CMSC 401 – Fall 2024

Assignment 3
(due Sun, 11/10 – 11:59pm)

Dr. Eyuphan Bulut

CMSC 401- Algorithm Analysis with Advanced Data Structures

VCU College of Engineering

Lecture Hall Assignment

- You are the course coordinator in a university and you need to assign the courses to the lecture halls.
- You did your homework and based on the number of enrollments in each course and some other features (e.g., distance, A/V support) you know which course can potentially be taught in which lecture hall(s).
- Given these potential assignments, you want to find the maximum number of courses that could be taught at the same time.

Assignment 3

- Write a program CMSC401_A3.java that reads the database of potential assignments between courses and lecture halls in the format below:
  - The number of courses, N, in the first line. N>=3, N<=100
  - The number of lecture halls, M, in the second line. M>=3, M<=100
  - Each of the next N lines shows the course id (in increasing order from 1 to N) followed by possible lecture halls (separated by space) for that course. You can assume that
    - Each course's potential lecture halls will be in increasing order
    - There will be at most **min(20, M)** possible lecture halls for each course

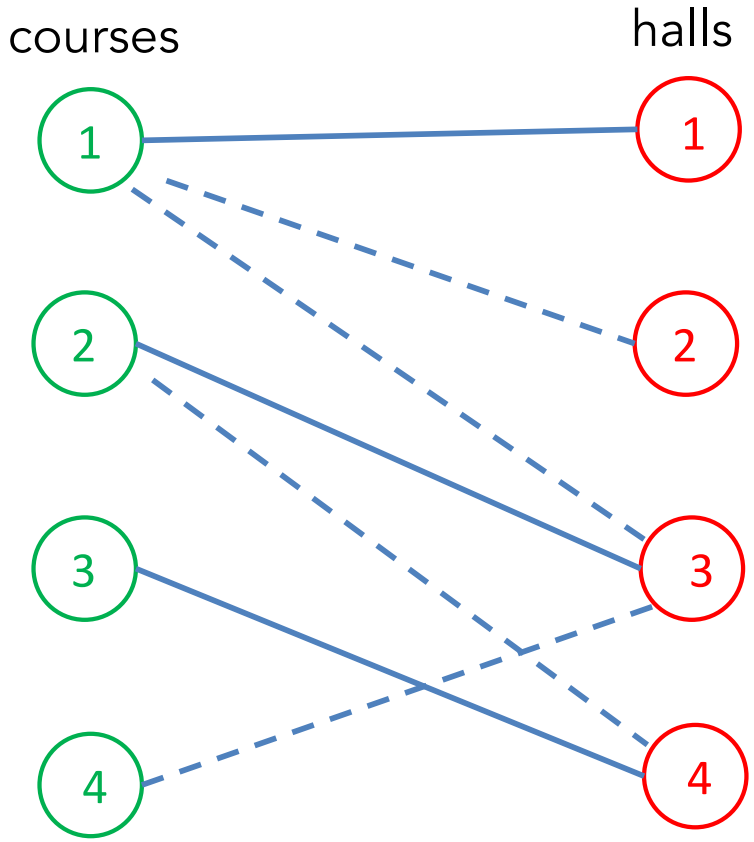
- And returns as output
  - **a single number**: the maximum number of courses that could be run at the same time
  - just one number, no comments, prompts etc.

Example #1

Input:

```
4
4
1 1 2 3
2 3 4
3 4
4 3
```

Output:
3



Solid lines show one example assignment with maximum course count

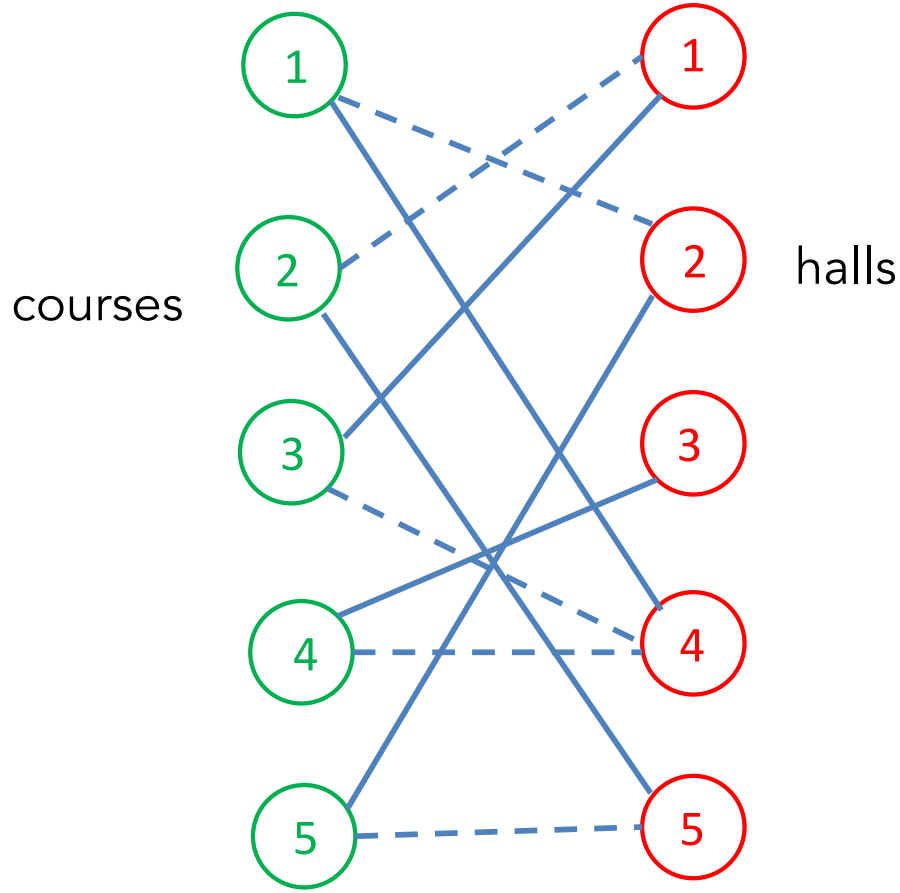
Example #2

Input:

```
5
5
1 2 4
2 1 5
3 1 4
4 3 4
5 2 5
```

Output:

5



Solid lines show one example assignment with maximum course count

Remarks

- The courses are named from 1 to N
- The lecture halls are named from 1 to M
- M does not need to be equal to N.
- Each course will have at least one potential lecture hall to be taught.
- You must implement Max-Flow algorithm (after adapting the graph) to solve it (see last slides of Lecture 18).
  - Other solutions will get max 2 even if they work.
- Any Java libraries, **classes, functions related to graphs, vertices, edges are** NOT allowed
  - Create your own (adjacency matrix or adjacency list)
- Using Java queue or priority queue (and other simple data structures such as lists, hash maps) is allowed
- No other text, comments, questions on output (you will lose points otherwise)

Submission

- Date due: Sun, Nov 10th, 11:59 pm

- Submission through Canvas
  - Just submit the single Java source code file named CMSC401_A3.java
    - **No need to zip. Don't worry about "-1", "-2" added to your file by Canvas for new versions.**
    - The file should have *your name* in a comment in the first line
    - Remember: in Java, class name should match the file name, and is case sensitive

- Please do NOT create your own packages
- Use standard I/O to read input (System.in, System.out) and output
- Make sure the **program compiles and WORKS!**
- Late submissions are accepted up to 2 days **only** with penalties!

- **Resubmission after grading:** It is allowed ONLY if you can fix your code with a minor change (1-2 lines of code change). You can resubmit only ONE time and a penalty of 0.5 points will be deducted.