.

Examples:

Given (3, 0, 2, 4, 6, 1), return (2, 4, 2, 1, 0, 0)

Given (8, 0, 13, 8, 11, 6, 4, 7), return (2, 6, 0, 1, 0, 1, 1, 0)

Given (1, 1, 1, 1, 1), return (0, 0, 0, 0, 0)

Problem 2 – The input is an nxn array of 1’s and 0’s, a[n][n]. Return the index of the row containing the largest number of 1’s. If more than one row are tied for the highest, return the lowest index.

Examples:

[ 0, 0, 0, 1]

[1, 1, 0, 1]

[0, 0, 0, 0]

[1, 0, 1, 0] return 1, because row 1 contains the most 1’s.

[1, 0, 0, 0]

[0, 1, 0, 0]

[0, 0, 1, 0]

[0, 0, 0, 1] return 0, because row 0 is the lowest index value with one 1 in it.

Important note: Use n as the basis for the complexity analysis. This problem is equivalent to finding the node in a graph that is connected to the most other nodes. For a graph continuing n nodes, the edges in a graph are often represented by an nxn matrix as described above. A 1 represents an edge, a 0 represented no edge, between two nodes.

Instructions: Provide an algorithm in pseudo code for each of the two problems listed above. Provide the worst case complexity, O( f(n) ) for each of the two problems listed above. You do not need a formal analysis for complexity, but provide justification for your result.

Rubric: Per grading rubric below.

|  |  |  |
| --- | --- | --- |
| **Deliverable** | **Points** | **Awarded** |
| Pseudo Code format | 6 |  |
| Algorithm Correctness | 14 |  |
| Complexity | 10 |  |
| Totals | 30 |  |