

Homework 4

Design a Java code that solves the following problem.

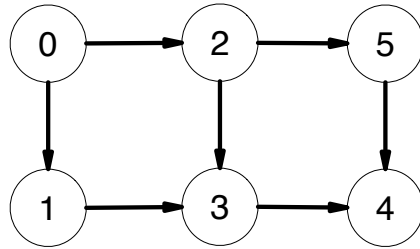
Given input directed graph G , your code should do the following:

1. Output all the vertices that have the same in and out degrees.
2. Output the average in-degree and the average out-degree of vertices.
3. Test if G has a cycle.
4. If G has a cycle you code should output at least one cycle.
5. Test if G has at most 3 cycles.
6. If the G has no cycles, then your code should output topological ordering of G .

Please note the following. You need to submit:

- Comment which data structures you use in your implementation.
- Comment what algorithms you are using in your code.
- Verify your code on the two digraphs provided in the figures below.
- Submit your compiled .jar file along with source code; your codes will be verified for plagiarism.
- Your codes will be tested on various digraphs.

The first example:



The input must be loaded from a .txt file. The contents in the file are listed as below:

1. 6
2. 0 1
3. 0 2
4. 1 3
5. 2 3
6. 2 5
7. 3 4
8. 5 4

To the input:

1. The first row represents the total amount of vertices. The total vertices are 6 in the digraph above, so the content in the first row is 6.

2. There are 7 degrees in the above digraph. So the next 7 rows represent all the degrees. For an instance, 0 1 represents that the digraph has a degree from the vertex 0 to the vertex 1.

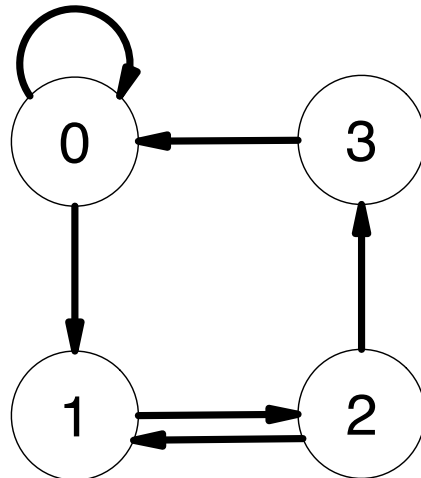
The expected output of your code must be saved in another .txt file. The contents in the file are listed as below:

1. 1 5
2. 1.16667 1.16667
3. Order:
4. 0 1 2 3 5 4
5. Yes

To the output:

1. The first row represents the vertices that have the same in-degree and out-degree.
2. The second row represents the average in-degree and out-degree of the vertices.
3. The third row is "Order:" if the digraph can be topological ordered. If not, it should be "Cycle(s):".
4. The fourth row represents the sorted topological order or a cycle. Here, the sorted topological order means that if multiple vertices have the same layer, it has to be sorted. Specifically, in this example, vertex 0 has no in-degree at the start. So 0 is the first element. Then, vertex 1 and vertex 2 are two candidates that have no in-degree. Since 1 is less than 2, so the second element must be 1, and the third element must be 2. After that, vertex 3 and vertex 5 are two candidates that have no in-degree. Since 3 is less than 5, so the fourth element is 3 and the fifth element is 5. Finally, the last element is the vertex 4. In other words, the answer will be regarded as wrong, if the topological order is 0 2 1 5 3 4 or 0 1 2 5 3 4.
5. The last row represents if the digraph has at most three cycles.

Here is the second example:



The input must be loaded from a .txt file. The contents in the file are listed as below:

1. 4
2. 0 0
3. 0 1
4. 1 2
5. 2 1
6. 2 3
7. 3 0

To the input:

1. The first row represents the total amount of vertices. The total vertices are 4 in the digraph above, so the content in the first row is 4.
2. There are 6 degrees in the above digraph. So the next 6 rows represent all the degrees. For an instance, 2 1 represents that the digraph has a degree from the vertex 2 to the vertex 1.

The expected output of your code must be saved in another .txt file. The contents in the file are listed as below:

1. 0 3
2. 1.5 1.5
3. Cycle(s):
4. 0 1 2 3 or 1 2 or 0
5. Yes

To the output:

1. The first row represents the vertices that have the same in-degree and out-degree.
2. The second row represents the average in-degree and out-degree of the vertices.
3. The third row is "Cycle(s):" because the above digraph has at least a cycle.
4. The fourth row represents at least one cycle. In the above digraph, there are 3 cycles. So you can list any one of them.
5. The last row is "Yes", because the digraph has at most three cycles.

Criteria:

How do markers compute your score?

1. You need to submit everything mentioned in page 1.
2. Your comments, verification and source code will be read and compared, to make sure the source code is from you.
3. The code will be checked by blackbox test. Five secret digraphs will be used to test on your output .txt file.
4. Each digraph takes 1 point, and the full points are 5.

Hint:

If you would like to get more points, please rigorously check if your output file has the same format as the output file explained in the two examples. Good luck!