

Advanced Algorithms (IEE5659)

Lab #1

Due: June 14, 2019 (to New E3)

Problem Formulation

Given a directed graph $G(V, E)$ and w_e denotes the weight of edge e . For each cycle $c_i \in G$,

$$w_{c_i} = \sum_{e \in c_i} \frac{w_e}{|c_i|},$$

where $|c_i|$ is the number of edges of c_i . The minimum mean cycle problem is to find the minimum w_c^* so that $\forall c_i \in G, w_{c_i} \geq w_c^*$.

Input/Output Format

Input Format:

- Each case is a directed graph.
- The first line defines the number of nodes and the number of edges.
- In the following each line, an edge is defined by a from node, a to node and a weight.
- Figure 1 gives an example.

```
2 2          //num_of_nodes num_of_edges
1 2 2        //weight of directed edge(1->2) is 2
2 1 3        //weight of directed edge(2->1) is 3
```

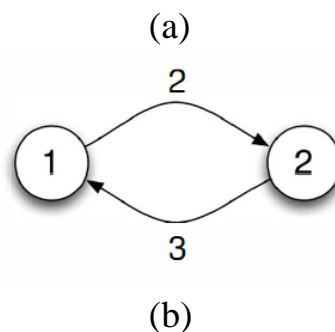


Figure 1: Input format (a) Text Format. (b) Graph Illustration

Output Format:

- If there is a cycle in the graph (example of Figure 2 (a)),
 - Write the value of minimum mean cycle in the first line.
 - Write nodes traversing along the cycle in the second line. (2 1 2 is also a correct answer in Figure 2 (a))
- If there is no cycle, (example of Figure 2(b))
 - Write “No cycle” on the first line.

```
2.50    //value of minimum mean cycle
1 2 1    //the cycle
```

(a)

```
No cycle
```

(b)

Figure 2: Output format (a) A cycle exists in graph. (b) No cycle in graph

Naming Rule and Executing Procedure

- Name of binary: **Lab1**
- Name of output file: **solution.txt**
- Execute command: “\$> ./Lab1 input.in”
(We will demo your lab on Linux environment. Please make sure your source codes can be compiled by Linux compiler(GCC).)

Grading Criteria

- Correctness of the program (no compile error, correct answer, correct output format)
- Running time (at most 30 secs)
- If all the cases are correct, you can get at least 80 points in this lab. Rest of 20 points are given by the ranking of runtime.

Submission

Please submit the following materials in a .zip file to New E3 by the deadline, specifying your student ID in the subject field (e.g., studentID_lab1.zip)

- Source codes (.cpp, .h ...)
- Executable binaries (Lab1)
- A text readme file (readme.txt), stating how to build and use your programs. If you have updated version, please name your newer version by v2, v3, ..., etc.

Reference Solution

An executable “ref” file is provided for your reference. Follow the usage, it will generate the reference solution.

- `$> ./ref`
- `$> # of node`
- `$> # of edge`
- `$> from node, to node, weight of edge`

The following example is shown in Figure 3.

```
$ ./ref
2
2
[case info] #node:2, #edge:2
1 2 2
2 1 3
[ref solution] MMC: 2.50
[ref solution] Cycle: 1 2 1
```

Figure 3: Example of executing ref