COMP 3761 Assignment 4

Due: Wednesday **Feb 11, 2015** at 6:30pm

Connected components of a graph [20 points]
A connected component of a graph is a maximal connected subgraph of the given graph such that for every pair of vertices u, v in the subgraph, there is an undirected path from u to v. Informally, any graph is composed of at least one connected component(s).

- Download the text input file *graph.txt*

  The file contains the edges of an undirected graph. An undirected edge (u, v) connects the vertex u to the vertex v.

  Vertices are labeled as positive integers from 1 to 875714. Each row of the file contains two numbers to indicate an undirected edge. For example, the 8th row looks liks "2 47646". This means that there is an undirected edge between vertex with label 2 and the vertex with label 47646.

- Your task is to implement the graph traversal algorithms of Depth-First-Search (DFS) and Breadth-First-Seach to compute the connected components of the given graph. Run your implementation of both DFS and BFS on the given graph.

- Identify the sizes of 5 largest connected components of the given graph.

  Output: Display the sizes (i.e. the number of vertices) of the 5 largest connected components in the given graph, in decreasing order of sizes. If your algorithm finds less than 5 connected components, then write 0 for the remaining terms. For example, if your algorithm finds only 3 connected components whose sizes are 400, 300, and 100, then your answer should be "400,300,100,0,0".

  WARNING: This is probably the most challenging programming assignment in the course. Because of the size of the graph you may have to manage memory carefully for your implementation. Do not waste the memory to store duplicate data. Do not make unnecessary copies of the graph data.

  Note: You can directly use the given input file name in your code, but do NOT use any absolute file path. Simply copy the file to the same directory where your java code is located for your test.

Bonus 5 points: For those who want to have additional challenge in this assignment, modify your DFS algorithm so that it also finds the largest cycle in the given graph. What is the number of vertices in the largest cycle (if there is any)?