

CS 339 Homework 5

(due on the class on 11/30/15)

1 (20 pts) Part I

In this part, you will learn how to draw graphs using a Java Library on Graph, named *JGraphT*. First, you will learn how to configure a Java Project in an IDE (e.g., eclipse, JGraph, etc.) so that you can import the classes implemented in the JGraphT library. Second, you will need to read sample codes and documents before you draw graphs. Two sample Java files are available in the Blackboard to let you start. JGraphT library is also available in the Blackboard. Documentation is available in the following webpage: <http://jgrapht.org/javadoc/overview-summary.html>. Last, draw the following two undirected weighted graphs, and save each of them as a jpeg file. (You may use print-screen key and the Paint software to save a printed screen image.)



Figure 1: Two undirected weighted graphs.

2 (80 pts) Part II

In this part, you will implement the Dijkstra's algorithm to find a shortest path between two specified vertices on a given weighted undirected simple graph. You can assume that the weight of an edge is an integer. You may either use Graph data structures defined in JGraphT or generalize a 0/1 adjacency matrix to an integer adjacency matrix to represent a weighted undirected simple graph.

Test the correctness of your code using two graphs illustrated in the above figure. In both cases, vertex a is the start vertex, and vertex c is the terminal vertex. Your code should print the shortest path as a sequence of vertices, in addition to the weight of the path. For example, your code should print

a, b, c 2

Give an analysis of the time complexity of your code in terms of big-O notation.

On submission:

Submit your source code and two jpeg files of drawn graphs to blackboard.