Homework Assignment #10: Programming Exercise #2

Note

This assignment constitutes 4% of your grade and is due 2:10PM Tuesday, June 16, 2015. Please write/type your answers/code on A4 (or similar size) paper. Drop your homework by the due time in Yih-Kuen Tsay's mail box on the first floor of Management College Building II. Late submission will be penalized by 20% for each working day overdue. You may discuss the problem with others, but copying answers/code is strictly forbidden.

Your work will be graded according to its correctness and presentation. Specifically, you should provide evidences showing that your program is correct. You should also organize and document your program in such a way that other programmers, for example your classmates, can understand it. **Some of you may be requested to demonstrate your program.**

Problem

Implement the algorithm (discussed in class) for computing the strongly connected components of a directed graph. (Note: you may want to take this opportunity to try the two different ways of updating the *High* value of a vertex when it sees another vertex through a cross or back edge.)

Please follow the input format as described below. The first line of an input contains one integer n ($\leq$ 1000), indicating the number of vertices in the graph; the vertices are then identified by numbers 1 through n. Each of the following lines represents the adjacency list of a particular vertex u, with the first integer giving the identifier of u followed by the identifiers of those vertices (in no particular order) that are connected by an edge from u. Below is a sample input file:

```
9
1 2 8
2 3
3 1 4 7
4 5
5 6
6 4
7 5
8 3 9
9 7
```

For the output, each line contains the identifiers of a strongly connected component.

```
4 5 6
7
```

```
9
1 2 3 8
```