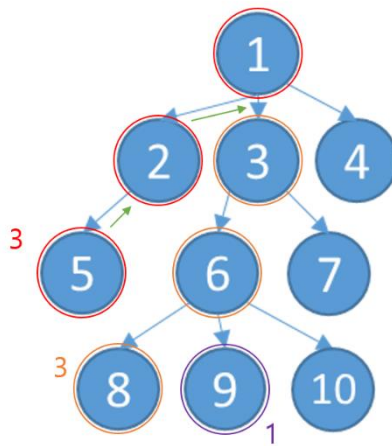


# WEEK7

- 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

전위 순회를 돌면서 자식이 없는 노드가 나올 때까지 순회 함수 호출 수를 리턴하고자 한다. 한번 리턴하면 다음 자식이 없는 노드가 나올 때까지 순회 함수 호출 수를 0부터 다시 카운트하여 리턴한다. 이때 순회 함수는 앞에 문제 1번과 같이 노드를 입력 받아 함수를 종료할 지 확인하고 자식 노드를 넣은 순회 함수를 호출하는 재귀 형식을 사용한다. 밑의 그림을 보자.



처음 순회 함수를 3번 호출되어 3를 리턴하고 다시 1인 노드로 돌아갈 때 순회 함수를 호출한 것이 아니므로 연두색 화살표는 카운트하지 않는다. 다시 8인 노드를 호출하기 위해 3번 순회 함수가 호출된다.

위 트리를 통해 출력되는 값은 다음과 같다: 3 3 1 1 1 1

## 입력

첫째 줄에는 트리의 개수  $N(1 \leq N \leq 100)$ 이 주어진다. 그 후  $N$ 번동안 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 대해 입력되는 노드 값( $1 \leq X \leq 100,000$ )에는 중복이 없다.

출력

각 트리에 대해 전위 순회하면서 나오는 순회 함수 호출 수를 순서대로 호출한다.

예제 입출력

예제 입력	예제 출력
2	3 3 1 1 1 1
9	2 1
1 2	
1 3	
1 4	
2 5	
3 6	
3 7	
6 8	
6 9	
6 10	
2	
1 2	
1 3	