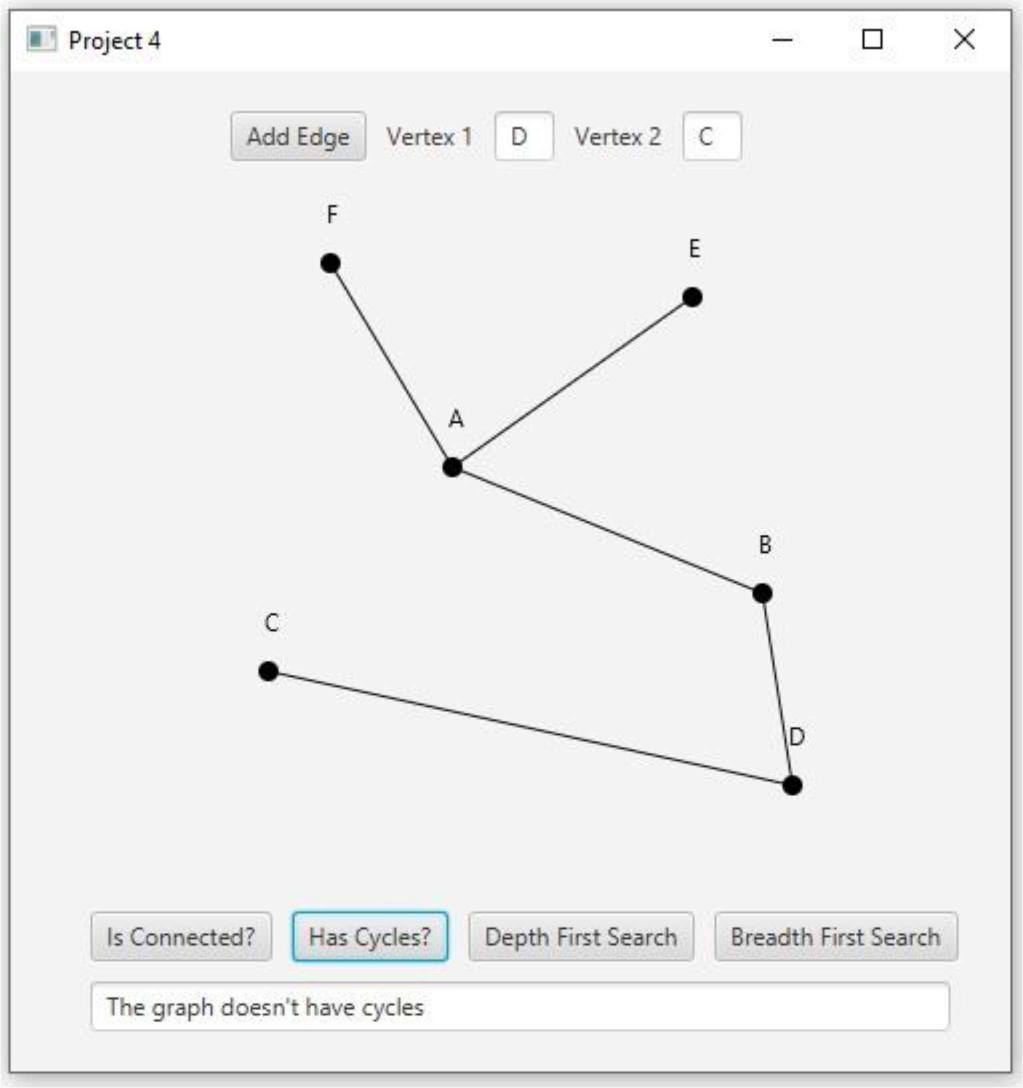
CMSC 315 Data Structures and Algorithms
Programming Project 4

The fourth programming project involves writing a program that allows the user to create an undirected graph that is visually displayed and to check whether the graph is connected and whether it has cycles in response to button clicks. In addition, it should provide buttons, when clicked, display a depth-first or breadth-first graph search. The GUI for the program is shown below:



Vertex 1 D Vertex 2 C

The graph doesn't have cycles

.

The user should be able to add vertices to the graph by left clicking the mouse at the desired location. The first vertex created should be labeled *A* and subsequent vertices labeled the next letter of the alphabet. Edges are added by supplying the vertices in the text fields labeled *Vertex 1* and *Vertex 2* and then clicking the *Add Edge* button. If the user enters a nonexistent vertex name, an error message should be displayed in the text field at the bottom.

The four buttons at the bottom should analyze the graph. The *Is Connected* button should indicate whether the graph is connected. The *Has Cycles* button should indicate whether the graph has cycles. In both cases, an appropriate message should be displayed in the text field at the bottom. The *Depth First Search* and the *Breadth First Search* buttons should display the list of the vertices resulting from the corresponding search, starting at vertex A.

The program should consist of four classes. The first class should be an immutable class that defines a vertex of the graph and contains the *x* and *y* coordinates of the vertex along with its name. It should have a constructor and three getter methods.

The second class should be a class that defines the graph. You may use any graph representation that allows an unlimited number of vertices and edges. It should have the following public methods

- A method to add a vertex
- A method to add an edge
- A method that checks whether the graph has cycles
- A method to check whether the graph is connected
- A method that returns a list of vertices resulting from a depth-first graph search
- A method that returns a list of vertices resulting from a breadth-first graph search

You may add any other methods as needed.

The third class is should be an extension of the `javafx Pane` class that visually displays the graph. It should contain an event handler that responds to mouse clicks that creates new vertices and a method that is called to draw edges.

The fourth class should contain the main method and should create the GUI including all the buttons and text fields. It should include event handlers for each of the buttons.

You are to submit two files.

1. The first is a `.zip` file that contains all the source code for the project. The `.zip` file should contain only source code and nothing else, which means only the `.java` files. If you elect to use a package, the `.java` files should be in a folder whose name is the package name. Every outer class should be in a separate `.java` file with the same name as the class name. Each file should include a comment block at the top containing your name, the project name, the date, and a short description of the class contained in that file.
2. The second is a Word document (PDF or RTF is also acceptable) that contains the documentation for the project, which should include the following:
   a. A UML class diagram that includes all classes you wrote. Do not include predefined classes.
   b. A test plan that includes test cases that you have created indicating what aspects of the program each one is testing
   c. A short paragraph on lessons learned from the project