IT 427, Design and Analysis of Algorithms

Programming Assignment 9: Maximum Match

Due date: Nov. 18, 2024, Monday, 11:59 PM 50 points (40 on programs, 10 on report)

In this assignment, we will determine whether a given undirected graph is bipartite, and if it is, we will find the maximum number of matches between the two disjoint and independent sets of vertices. We will not use a new algorithm to solve the problem. Instead, we will *reduce* the problem to a max-flow problem.

Prepare your programs on Linux Server:

- Make your directory ∼`/IT427/asg9/`. All programs and needed files should be saved under this directory before run `submit427.sh` for submission.
- Check the contents of my `/home/ad.ilstu.edu/cli2/Public/IT427/asg9`. There are four files: `undG1.txt`, `undG2.txt`, `undG3.txt` and `undG4.txt` under the directory.

Program requirement: The name of the program should be `maxMatch.py` and I will run your program on our Linux server as follows.

```
puthon3 maxMatch.py undG1.txt
```

If your program fails to compile, you will get 0 point. In addition to the four standard input files, I will test your program on some different files that contains undirected graphs in the following format.

Input: Since undirected graphs must be symmetric, if vertices x and y are adjacent, only one of (x, y) and (y, x) is shown in the edge section. For example, in the following file, where `G1` has $(0, 2) \in E$, we just list $(0, 2)$ and omit $(2, 0)$.

```
20 Undirected Graphs in undG1.txt.

** G1:  |V|=10, V=0..9
Edges (u, v) E = {
     ( 0, 2)
     ( 0, 4)
     ( 0, 5)
     ( 0, 9)
     ( 1, 2)
     ......
     ( 7, 9)
     ( 8, 9) }
-------------------

** G2:  |V|=10 V={0..9}
     ......
     ......

** G20:  |V|=40 V={0..39}
     ......
     ......
```

Output: For each graph, if the graph is bipartite, show the maximum matches followed by a line to indicate how many pairs are there. Note that, there may be many different maximum matches and due to different implementation, the set of pairs may be different, but the maximum size must be the same. If the graph is not bipartite, indicate so. Follow the following output format:

```
Maximum number of matches in bipartite graphs in [undG1.txt]

** G1:  |V|=10
     (0, 2)
     (1, 3)
     (6, 4)
     (7, 5)
     (8, 9)
     Matches:  5 pairs

** G2:  |V|=10
     (0, 2)
     (1, 4)
     (3, 9)
     (5, 8)
     Matches:  4 pairs

     ......

** G5:  |V|=10
     Not a bipartite graph

** G6:  |V|=20
     Not a bipartite graph

     ......
     ......
```

★ Any plagiarism will receive 0 and be reported to the school ★

Submission: Programs (40 points) and Reports (10 points)

Submission details are same as the previous assignment. Run the submission script with the submission number changed to 6, but you can use the same secret name as follow:

```
bash /home/ad.ilstu.edu/cli2/Public/IT427/submit427.sh peekapoo 9
```

Note: Since I will keep updating `submit427.sh` for different assignment, you have to run the script from my `/home/ad.ilstu.edu/cli2/Public/IT427/` directly for the most recent updated version, i.e., don't copy it to your own directory.

1. Programs: 40 points. Submission on Linux server.

The score is based on the correctness and the programming style, which includes efficiency, appropriateness of data structures, and documentation of your programs. At the beginning of every program file, put a section of comments including (1) your full name, (2) student ID, (3) a pledge of honesty that you do not copy/modify from other's codes and (4) a declaration of your copyright that no one else should copy/modify the codes. You will receive:

(a) 95 $\sim$ %: No error with a good programming style.
(b) 80 $\sim$ %: Minor error and fair programming style.
(c) 60 $\sim$ %: Some error and not so good but acceptable programming style.
(d) 40 $\sim$ %: Too many error and bad programming style, but meaningful.
(e) 20 $\sim$ %: Compilable but not working and the program must show reasonable trying.
(f) 0 $\sim$ %: : Fail to meet any of aforementioned qualities or plagiarism involved.

2. Report: 10 points. Submission through Canvas.

   You have to write up a report and prepare it in pdf format. You don't have to put program output on the report as I will run and exam your program directly on some different input files.

   The report should includes brief descriptions of your program, summary of the methods, data structures, and efficiency analysis on time and space in details in terms of big-O notations. If there is any difficulties encountered in this assignment, you can report it. If your analysis is not clearly related to your program with sufficient justification, your report score will not be higher than 50%.