BBM204

Take-Home 3

Submission Date: 23.04.2020

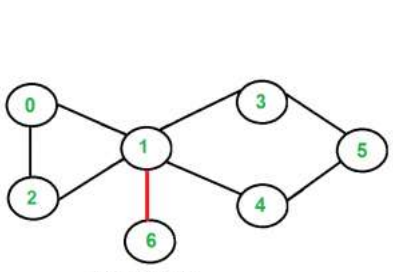
Due Date: 26.04.2020 23:59

Programming Language: Java

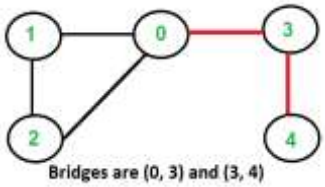
QUESTIONS

1. In a graph, if two or more edges that join the same pair of vertices, then those edges are called parallel edges. Write a linear-time algorithm to count the parallel edges in a ***undirected graph*** which is built by graph's input.txt .

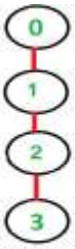
2. An edge in an undirected connected graph is called as a bridge that, if removed , would separate a connected graph into two disjoint subgraphs or make different components of the graph. A graph is 2-edge connected if it contains no bridges. Design and write a DFS-Based program for finding bridges and determing whether a given graph is edge connected. Some example graphs with bridges highlighted with red color are shown below. Your program should read input.txt, print bridges or print "Graph is two-edge connected" if graph is edge connected.



Bridge is (1, 6)



Bridges are (0, 3) and (3, 4)



Bridges are (0,1), (1,2) and (2,3)

3. Given a digraph G, Design and write an algorithm to determine whether a digraph has a unique topological ordering. If there is a Hamiltonian path, the topological sort order of this digraph will then be unique. Otherwise, if digraph has more than one valid topological orderings, then topological sort does not form a Hamiltonian path. In this case, it is always possible to form a second valid ordering just by swapping two consecutive vertices which are not connected by an edge to each other.

input1.txt	input2.txt	input3.txt	output.txt
4 4 0 1 1 2 2 3 3 0	7 8 0 1 0 2 1 5 2 1 3 6 3 4 4 6 5 6	9 9 0 1 1 2 1 3 4 5 5 6 6 8 6 7 7 2 8 3	Q1 0 Q2 5-6 1-5 Q3 False

Figure 1: Input files and their output

NOTES

- Your input files and output file format should be as shown in Figure 1. Input file has count of vertices and edges in the first and second line respectively. This lines followed by graph matrix. The vertex number arrival sequence is consecutive and the first index starts from 0.
- You should read three txt file that you will use to create a graph such as input1.txt, input2.txt and input3.txt respectively for each question from commandline, and you should print the output of each question into the output.txt file as shown in Figure 1. For testing your program command-line parameters should be as shown below:

  >javac main.java

  >java main <input1.txt> <input2.txt> <input3.txt> <output.txt>

- No late submission is permitted.
- It is not an obligation but we suggest Eclipse platform since it is free and it gets updates constantly.
- Use submit system that has been configured for you. No other medium except Submit System will be accepted.
- Use UNDERSTANDABLE names for your variables, functions, and classes (please be sure you are obeying name convention).
- Write READABLE SOURCE CODE blocks.
- Use EXPLANATORY COMMENTS in your source codes.
- Your work must be original. Duplicate or very similar works are both going to be considered as cheating.
- **You can use the input specifications and graph classes you use in the course.**
- You will submit your work from https://submit.cs.hacettepe.edu.trwith the file hierarchy as below:

Src|--

|-- main.java

|-- .java* (optional)

|-- output.txt