

## CSC 340 HOMEWORK 2

PROFESSOR GODFREY MUGANDA

You may discuss approaches to the problem with other students in the class, or with the professor. Do not post on any online site asking for help, and do not search the Internet to solutions for these problems.

1. Implement the algorithm for the *all pairs shortest paths* problem studied in class. The input for problem is a file containing the following data:

- (a) A number  $n$  designating the number of vertices in the digraph on the first line.
- (b)  $n$  lines, with each line containing  $n$  integers separated by whitespace. These numbers are the weights in the adjacency matrix of the input graph. Use 0 for the weight of an edge from a vertex to itself (self loop) and -1 for the weight of a non existing edge.

The output from your program should be

- (a) the original adjacency matrix,
- (b) the final matrix showing the lengths of the shortest path between every ordered pair of vertices.
- (c) the matrix of “pass-through” vertices for each ordered pair of vertices,

Once these data items have been output, your program should loop, asking the user to enter pairs of vertices  $i$  and  $j$  on a line. For each such pair, the program prints the length of a shortest path from  $i$  to  $j$ , followed by the sequence of vertices starting with  $i$  and ending with  $j$  that represents such a path. The program terminates when the user enters -1 and -1 when prompted for  $i$  and  $j$ .

Turn in a well-documented program, in Java or C++, or C#, that implements your solution.

Do Problems 6.1, 6.2, and 6.3 in the course textbook. No computer implementation is necessary. Turn in a pdf or  $\text{\LaTeX}$  file with your solution. Make sure your algorithms are neatly written and unambiguous.

For each problem, make sure you maintain whatever information is needed to recover the solution, and then give an algorithm that actually recovers the solution.

This homework is due Monday of Week 5 at midnight.