CS 180: Introduction to Algorithms and Complexity Midterm Exam

Feb 17, 2021
Name -
UID -
Section -

* Print your name, UID and section number in the boxes above, and print your name at the top of every page.

* Exams will be scanned and graded in Gradescope. Use Dark pen or pencil. Handwriting should be clear and legible.

. The exam is a closed book exam. You can use your notes taken during the lecture.

. There are 4 problems. Each problem is worth 25 points.

. Do not write code using C or some programming language. Use English or clear and simple pseudo-code. Explain the idea of your algorithm and why it works.

. Your answer are supposed to be in a simple and understandable manner. Sloppy answers are expected to receiver fewer points.

. Don't spend too much time on any single problem. If you get stuck, move on to something else and come back later.

Problem 1

Cellphone company wants to install k towers to serve customers at locations $\mathcal{X} = x_1, ...x_n$. For simplicity we assume all locations are co-linear - they are points on the x axis. Each customer phone will be programmed to communicate with a single tower, and the company also has to decide the location of the k towers (the location of a tower is not necessarily the location of a customer). The assignment of customers to towers and the location of the towers are to be chosen such as to minimize the sum of the squares of the distances from a customer to the tower it is assigned to.

a.

Given a tower. Argue that in an optimal solution the customers assigned to it are consecutive locations in the sorted sequence of locations of customers.

b.

In an optimal solution what is the relation between between the location of customers served by a tower and the location of the tower? (High-school algebra)

c.

Assume in $\mathcal{X} = x_1, ..., x_n$ the locations are sorted. For consecutive customer locations in the sorted order $x_i, x_{i+1}, \ldots, x_j$ if we assign these customers to a single tower, and the tower serves only these locations, the cost is denoted by $C_{i,j}$. Give an efficient algorithm to find these costs for all (i, j), $i < j$. Each arithmetic operation is a unit cost. (hint: try to reuse computations)

d.

Outline an Dynamic Programming algorithm to solve the problem (Define the subproblems, and write the recursive algorithm). Analyse the complexity of the bottom up algorithm.

Problem 2

Professor Gafni has developed a hardware priority queue on his laptop. The priority queue can store up to p elements. This special priority queue can perform the INSERT and EXTRACT-MIN operations in $O(1)$ time no matter how many elements are stored in the queue. We wish to use this new Gafni-queue to sort $n = p^2$ elements stored in memory.

Give an linear time algorithm to sort the n elements.

Problem 3

In a country far far away, all highways connecting cities are one way, such that if the cities are nodes and highways are directed edges, then we get a simple strongly connected directed graph. Alas, this country is a greedy country: All highways have a toll (different for each highway), and so going from city i to city j we will have to pay the tolls on the road between them. In addition, each city has fee payable upon leaving the city. So, if I'm going from city i to city j through city w I will have to pay 2 tolls $i->w$ and $w->j$ in addition to two exit fees, from i and from w.

You relocated to the country because of the Pandemic.

a.

From your city, you wish to find the cheapest route to all other cities in the country. Suppose you have a computer program that solves the shortest path problem as discussed in class. How will you use it to solve our problem? The computer algorithm accepts directed weighted graphs. It does not understand the fees of cities.

b.

Design an $O(|E|logn)$ algorithm ($|E|$ is the number of roads, n is the number of cities. To get this complexity you have to make an obvious simple assumption).

c.

You live in city i and for all cities j, if you visit j you cannot crash there for too long, and eventually will have to return home to i. For all j find the minimum cost of the round trip. Design, give $O(|E|logn)$ algorithm.

Question 4

Given a undirected weighted graph $G = (E, V)$. With each spanning tree we associate a number which is the weight of the heaviest edge in the tree. Find the spanning tree whose number is minimum among all spanning trees.
Explain.
(Hint: this involves a simple observation. The question is how to justify this observation. There are many ways to justify. The easiest is probably to recall where "exponentiation" appeared in the HW and use it.)