

---

**CS 302, Spring 2007**  
**Assignment #7**  
**Paired Power Project**  
**(or fun with MSTs)**

Due 5/1/07, in class

**Introduction**

A windstorm has hit Maryland and you are on the spot. Power lines are down all over the state, repair crews are standing by, and as chief programmer, you are responsible for deciding where to send the repair crews. Time is critical!

You know the state of the power grid before the storm hit and you just received the final set of damage reports indicating which lines were lost; what do you do?

Don't panic. Just grab an algorithm text off the shelf, look up a minimum spanning tree routine and get it running. Consider the power grid a graph of nodes, with one power plant and lots of consumers like schools, homes, and businesses. The power grid will be a set of edges between the nodes (so we have the unrealistic model of power lines jumping from building to building.) If the graph is connected, everyone is receiving power. If it is disconnected then only those in the component with the power plant have power.

To start, you will input nodes and edges to represent the grid under normal conditions. Initially the graph should be connected (and you should test this) or someone screwed up. Then the storm hits and damage reports pour in. Each report indicates a lost line and the estimated time to repair. Once all the damage reports are in, you run connected components again to list the consumers without power (and alert public relations the mess they're in). Finally, programming to the rescue as you run a minimum spanning tree to list those edges that will reconnect the network in the minimum time.

You will be given three input files. The first file lists the buildings in your graph. You are only responsible for central Maryland, which we will assume covers the area between  $38.9^\circ$  and  $39.4^\circ$  latitude, and  $76.4^\circ$  and  $76.9^\circ$  longitude. Each line will contain a building name, a location with latitude followed by longitude, and the number of occupants in the building. You may assume the power plant is given the name "Power-Plant". A second file will indicate power lines by listing two buildings on the same line if there is a power line between the buildings. Finally, the third file will list downed power lines by the buildings at the end points and repair times in minutes.

**The Assignment** (to be done in groups of 3, but not the person you paired with for Assignment 6)"

- (1) Your program should take three files described above as command line arguments. (Report the correct usage if the names are not given).
- (2) Be sure to check for a connected graph (exit, with appropriate error message, if the input is not a single connected component).
- (3) Analyze the report of downed power lines.
- (4) When you choose your MST algorithm, weigh the pros and cons of each algorithm and decide on the most efficient.
- (5) Report on the customers without power by listing the building names.
- (6) Then report the lines that must be fixed (including the time to restore each line) in order to restore power to all customers in the least amount of time.
- (7) Output the number of minutes required to restore power.
- (8) A large grid of about 10,000 buildings with power lines and outages will be posted by May 3<sup>th</sup>. Once your program works on some small grids, try it out on the large one, and report the number of lines that must be fixed and the time it will take to fix them.

**What to hand in**

- (1) (by 4/20)  
A one-page design.  
A black-box test plan.
- (2) (by 5/7) A 2up well formatted listing of your code.
- (3) (by 5/7) Email me a single tar file that includes
  - the file `Inventory`
  - all your source files
  - your makefile
  - any auxiliary files that you have accumulated
  - the output from executing your program on this test plan (number each test).
  - a discussion of your chosen MST algorithm
  - the statistics for fixing the large power grid

**Notes**

- (1) Work in groups of three, but not the same person you paired with for Assignment 6.
- (2) Sample files `buildings.txt`, `powerGrid.txt`, and `downedLines.txt` are on the course web page.
- (3) Shamelessly stolen from Dr. Eastman!