CSC 222: Programming Project # 3: Graphs

Instructor: V. Paúl Pauca

due date: April 24

For this assignment, you can write code to answer the questions in each section in the computer language of your choice. You can use whatever online resources you want, however, you must write your own graph codes. You are not allowed to use existing code from the web or existing graph packages. Answer the questions below in a small report but also provide your code so that I can reproduce your results.

1 Facebook Interactions (undirected graph)

From Netvizz Facebook App: Netvizz can extract two types of social networks from groups (groups are not pages!) you are a member of (users are anonymized):

- friendship connections (API limits for group data are changing regularly, current version should be able to get up to 5000 group members. This may take a very long time, i.e. hours).
- interactions (if a user comments or likes another user's post, a directed link is created - currently the last 200 posts are take into account)

`http://csweb.cs.wfu.edu/ pauca/csc222/FBdata.txt`

File `FBdata.txt` contains a Facebook interaction social network extracted using Netvizz. Answer the following questions regarding this network:

1. *(10 points)* Which user has the largest number of interactions?
2. *(10 points)* What is the average number of interactions between users in the group?
3. *(30 points)* There is one subgroup of users that interacted the most among each other. How many users are there in this subgroup?

2 WFU WWW Network (directed graph)

A web crawler created by Cleve Moler's is used to create an adjacency graph from a given root URL. The code simply looks for the string `http:` within the root URL and follows Web links until a graph with n nodes is formed.

`http://csweb.cs.wfu.edu/ pauca/csc222/cswebData.txt`

File `cswebData.txt` contains a graph of 400 nodes created using `http://csweb.cs.wfu.edu` as the root URL with Cleve Moler's web crawler code. Answer the following questions regarding this graph:

1. *(10 points)* What web page has the largest number of links within it?
2. *(10 points)* What webpage is linked the most from other webpages?
3. *(30 points)* If there is a cycle in this graph, what is the size of the largest cycle?
4. (optional) *(10 points)* is there a path from `www.wfu.edu` to `csweb.cs.wfu.edu`? If so, can you find the path with the smallest number of links between `www.wfu.edu` and `csweb.cs.wfu.edu`?

Input file format. The `txt` files you are to download are formatted using the following very simple scheme:

- The first line specifies the number of vertices
- The next lines contain the names of each vertex
- ***** ends the vertex information
- The line after specifies the number of edges
- The next lines contain two vertex names separated by space specifying an edge from the first vertex to the second

Here is a sample file in this format (graph 3.6 from the textbook):

```
12
A
B
C
D
E
F
G
H
I
J
K
L
*****
26
A B
A E
B A
C D
C G
C H
D C
D H
E A
E I
E J
G C
G H
G K
H C
H D
H G
H K
H L
I E
I J
J E
J I
K G
K H
L H
```

What to turn in.

- Your working source code for this project and a README file explaining how to compile and use the code
- A small report with answers to the above questions