HW 9

Due on **Monday, April 6, 2020 at 11pm**

CS/ECE 374 B: Algorithms & Models of Computation, Spring 2020 Version: **1.01**

Submission instructions as in previous homeworks.

25 (100 PTS.) COVID-19 II

Since last week's homework, the number of corona virus cases has increased from 370,000 to more than 800,000; more than double within one week. At this rate, tens of millions will contract the virus before the end of the semester. Please stay at home and avoid social contact.

As we saw in last week's homework, graphs can be useful in analyzing and tracking the spread of viruses. Suppose you are given a graph G with n vertices and m edges and a parameter k. Each vertex represents an individual and each edge represents a social connection that can lead to a virus transmission. For every individual/vertex $v \in \mathsf{V}(\mathsf{G})$, you also given a value $C(v)$ which is "$+$" if the individual tested positive for the corona virus, and "$-$" otherwise.

An individual is ***at risk***, if he or she already tested positive for the virus or there are at least k other distinct individuals who tested positive for the virus and can reach the individual, i.e., there are paths in the graph from at least k vertices that tested positive to the vertex of interest (here, think about k as being a small number compared to n).

25.A. (50 PTS.) For the case that G is a DAG, describe an algorithm, as fast as possible, that finds all the ***at risk*** individuals. What is the running time of your algorithm? Argue briefly that your algorithm is correct.

25.B. (50 PTS.) For the case that G is a general directed graph, describe an algorithm, as fast as possible, that finds all the ***at risk*** individuals. What is the running time of your algorithm? Argue briefly that your algorithm is correct.

Note that a simple solution to both parts would be to reverse all the edges of the graph, find the reach of each vertex v using graph search and determine if there are k vertices that tested positive and can be reached from v. However, such a solution is $O(n(n+m))$ and is worth 0 points.

26 (100 PTS.) Taking the Bus

You decide to take the bus to Walmart from home. Since you planned ahead, you have a schedule that lists the times and locations of every stop of every bus in Champaign-Urbana. It is extremely cold so you'd still like to spend as little time waiting at bus stops as possible. Unfortunately, there isn't a single bus that visits both Walmart and your home; you must transfer between buses at least once.

Describe and analyze an algorithm to determine a sequence of bus rides from your home to Walmart, that minimizes the total time you spend waiting at bus stops. You can assume that there are b different bus lines, and each bus stops n times per day. Assume that the buses run exactly on schedule, that you have an accurate watch, and that it is too cold to even contemplate walking between bus stops.

27 (100 PTS.) **Fluctuations**

We are given a directed graph with n vertices and m edges ($m \geq n$), where each edge e has a weight $w(e)$ (you can assume that no two edges have the same weight). For a cycle C with edge sequence $e_1 e_2 \cdots e_\ell e_1$, define the *fluctuation* of C to be

$$f(C) = |w(e_1) - w(e_2)| + |w(e_2) - w(e_3)| + \cdots + |w(e_\ell) - w(e_1)|.$$

27.A. (20 PTS.) Show that the cycle with the minimum fluctuation cannot have repeated vertices or edges, i.e., it must be a simple cycle.

27.B. (80 PTS.) Describe a polynomial-time algorithm, as fast as possible, to find the cycle with the minimum fluctuation.