CS F426 Graph Mining Lab - 1 Readme

- There are 5 questions given to implement as part of lab-1 with the two files containing edges of the graph as test inputs.
- For the questions containing sub-parts, each part is implemented in a separate code file.
- All the questions are implemented in the C++ Programming language.
- The naming convention is done according to the question number and the subpart of it. For Example, part a of q1 is named Q1_a.cpp, and this holds true for all the questions.

Instructions to run the codes:

Q1.
There are 2 parts to the question. Part A, Part B can be run in the same manner. The code is documented to the best of my ability so that each function clearly describes its functionality. To run the test cases, the file input should be given which should be in the **.txt** format without any comments in the beginning. A variable called ***file*** is initialized with graph-1.txt as of now. To test on any input file, the ***file*** variable should be initialized with the name of the desired input file.
NOTE: Make sure that the input files are in the same directory as the code of execution.
Better to run on a Linux machine with a GCC compiler.
Command line1: g++ Q1_a.cpp
Command line2: ./a.out

Command line1: g++ Q1_b.cpp
Command line2: ./a.out

Q2.
The input file should be loaded in the same way as mentioned in the Q1.
External support files/classes/headers.

1. supportLib.cpp
2. supportLib.hpp
3. pbPlots.cpp
4. pbPlots.hpp

These files help in creating the plot in an image format.
NOTE: Make sure that all the above-mentioned file(s) are in the same directory as the code of execution.
Should be run on linux machine with a gnu gcc compiler for a better plot
Command line1: g++ Q2.cpp supportLib.cpp pbPlots.cpp -lm
Command line2: ./a.out
A png file will be generated in the same directory which contains the plot of the given input graph file.

Q3.
The input file should be loaded in the same way as mentioned in Q1.
It is asked to compute the number of 3-cycles (triangle) and the 4-cycles (non-induced) in the given graphs.
It contains two parts.
No external files are needed for it to run the code and has only a single part.
Should be run on a Linux machine with a GCC compiler.

Command line1:g++ Q3.cpp
Commandline2: ./a.out
The number of triangles in the graph is printed.

Q4.
The input file should be loaded in the same way as mentioned in the Q1.
It is asked to compute the diameter of the graph.
External files (header) to be included:

1. Graph.h

This is a Graph class containing all the required constructors and methods to compute the result.
NOTE: Make sure that all the above-mentioned file(s) are in the same directory as the code of execution.
Should be run on a Linux machine with a GCC compiler.
Commandline1: g++ Q4.cpp
Commandline2: ./a.out
The diameter of the graph is printed in the command line for the corresponding input graph file.

Q5.
The input file should be loaded in the same way as mentioned in the Q1.
It is asked to compute the maximum connected component in the graph.
No external files are needed for it to run the code and has only a single part.
Should be run on a Linux machine with a GCC compiler.
Command line1:g++ Q5.cpp
Commandline2: ./a.out
The maximum connected component is printed in the terminal for the corresponding input graph file.