Problem 1

```
Graph ADT Specifications
Contructor (no paramenters)
    effects Constructs a new Empty graph with empty node_list and edges

Constructor (with parameters)
     param nodeList The ArrayList of Nodes to be added.
     param edgeList The ArrayList of Edges to be added (Start, End, Weight).
     requires nodeList of type ArrayList<String>
     requires edgeList of type Arraylist<Edge>
     effects Constructs a new graph based off of the parameters given.

addNode
**Takes in a list of new nodes to add, if only one, then list will have single element
     param new_nodes The ArrayList of new node(s) to be added
     requires new_nodes be of type ArrayList<String>
     requires new_nodes.size() >= 1 (if 0, then list does not change)
     modifies node_list will increase in size, numVertices will increase in value
     effects The graph will contain more nodes than before the method was called.

addEdge
**takes in list of edges to add (if no node is present, then it makes the node)
     param new_edges Arraylist of all of the new edges to add
     requires new_edges be of type ArrayList<Edge>
     requires new_edges.size() >= 1 (if 0, then list does not change)
     modifies edge_list will increase in size, numEdges will increase in value
     modifies node_list may increase in size and numVertices may also increase in size
     effects The graph will contain more edges (and potentially more nodes).

ListNodes
** Returns a iterator to a list of all the nodes in sorted order
     effects creates a new sorted arrayList of the nodes
     returns An iterator pointing to the first element of the sorted list of vertices


ListEdges
** Returns a iterator to a list of all the edges
     returns An iterator pointing to the first element of a list of edges


CheckRep()
** Ensures the invariant holds
     throws RuntimeException if any of the conditions are not met.
```