

## Programming Project 2

Due **12/2/10** before class

### Graphs (30 points)

Implement Prim's MST algorithm as well as one of the following algorithms: BFS, DFS, Topological Sort, Kruskal's MST (with any implementation to work around the union/find). Use the adjacency lists representation to store your graphs.

Test your algorithms on test graphs that you specify (hard-coding is fine). Your test graphs need to test non-trivial cases of your algorithms and they have to contain at least 10 vertices. You may assume for convenience that the vertices of your graph are labeled 0, 1, 2, 3, .... Write a short report (ideally a Word document; but if you know how to use LaTeX that's even better) in which you document your test cases with pictures: Each test case should have a picture of a graph, and explain why the result of your program is the correct output of the algorithm. The tests, including the report, will be worth 10 points.

- You are allowed to turn in this programming project in groups of two.
- You can use Java, C, or C++ for this project. If you want to use a different programming language, check with our TA first.
- **The name of your project directory should be**  
`project2_<lastName1><firstName1><lastName2><firstName2>`
- Zip up a directory with your entire project (source code and report). Turn in the zip file by emailing it to me (carola@cs.utsa.edu) with the subject "CS3343 project 2 submission".
- All projects need to compile. If your program does not compile you will receive 0 points on this project.
- Do not use any fancy libraries. We should be able to compile it under standard installs of Java, C, or C++ under linux and/or windows. You may want to include some comments how you compiled the project.