

Minimum Spanning Trees

Hello, welcome to competitive programming.

Today we are going to talk about minimum spanning trees and their variations.

Objectives

...

In this video we are going to talk about Minimum Spanning Trees in general, what some of their properties are, and some interesting variations that can occur. Actual algorithms to compute them will be in other videos.

An Example

...

As a running example, consider the following connected, weighted graph. Let's say that the edges represent a cost, and so we would like to delete some edges to reduce the cost, but we would also like to keep the graph connected.

So, in this graph you see the most expensive edge is a 10, so maybe we could delete that one. And then the 9 could also be safely deleted. Keep it up, and what you get is this...

(next)

The highlighted edges form what is called a spanning tree. And this particular spanning tree is minimum for the graph.

It's easy to imagine the kinds of problems you might want to solve. The nodes could represent villages and the edge weights the cost of building a road between them,

and you want to connect all the villages. The villagers in f might be very upset when they have to travel to c , but at least the government saved money on roads.

Properties of MSTs

There are a few properties of MSTs that you must know. Every connected graph is guaranteed to have at least one minimum spanning tree. It is possible for a graph to have many MSTs, especially if edge weights repeat.

Since the MST is a tree, the number of edges will be equal to the number of vertices minus one.

Also, since it is a tree, removing any edge will disconnect it.

Variations

There are some interesting variations of MST problems. The easiest variation is to find a maximal spanning tree. We just select maximal edges instead of minimum.

Another variation is called a minimum spanning subgraph. Sometimes you are given a problem where some edges are added already and you must connect the rest of the graph. The result is a subgraph that is sort of minimum; perhaps we can say it's the minimum subgraph that follows the constraints.

If the graph is not connected, you will end up creating a minimum spanning forest.

Finally, one thing that could happen is you would be asked to find the second most minimum spanning tree. The approach for this is to find the actual MST, and then make $(|E|)$ runs, each one recomputing the MST with out one of the original edges.

In contests, problem setters will try to disguise the fact that a problem can be solved using MST. So as always, it is important to contest strategy to understand the structure of the problem you are trying to solve.

In the next videos we will go over some MST algorithms.

Next
Transcript — Points, Lines, and Vectors →

Copyright © 2019 - Mattox Beckman

