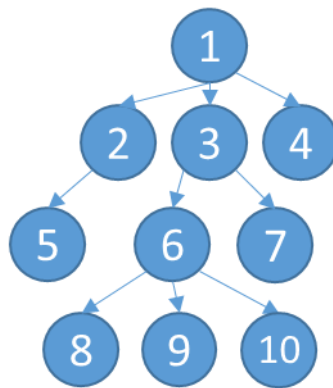


WEEK7

- 주의 사항: 부정행위 금지(채점서버 외 인터넷 사용금지), STL 사용금지 (string, vector는 사용 가능)
- 표준 입출력 사용을 권장 (C는 scanf / printf, C++은 cin / cout)

문제 2 (Hard version)

연우는 입력된 트리의 내용을 순회하여 출력하는 프로그램을 작성하려고 한다.



예를 들어 위와 같이 트리가 입력되었을 때, 트리를 순회한 결과는 다음과 같을 것이다. 단, 루트(root)의 깊이(depth)는 0이며, 트리의 높이(height)는 트리에서 임의의 노드 중 가장 긴 깊이를 의미한다.

- 전위 순회한 결과: 1 2 5 3 6 8 9 10 7 4
- 후위 순회한 결과: 5 2 8 9 10 6 7 3 4 1
- 트리의 높이: 3

연우를 도와 트리의 순회 결과를 출력하는 코드를 작성해보자.

Hard: 트리의 전위 순회(preorder traversal) 결과, 후위 순회(postorder traversal) 결과, 트리의 높이를 출력한다.

입력

첫째 줄에는 트리의 개수 $N(1 \leq N \leq 100)$ 이 주어진다. 그 후 N 번 동안 트리에 대한 정보가 다음과 같이 반복되어 주어진다.

- 첫째 줄에 하나의 트리에 삽입될 데이터의 수 $M(1 \leq M \leq 1,000)$ 이 주어진다.
- 둘째 줄부터 M 개의 줄에 걸쳐 부모 노드 - 자식 노드 쌍이 주어진다.
(부모 노드는 항상 트리에 존재)

각 트리에는 데이터를 삽입하기 전부터 이미 루트 노드 1이 존재한다고 가정한다. 또한 각 트리에 대

해 입력되는 노드 값($1 \leq X \leq 100,000$)에는 중복이 없다.

출력

각 트리에 대해, 첫째 줄에 전위 순회한 결과를, 둘째 줄에 후위 순회한 결과를, 셋째 줄에 트리의 높이를 출력한다.

예제 입출력

예제 입력	예제 출력
2	1 2 5 3 6 8 9 10 7 4
9 // 첫 번째 트리 (위 예시 그림)	5 2 8 9 10 6 7 3 4 1
1 2	3
1 3	1 2 3
1 4	2 3 1
2 5	1
3 6	
3 7	
6 8	
6 9	
6 10	
2 // 두 번째 트리	
1 2	
1 3	