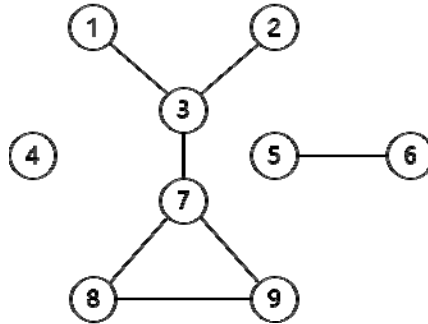


Connected Component Algorithm

Undirected 그래프 G 가 주어졌을 때, 이 그래프의 connected component를 계산하는 알고리즘을 구현하시오.

예를 들어, 9개의 노드와 7개의 에지로 구성된 아래 그래프 G 는 3개의 connected component를 가지고 있다.



<< 주의 >>

다음과 같은 경우 혹은 그 외의 편법을 통한 프로그램을 제출하는 경우에는 이 문제의 점수뿐만 아니라 다른 모든 과제의 점수를 0 점으로 처리함.

1. 위 해결 프로그램을 Recursive DFS에 의한 구현 방법 이외의 다른 방법으로 구현하여 제출하는 경우.
2. 채점 프로그램 시스템에 Recursive DFS에 의한 구현 방법 이외의 다른 방법으로 구현한 프로그램을 한 번이라도 제출하는 경우.

입력

입력은 표준입력(standard input)을 사용한다. 입력은 t 개의 테스트 케이스로 주어진다. 입력 파일의 첫 번째 줄에 테스트 케이스의 개수를 나타내는 정수 t 가 주어진다. 두 번째 줄부터 테스트 케이스가 입력된다. 각 테스트 케이스에 해당되는 첫 번째 줄에는 어떤 그래프의 노드의 개수를 나타내는 정수 n ($3 \leq n \leq 1,000$)이 주어진다. 각 노드는 1부터 n 까지 차례로 번호가 부여된다. 그 다음 n 개의 줄에는 한 줄에 하나의 노드 k ($1 \leq k \leq n$)에 인접한 노드의 번호가 주어진다. 각 줄에는 먼저 노드 번호 k ($1 \leq k \leq n$)가 주어지고, 그 다음에는 노드 k 에 인접한 노드들의 개수 m ($0 \leq m \leq n - 1$)이 주어진다. 그 다음에는 m 개의 정수 $v_1 v_2 \dots v_m$ 이 주어지는데, v_1, v_2, \dots, v_m 은 노드 k 에 인접한 노드들의 번호를 나타낸다. m 이 0인 경우는 인접한 노드가 없는 노드를 의미한다. 이 노드 번호들은 정렬되지 않고 임의의 순서로 입력될 수 있다. 각 정수들 사이에는 한 개의 공백이 있으며, 잘못된 데이터가 입력되는 경우는 없다.

출력

출력은 표준출력(standard output)을 사용한다. 입력되는 테스트 케이스의 순서대로 다음 줄에 이어서 각 테스트 케이스의 결과를 한 줄에 출력한다. 각 테스트 케이스에 해당하는 출력의 각 줄에 입력되는 그래프의 connected component의 개수를 출력한다. 그 다음에는 각 connected component의 노드의 개수를 오름차순으로 출력한다. 각 정수들 사이에는 한 개의 공백을 둔다.

입력과 출력의 예

입력	출력
3	3 1 2 6
9	3 1 1 1
1 1 3	1 3
2 1 3	
3 3 1 2 7	
4 0	
5 1 6	
6 1 5	
7 3 3 8 9	
8 2 7 9	
9 2 7 8	
3	
1 0	
2 0	
3 0	
3	
1 2 2 3	
2 2 1 3	
3 2 1 2	