Basic Algorithms CSCI-UA.0310

Homework 8

Due: April 12th, 4:00 PM EST

Instructions

Please answer each **Problem** on a separate page. Submissions must be uploaded to your account on Gradescope by the due date and time above.
Please note that no late submission will be accepted for this homework.

Problems To Submit

Problem 1 (20 points)

Recall the activity selection problem discussed in the lecture. We developed a greedy approach which in every step picks the interval with the earliest finish point among the remaining ones. Modify it in order to develop a different greedy algorithm picking the first interval based on another property. Show that your greedy algorithm actually works (You can assume that the greedy approach discussed in the lecture gives an optimal answer and may use it to prove the correctness of your algorithm).

Problem 2 (15 points)

Recall that if the input graph $G = (V, E)$ is given by an array of adjacency lists, then the running time of the BFS algorithm on G is $\Theta(|V| + |E|)$. Now suppose G is given using an adjacency matrix. Rewrite the pseudo-code for the BFS algorithm on G using its adjacency matrix. Find the time and space complexities using Θ expressions.

Problem 3 (15 points)

Given an undirected graph $G = (V, E)$, develop an $O(|V| + |E|)$-time algorithm to check whether G is connected. Write the pseudo-code of your algorithm and justify why your algorithm satisfies the required running time.

Problem 4 (25 points)

Let G be a directed graph on n vertices which is given as input by the adjacency matrix $V[1 \ldots n][1 \ldots n]$. Develop an $O(n)$-time algorithm to check if there is any vertex in G that has edges coming to it from all other vertices of G but no edges going out from it. Justify why your algorithm runs in $O(n)$ time.

Problem 5 (13+12 points)

We want to develop a divide and conquer-based algorithm to find the convex hull of n given points.

(a) First, given two convex polygons P and Q which have n points in total, develop an $O(n)$-time algorithm to find the convex hull of their union.
You have to fully explain your algorithm. You do NOT need to write the pseudo-code of your algorithm.

(b) Given n points in the plane, use part (a) to develop a divide and conquer-based algorithm to find their convex hull. Your algorithm must work in $O(n \log n)$ time.
Write the pseudo-code of your algorithm and justify why it satisfies the required running time.