Homework #11

You should try to solve these problems by yourself. I recommend that you start early and get help in office hours if needed. If you find it helpful to discuss problems with other students, go for it. **You do not need to turn in these problems. The goal is to be ready for the in class quiz that will cover the same or similar problems.**

Problem 1: Minimum Spanning Trees

Let us say that a graph $G = (V, E)$ is a *near-tree* if it is connected and has at most $n + 8$ edges, where $n = |V|$. Give an algorithm with running time $O(n)$ that takes a near-tree G with costs on its edges and returns a minimum spanning tree of G. You may assume that all of the edge costs are distinct.

Problem 2: Bottleneck Edges in Minimum Spanning Trees

One of the basic motivations behind the Minimum Spanning Tree Problem is the goal of designing a spanning network for a set of nodes with minimum *total* cost. Here we explore another type of objective: designing a spanning network for which the *most expensive* edge is as cheap as possible.

Specifically, let $G = (V, E)$ be a connected graph with n vertices, m edges, and positive edge costs that you may assume are all distinct. Let $T = (V, E')$ be a spanning tree of G; we define the *bottleneck edge* of T to be the edge of T with the greatest cost.

A spanning tree T of G is a *minimum-bottleneck spanning tree* if there is no spanning tree T' of G with a cheaper bottleneck edge.

(a) Is every minimum bottleneck tree of G a minimum spanning tree of G? Prove or give a counter example.

(b) Is every minimum spanning tree of G a minimum bottleneck tree of G? Prove or give a counter example.

Problem 3: Computing the maximum flow

Consider the graph $G = (V, E)$ with edges $e = (u, v) \in E$ and capacities $c(e)$ shown in the figure below. The capacity $c(e)$ is annotated for each edge $e \in E$, v_0 is the source, and v_5 is the sink.

Compute the maximum flow for G.