Data Structure Assignment 7 (Bonus)

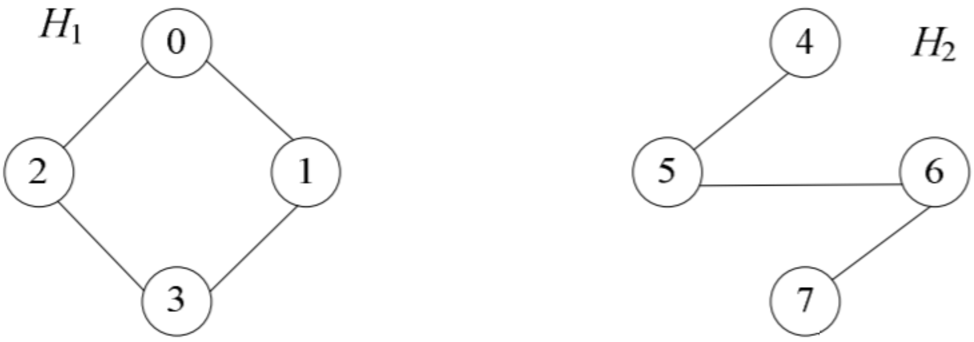
Programming homework 1

Bipartite graph

A bipartite graph, G=(V,E), is an undirected graph whose vertices can be partitioned into two disjoint sets V_1 and $V_2=V-V_1$ with the properties:

- No two vertices in V_1 are adjacent in G
- No two vertices in V_2 are adjacent in G

Graph G_4 (Textbook P270, Fig 6.5) is bipartite for example:



G_4

A possible partitioning of V is V_1= {0,3,4,6} and V_2= {1,2,5,7}.

Please write a function to determine whether a graph is bipartite. If the graph is bipartite, your function should obtain a partitioning of vertices into two disjoint sets, V_1 and V_2, satisfying the two properties listed. Show that if G is represented by adjacency list, then this function has a computing time of O(n+e), where n=|V(G)| and e=|E(G)| (| |is the cardinality of the set, that is, the number of elements in it)

Note: No point will be given if you don't use "adjacency lists".
Show that your function has a computing time of O(n+e) in your readme file

Please name your submitted file as "bipartite.c/bipartite.cpp" for this homework

Input: *//Use G_4 as input for this case*

8 ***//number of vertexes (0~7)***

7 ***//number of edges***

//input below are the edges of graph and they can be in any order

0 1

0 2

1 3

2 3

4 5

5 6

6 7

Output:

true ***//Input graph is bipartite***

0 3 4 6 ***//V_1 set***

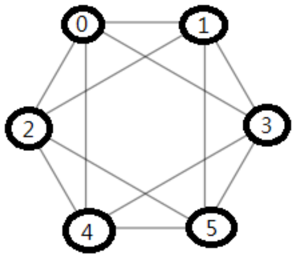
1 2 5 7 ***//V_2 set***

//Another possible bipartite partition

true

0 3 5 7

1 2 4 6



Input:

6 ***//number of vertexes (0~5)***

12 ***//number of edges***

//input below are the edges of graph and they can be in any order

0 1

0 2

0 3

0 4

1 2

1 3

1 5

2 4

2 5

3 4

3 5

4 5

Output:

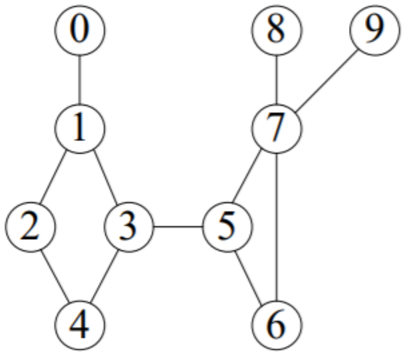
false

Programming homework 2

Bridge

An edge (u, v) of a connected graph G is a "bridge" if and only if its deletion from G produces a graph that is no longer connected. In the graph of figure below:

(Textbook P287 Fig6.19-(a)):



the edges **(0,1)**, **(3,5)**, **(7,8)** and **(7,9)** are bridges. Write a function that finds the bridges in a graph. Your function should have a time complexity of **O(n+e).**

Hint : use "bicon"(Textbook P290,program 6.6) as a starting point.

(In case you don't have a textbook. You can find program 6.6 in the course ppt.)

Note: No point will be given if you don't use "adjacency lists".
Show that your function has a computing time of O(n+e) in your readme file

Please name your submitted file as "bridge.c/bridge.cpp" for this homework

Input:

10 ***//number of vertex (0~9)***

11 ***//number of edges***

//input below are the edges of graph

//input edges can be in any order

0 1

1 2

1 3

2 4

3 4

3 5

5 6

5 7

6 7

7 8

7 9

Output:

0 1 ***//(0,1) is bridge***

3 5 ***//(3,5) is bridge***

7 8 ***//(7,8) is bridge***

7 9 ***//(7,9) is bridge***

//Output can be in any order as long as they are correct

General information:

- Deadline: 2019/12/13 23:55.
- Submit your programming assignment to Moodle system.
- Submitted file format: student-ID_Name.zip, e.g. F12345678_王曉明.zip
- Your submitted file must contain Source Code & Readme file (Program description)
- Late homework will not be accepted
- There is a "zero tolerance" for plagiarism. You will receive a score of zero if you get caught plagiarizing.

資料結構課程規定

1. 程式執行環境: Windows、Linux。
2. 程式語言: C/C++
3. 程式作業只需提供 source code 和 readme 的說明文件。Source code 只接受.cpp 和.c 檔，其餘檔案類型恕不接受，說明文件請含括您的程式內容的解說，例如，程式執 行流程，程式架構，如何設計功能等，請不要複製題目或複製程式碼註解貼上。
4. 紙本作業請列出推論過程，僅列出答案而未列出過程者，不給分。繳交紙本作 業時，記得寫名字以及學號，請務必記得用 A4 紙張作答，若超過一張，請自行裝 訂起來再交給助教，否則將斟酌扣分。
5. 紙本作業與程式作業請勿抄襲，如有發現一律 0 分計算。
6. 程式作業上傳至 moodle 各章節底下的繳交區。
7. 程式作業與手寫作業皆不接受遲交與補交。程式作業在公布之後的兩個禮拜內 將會開放上傳繳交，請同學們盡早完成作業，避免在最後期限內的一、兩個小時上 傳 moodle 導致發生問題。
8. 每個程式的程式分數佔 80%，說明文件佔 20%，若並未完成作業題目所有要求、 所交程式碼無法執行或執行結果錯誤，將會依照題目要求和執行結果的完程度評 分，其餘的評分項目由當次作業批改的助教來決定。
9. 每次作業壓縮檔(例如 zip, 7z 檔)名稱必須以學號命名，若未註明一律扣該次成績 20 分。

本課程的 TA 時段如下:
星期二 (Tues.) p.m. 3:00 – 5:00
星期四 (Thu.) p.m. 6:00 – 8:00

若是同學有資料結構相關問題可至高速網路實驗室(資訊系新大樓 5F,65503 室)詢 問，為免同學撲空，如同學無法在助教時間前來，煩請事先與助教預約時間。如有 任何問題請寫信給助教。

助教 何允中 e-mail: p76071323@mail.ncku.edu.tw

潘文傑 e-mail: p76081433@mail.ncku.edu.tw

蘇奕瑋 e-mail: p76081108@mail.ncku.edu.tw

張傑閔 e-mail: p76084499@mail.ncku.edu.tw

蘇雨菲 e-mail: p76075068@mail.ncku.edu.tw

Course Provisions

1. Program execution environment : Windows、Linux

2. Programming language : C/C++ (Languages other than C/C++ are not accepted)

3. Submitted programming homework must include source code in .cpp or .c data type, and readme document. You are required to address the **program architecture**, **program functions and how you design your program** in readme file. Do not just write the pseudo code or even just copy and paste your code!

4. You won't get any point for paper homework if you only write the answers without addressing your process and reasons. **Please do your work on A4 papers. If there is more than one page, please staple them together, and write your student id & name on each page**. Points will be deducted otherwise.

5. **There is a "zero tolerance" for plagiarism. You will receive a score of zero if you get caught plagiarizing.**

6. Please submit your programing homework to moodle.

7. Late homework is not accepted.

8. Programming homework grade is divided into two parts: 80% for the code and 20% for the readme file. **Partial points will still be awarded if the output results of your program are partly correct**. The remaining grading standards are decided by the TAs.

9. **Please name the filename of your submitted compressed file (e.g. zip, 7z) after your** student ID number. 20 points will be deducted otherwise.

TA time of the course:
Tues. 15:00 - 17:00
Thu. 18:00 – 20:00

If you have any question, please come to our lab at TA time (CSIE Bldg. Room 65503). If you are not available to come at TA time, please make another appointment with the TAs. You can also mail us about your questions.

TA	何允中	e-mail: p76071323@mail.ncku.edu.tw
	潘文傑	e-mail: p76081433@mail.ncku.edu.tw
	蘇奕瑋	e-mail: p76081108@mail.ncku.edu.tw
	張傑閔	e-mail: p76084499@mail.ncku.edu.tw
	蘇雨菲	e-mail: p76075068@mail.ncku.edu.tw