CS 170 Homework 4

Due **2/20/2024, at 10:00 pm (grace period until 11:59pm)**

1 Study Group

List the names and SIDs of the members in your study group. If you have no collaborators, you must explicitly write "none".

2 Arbitrage

Shortest-path algorithms can also be applied to currency trading. Suppose we have n currencies $C = \{c_1, c_2, \ldots, c_n\}$: e.g., dollars, Euros, bitcoins, dogecoins, etc. For any pair of currencies c_i, c_j, there is an exchange rate $r_{i,j}$: you can buy $r_{i,j}$ units of currency c_j at the price of one unit of currency c_i. Assume that $r_{i,i} = 1$ and $r_{i,j} \geq 0$ for all i, j.

The Foreign Exchange Market Organization (FEMO) has hired Oski, a CS170 alumnus, to make sure that it is not possible to generate a profit through a cycle of exchanges; that is, for any currency $i \in C$, it is not possible to start with one unit of currency i, perform a series of exchanges, and end with more than one unit of currency i. (That is called *arbitrage*.)

More precisely, arbitrage is possible when there is a sequence of currencies $c_{i_1}, \ldots, c_{i_k}$ such that $r_{i_1,i_2} \cdot r_{i_2,i_3} \cdot \cdots \cdot r_{i_{k-1},i_k} \cdot r_{i_k,i_1} > 1$. This means that by starting with one unit of currency c_{i_1} and then successively converting it to currencies $c_{i_2}, c_{i_3}, \ldots, c_{i_k}$ and finally back to c_{i_1}, you would end up with more than one unit of currency c_{i_1}. Such anomalies last only a fraction of a minute on the currency exchange, but they provide an opportunity for profit.

We say that a set of exchange rates is arbitrage-free when there is no such sequence, i.e. it is not possible to profit by a series of exchanges.

(a) Give an efficient algorithm for the following problem: given a set of exchange rates $(r_{i,j})_{i,j \in n}$ which is *arbitrage-free*, and two specific currencies a, b, find the most profitable sequence of currency exchanges for converting currency a into currency b. That is, if you have a fixed amount of currency a, output a sequence of exchanges that gets you the maximum amount of currency b.

Hint 1: represent the currencies and rates by a graph whose edge weights are real numbers.

Hint 2: $\log(xy) = \log(x) + \log(y)$

(b) Oski is fed up of manually checking exchange rates, and has asked you for help to write a computer program to do his job for him. Give an efficient algorithm for detecting the possibility of arbitrage.

For both parts (a) and (b), give a three-part solution.

3 Not So Crazy Delivery

PNPenguins are running a new food delivery business where they wish to deliver ice-cakes to all their fellow penguins in the shortest amount of time. The penguins community is a network consisting of m households total and r two-way/bidirectional roads. The time it takes to travel between two households h_1,h_2 is given by $c(h_1, h_2)$. Due to limited budget, PNPenguins can only afford to have a maximum of k delivery centers within the community. PNPenguins have found a subset of n households ($m >> n > k$) within the community who are willing to sell their home to PNPenguins and become a delivery center. For this problem, assume that households always get ice-cakes delivered by the closest delivery center to them.

a) Design an algorithm that helps PNPenguins to find the most optimal locations of delivery centers; that is, a set of delivery centers that minimizes the maximum delivery time needed to all households. This algorithm need not run in polynomial time with respects to all variables. **Please provide a 3-part solution.**
Hint: First write an algorithm which computes the maximum delivery time for some fixed set of delivery centers. Then, use this algorithm to check the maximum delivery time for each possible set of delivery centers.

PNPenguins have now decided to use the locations you chose from part a). But right before they start the delivery service, Pesla (a highly innovative iceship company headquartered in PNP-Borderland) releases a new magical iceship model that allows passengers to skip over the t most time-consuming roads along any of their journeys. PNPenguins all agree to integrate this new technology into their delivery business.

b) Design an algorithm that finds the updated/new maximum delivery time for each household.
Please provide a 3-part solution.
Hint: create a graph such that edges and nodes encode information about whether an edge is skipped when traversed by a graph algorithm.

4 Three-Legged Race

You are a teacher for two classes, each of n students. You are organizing a three-legged race, where students are paired with students in the opposing class. Each student must be paired up, and each student will only be paired with one other student. In an ideal three-legged race, you and your partner are evenly matched in terms of stride. Luckily, you have data on the stride lengths of all your students. Design an algorithm to minimize the total difference in stride length between pairs.

Please provide a 3-part solution.

5 Cars vs. Bikes

PNPenguin is an avid cyclist, and has traveled to Penguinland to go on an epic bike ride. Unfortunately for PNPenguin, however, car traffic in Penguinland makes it dangerous to ride on certain roads.

More specifically, Penguinland consists of N buildings and M unidirectional roads each connecting two buildings, with each road having a nonnegative length l_i. One of the buildings is Penguincorp, a large local company, and all the other $N-1$ buildings are houses of Penguinland residents, all of whom work at Penguincorp. Every morning, each resident of Penguinland will drive from their house to Penguincorp on the shortest possible route. Each resident has a unique shortest route.

To avoid being hit by any cars on his bike ride, PNPenguin wants to find which roads definitely won't be traveled by any Penguinland drivers on their morning commutes. Write an efficient algorithm to find this list of roads in $O((N+M)\log(N))$.

Please provide a 3-part solution.

6 [Coding] Shortest Paths and Huffman Encoding

For this week's coding questions, we'll implement some shortest paths algorithms, as well as huffman encoding. There are two ways that you can access the notebook and complete the problems:

1. **On Datahub**: click here and navigate to the `hw04` folder.
2. **On Local Machine**: `git clone` (or if you already cloned it, `git pull`) from the coding homework repo,

   https://github.com/Berkeley-CS170/cs170-sp24-coding

   and navigate to the `hw04` folder. Refer to the `README.md` for local setup instructions.

Notes:

- *Submission Instructions:* Please download your completed submission `.zip` file and submit it to the Gradescope assignment titled "Homework 4 Coding Portion".
- *Getting Help:* Conceptual questions are always welcome on Edstem and office hours; *note that support for debugging help during OH will be limited.* If you need debugging help first try asking on the public Edstem threads. To ensure others can help you, make sure to:

  1. Describe the steps you've taken to debug the issue prior to posting on Ed.
  2. Describe the specific error you're running into.
  3. Include a few small but nontrivial test cases, alongside both the output you expected to receive and your function's actual output.

  If staff tells you to make a private Ed post, make sure to include *all of the above items* plus your full function implementation. If you don't provide them, we will ask you to provide them.
- *Academic Honesty Guideline:* We realize that code for some of the algorithms we ask you to implement may be readily available online, but we strongly encourage you to not directly copy code from these sources. Instead, try to refer to the resources mentioned in the notebook and come up with code yourself. That being said, we **do acknowledge** that there may not be many different ways to code up particular algorithms and that your solution may be similar to other solutions available online.