

# Algorithms

## Homework 3: due 31 May 2018

1. You are to write a program that finds strongly connected components of a directed graph. Your program should be able to find strongly connected components when the input graph is given as

- an adjacency matrix,
- an adjacency list, and
- an adjacency array (using one array for vertices and another array for edges).

Your program should read the input graph from an input file. The input file contains  $n$  (the vertices are  $1, 2, \dots, n$ ) in the first line, and the  $(i + 1)$ st line contains edges going out of vertex  $i$  (the first number in the line is the number of edges). For example, 3 2 10 7 in the second line means that there are edges  $(1, 2)$ ,  $(1, 10)$ , and  $(1, 7)$ , and 0 in the third line means that there are no edges going out of vertex 2.

Your program should output each strongly connected component in a line such that vertices in a line are sorted, and the lines appear in the output in the lexicographic order.

2. Your program should proceed as follows.

- (1) Read the input graph from an input file
  - (2) Run your program when the graph is given as an adjacency matrix, and measure the time. Print the output and the time.
  - (3) Run your program when the graph is given as an adjacency list, and measure the time. Print the output and the time.
  - (4) Run your program when the graph is given as an adjacency array, and measure the time. Print the output and the time.
- Measure the time on various inputs (different values of  $n$ , sparse and dense, etc.) and discuss the results.
  - Hand in your report, program, and an example running by email to TA ([hwkim@theory.snu.ac.kr](mailto:hwkim@theory.snu.ac.kr)).
  - Write down the environment you run your program and how to run your program in your report.
  - Write comments appropriately in your program.