Algorithms, Fall 2019-20, Homework 5
due Sunday, November 17, 2019, 11:59pm

Problem 1

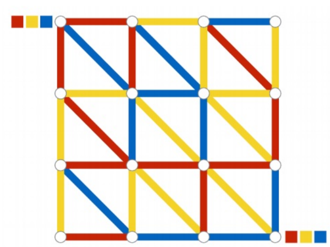
Given is an undirected graph G that represents a map of a town: each vertex corresponds to an intersection and each edge corresponds to a (two-way) road between two intersections. After living happily in this town for years, its inhabitants suddenly realized that they cannot get to all the places in their town! Help them to find the smallest number of roads they need to build to be able to go from any intersection to any other intersection in this town. In other words, find the smallest k such that, after adding k edges to the graph, it becomes connected. Design an $O(m+n)$ algorithm for this problem. (Recall that n is the number of vertices and m the number of edges in G.)

Problem 2

Given are n courses and for each course given are its prerequisites. Let P_i be the set of prerequisite courses for the i-th course and let $m = |P_1| + |P_2| + \ldots + |P_n|$. Give an $O(n+m)$ algorithm that finds the size of the longest prerequisite chain, i.e., the longest sequence of courses for which for every element in the sequence the previous element is its prerequisite. You may assume that the data is consistent, i.e., there are no "prerequisite loops."

Problem 3

Given is a colorful maze such as the one shown here:



To pass through the maze, you start at the top left corner and want to get to the bottom right corner, and as you pass through the edges, they have to alternate colors in this order: red, yellow, blue. (That is, you have to start with a red edge, end with a blue edge, and after each red edge you need to take a yellow edge, after each yellow edge you need to take a blue edge, and after each blue edge you need to take a red edge).

We will represent the colorful maze as an undirected graph G, where each of its edges has been colored by one these colors: red, yellow, or blue. Given are also two of G's vertices s and t. Design an $O(m+n)$ algorithm that will find the smallest number of edges one needs to take to go from s to t while obeying the rules of the maze.

Problem 4

In a different town, full of one-way streets, the town council suddenly realized that it is not possible to get from every location to every other location. They want to fix this but their budget is limited – they can build a single new road. Help them!

Given is a directed graph G (where vertices represent intersections and edges one-way streets). Determine if there exists a pair of vertices u and v such that adding an edge from u to v makes the graph strongly connected. Your algorithm should run in time $O(m+n)$.

Problem 5

Given is a weighted undirected graph $G = (V, E)$ and a subset of its edges $F \subseteq E$. An *F-containing spanning tree of G* is a spanning tree that contains all edges from F (the spanning tree might contain other edges as well). Design an algorithm that finds the cost of a minimum-cost F-containing spanning tree of G. Your algorithm should run in time $O(m \log n)$ or $O(n^2)$.