CPE 111 Programming with Data Structures
International Sections, January 2021
Laboratory Exercise 9-2

Objective
This lab is intended to give you the opportunity to work with a graph implemented using an adjacency matrix.

Instructions

Download all files from the **demos/Lecture9** area of the website. The header file **abstractGraph.h** defines an set of functions for a graph. I have already provided an implementation that uses an *adjacency list* (linked list of linked lists).

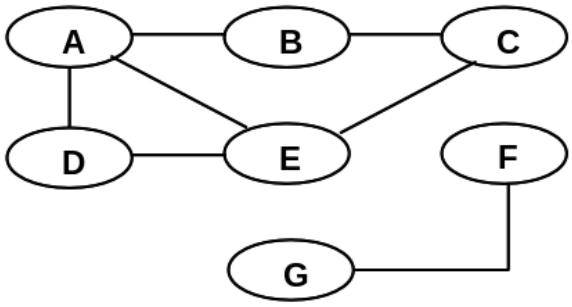
Your assignment for this lab is to create a C module called **matrixGraph.c** that uses an *adjacency matrix* to implement the graph.

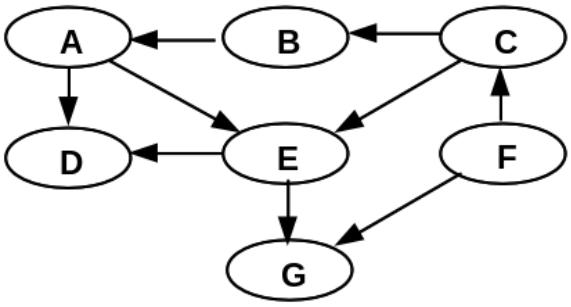
Your code must implement all the functions for which I provide code in **linkedListGraph.c**. (Exception: You do not have to implement the function **isReachable**.) You should modify the make file to build a new executable called **graphTesterM**, which links in **matrixGraph.c** instead of the linked list version.

Be sure to include code in the **addVertex** function to make sure you have not exceeded the maximum number of vertices.

Your **removeVertex** function does not need to move rows and columns of the matrix to make it compact (that is, to make sure there are no empty rows or columns), but if you don't do the moving, you need to somehow mark the removed vertices as no longer valid.

Test your new module by building and traversing the following graphs.





Your program should behave exactly like **graphTester** when linked to the other graph implementation, except that if you set the maximum number of vertices to a small number, you might get "graph is full" errors from **addVertex**. (Test that this works).

Upload the **matrixGraph.c** module and your modified version of the make file.