

Computer Science 271  
Project 7  
Due Wednesday, December 8

---

*You will complete this project with a partner. Both individuals are expected to contribute equally to all parts of the project.*

1. Write a vertex class and a weighted, directed graph class named **Graph**. (The class can also represent an undirected graph if, for every edge  $(u, v)$  there is also an edge  $(v, u)$ .) Your graph class should be based on an adjacency list representation but may also contain a matrix of edge weights. Each vertex should have a unique integer identifier between 0 and  $n - 1$ . This identifier will be used as the index of the vertex in the adjacency list. The vertex class also needs to contain any attributes used by algorithms implemented in your graph class (see below).

A constructor for your graph class should take a filename as a parameter, and initialize the object to be the graph represented by that file. The file will contain the number of vertices in the graph followed by an adjacency matrix representation of the weighted graph. For example, the following file represents a particular weighted, undirected graph with 5 vertices:

```
5
0 1 2 0 0
1 0 1 1 0
2 1 0 0 3
0 1 0 0 1
0 0 3 1 0
```

As usual, include pre/postconditions, throw exceptions where appropriate, and include unit tests. Also, remember to implement the copy constructor, destructor, and assignment operator.

2. Add a method

```
void Graph::dfs()
```

to your graph class that performs a depth-first search of the graph. Your method should print the vertices in the order they are visited.

3. Add a method

```
bool Graph::cycle()
```

to your graph class that indicates whether the graph contains a cycle.

4. Add a method

```
void Graph::Prim(int root)
```

to your graph class that uses Prim's algorithm to construct a minimum spanning tree of an undirected weighted graph, starting from the vertex labelled **root**. The method should simply print the edges in the MST. You will need to use your previously written minimum priority queue template class.

Submit all of your individual source files and a single PDF containing your code, following the instructions in the previous projects. Be sure to indicate the names of all group members on all of your submitted files.