CS 4310 - Algorithms - Fall 2018

HomeWork4

Given: October 25, 2018

Due: November 9, 2019

General Comments: Read questions carefully and answer ALL parts of the question. Show all your work, otherwise no partial credit. No credit without proper justifications. State all your assumptions. No Algorithm is complete without its time and space complexity. When presenting an algorithm, first indicate if it is like a well-known algorithm, second describe intuitively how the algorithm works (which may be supported by examples), third give its pseudo code and finally analyze its time and space complexity. Always describe general idea of the algorithm before giving its pseudo-code. Do not reinvent the wheel, i.e., if a well-known algorithm can be modified to solve a problem efficiently, use that solution and clearly indicate the changes required. Do not unnecessarily complicate a solution, i.e., if a simple but efficient solution exists then we should use it. Finally, if you just write pseudo-code of a well-known algorithm without indicating how it applies or modified to the problem at hand, no credit will be given.

1. How Many Tables

Problem Description

Tomorrow is Jack & Lucy's wedding day. They have a lot of friends who will attend their wedding. Afterwards, they will have dinner together, so Jack & Lucy want to know *minimum* number of tables they need to setup. And you must consider that not all the friends know each other, and friends do not want to sit with strangers.

One rule for this is that if I tell you A knows B, and B knows C, that means A, B, C know each other (i.e., "X knows Y " is a transitive relationship), so they can sit on the same table.

For example, if I tell you A knows B, B knows C, and D knows E, so A, B, C can sit on one table, and D, E must sit on another table. Therefore, this instance needs at least 2 tables.

Input

The input starts with an integer T(1<=T<=25) which indicate the number of test cases. Then T test cases follow. Each test case starts with two integers N and M (1<=N, M<=1000). N indicates the number of friends; the friends are marked from 1 to N. Then M lines follow. Each line consists of two integers A and B (A != B), that means friend A and friend B know each other. There will be a blank line between two cases.

Output

For each test case, just output how many tables they need at least. Do NOT print any blanks.

Sample Input

```
2

5 3

1 2

2 3

4 5


5 1

2 5
```

Sample Output

```
2

4
```

First design and analyze your algorithm to solve this problem. Then implement and measure the timings. Does your theoretical analysis match what you observe? If not, why not.

More may be added…

General Instructions on submitting your homework.

- For assignments, submit a SINGLE zipped file of your source codes, scripts (to run your program if any) and a brief report along with a copy of a couple of sample executions of your solution to the class's eLearning. No need to say, but you should be using good conventions and programming practices in developing your programs [just in case you forgot, refresh them from some of the coding conventions etc. links provided on the TopicsCovered page.]
- Use <hw#cs4310_yourlastname_mmddyy. {zip, ppt, doc, tex}> as the naming convention for your zipped, ppt, MS-Word, or Latex files when submitting on eLearning. Replace '#' with the appropriate homework number.
- There will be significant point penalties for not following the naming convention above, good coding practices, submitting a different format of archive file or if your program does not run. Make sure it is a .zip and NOT another format (no .rar,.tar,.tar.gz, etc.)
- Beginning of the first file should clearly identify you, the class, submission date, and the main goals of the homework / programming assignment. Also **add credit to others if you had to resort to looking up the solution elsewhere**. Finally add one of the sentences: “I give permission to the instructor to share my solution(s) with the class." or “ I do NOT give permission to the instructor to share my solution(s) with the class."
- Attach Plagiarism-free declaration.

Any student may be asked to show and discuss his or her solution in class, so be ready with your presentation.