HW 9

Due on **Wednesday, November 11, 2020 at 10am**

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

Submission instructions as in previous homeworks.

Any dynamic programming solution should be done using an iterative algorithm.

25 (100 PTS.) OLD Homework problem (not for submission):

Rainbow walk

We are given a directed graph with n vertices and m edges ($m \geq n$), where each edge e has a color $c(e)$ from $\{1, \ldots, k\}$.

25.A. (20 PTS.) Describe an algorithm, as fast as possible, to decide whether there exists a closed walk that uses all k colors. (In a walk, vertices and edges may be repeated. In a closed walk, we start and end at the same vertex.)

25.B. (80 PTS.) Now, assume that there are only 3 colors, i.e., $k = 3$. Describe an algorithm, as fast as possible, to decide whether there exists a walk that uses all 3 colors. (The start and end vertex may be different.)

26 (100 PTS.) OLD Homework problem (not for submission):

Stay safe

We are given an *undirected* graph with n vertices and m edges ($m \geq n$), where each edge e has a positive real weight $w(e)$, and each vertex is marked as either "safe" or "dangerous".

26.A. (35 PTS.) Given safe vertices s and t, describe an $O(m)$-time algorithm to find a path from s to t that passes through the smallest number of dangerous vertices.

26.B. (65 PTS.) Given safe vertices s and t and a value W, describe an algorithm, as fast as possible, to find a path from s to t that passes through the smallest number of dangerous vertices, subject to the constraint that the total weight of the path is at most W.

27 (100 PTS.) OLD Homework problem (not for submission):

Stay stable

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. (10 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. (90 PTS.) Describe a polynomial-time algorithm, as fast as possible, to find the cycle with the minimum fluctuation.