

9월

항저우의 중앙 광장에는 오래된 유명한 나무가 있는데, 이 나무는 0부터 $N - 1$ 까지 정수로 번호가 매겨진 N 개의 노드가 있는 트리로 생각할 수 있다. 노드 0이 루트이다.

자식이 없는 노드는 **리프 노드**라고 한다. 나무에서 잎이 떨어질 때마다, 리프 노드 하나를 골라서 삭제한다. 하루에 여러번 잎이 떨어질 수 있다.

0부터 $M - 1$ 까지 정수로 번호가 매겨진 M 명의 자원봉사자가 이 나무를 지키고 있다. 각각은 독립적으로 다음과 같은 방법을 써서 잎이 떨어지는 상황을 기록한다.

매일매일, 새롭게 떨어진 잎의 인덱스들을 (즉, 이 날 삭제된 노드들의 인덱스들을) 수집한다. 그리고 이들을 이전에 떨어진 잎들 뒤에 기록하는데, 어떤 순서도 가능하다.

예를 들어, 첫날 잎 3과 4가 떨어졌다면 3, 4 또는 4, 3으로 기록한다. 둘째날 잎 1과 2가 떨어졌다면, 1, 2 또는 2, 1로 기록한다. 최종적인 기록은 (3, 4, 1, 2), (4, 3, 1, 2), (3, 4, 2, 1), (4, 3, 2, 1) 중 하나이다.

이 과정은 K 일 동안 반복되는데, **하루에 적어도 잎 하나는 떨어진다**. 최종적으로는 루트 노드만 남게 된다.

여러분은 추운 겨울 항저우를 방문하게 되었다. 이 나무의 앙상한 가지를 보니, 낙엽이 떨어지는 모습을 상상하지 않을 수 없었다. 낙엽이 며칠동안 떨어졌는지 궁금한데, 알수 있는 것은 자원봉사자 M 명이 남긴 기록밖에 없다. 이 기록으로부터, K 의 가능한 최대값을 구하시오.

Implementation Details

다음 함수를 구현해야 한다.

```
int solve(int N, int M, std::vector<int> F,  
          std::vector<std::vector<int>> S);
```

- N : 나무의 노드 수.
- M : 자원봉사자의 수.
- F : 길이 N 인 정수 배열. $1 \leq i \leq N - 1$ 일 때, $F[i]$ 는 노드 i 의 부모 노드의 인덱스이다. $F[0]$ 는 항상 -1 이다.
- S : M 개의 배열의 배열. S 의 각 원소는 길이 $N - 1$ 인 정수 배열이다. $S[i][j]$ 는 자원봉사자 i (자원봉사자 번호는 0부터 시작한다)의 기록에서 j 번째 위치에 기록된 값이다.
- 이 함수는 위 규칙을 따랐을 때 가능한 K 의 최대값(즉, 잎이 떨어질 수 있는 기간의 최대 길이)을 리턴해야 한다.

- 각각의 테스트 케이스에 대해서, 그레이더는 이 함수를 한 번 이상 호출할 수 있다. 각각의 호출은 별개의 새로운 시나리오로 처리된다.
- 주의: 이 함수가 한번 이상 호출될 것이기 때문에, 여러분은 직전 함수 호출이 이번 함수 호출에 미칠 수 있는 영향에 대해서 주의를 기울여야 한다. 특히, 전역변수에 저장된 값들에 유의하라.

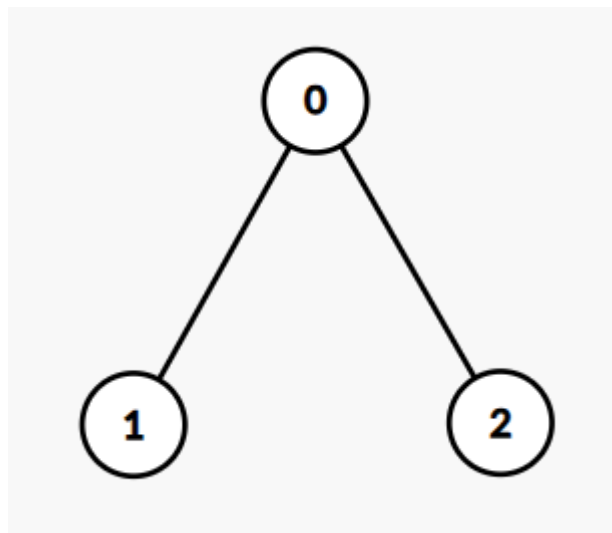
Examples

Example 1

다음 호출을 생각해보자.

```
solve(3, 1, {-1, 0, 0}, {{1, 2}});
```

해당하는 트리는 다음과 같은 모양이다.



앞 1과 2는 같은 날에 떨어질 수도 있고, 첫날 앞 1이 떨어지고 다음날 앞 2가 떨어질 수도 있다. 앞이 떨어질 수 있는 기간은 2일을 초과할 수 없다.

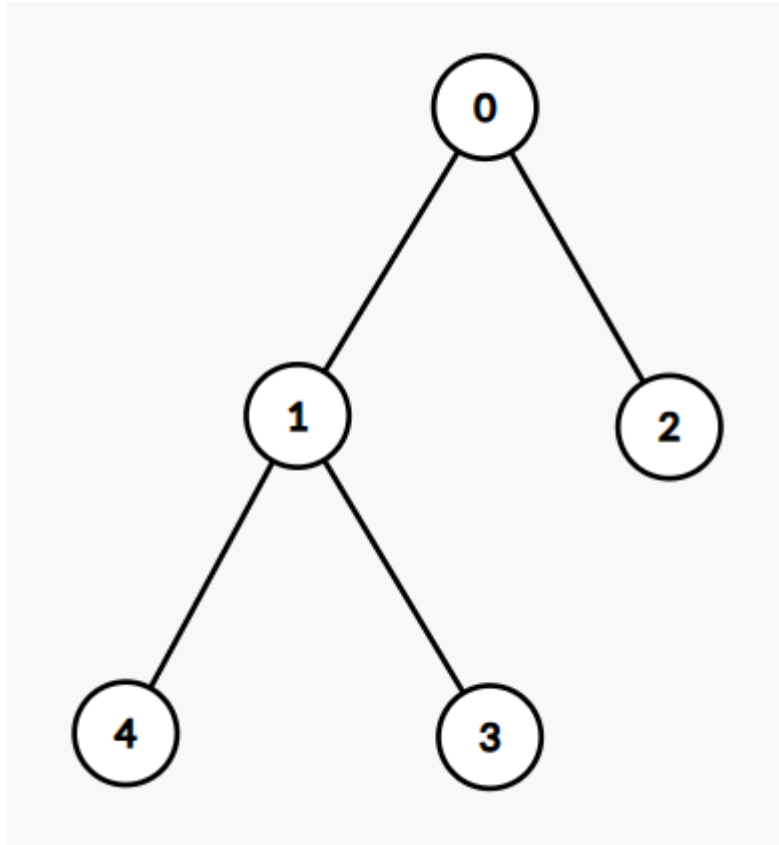
따라서, 이 함수의 리턴값은 2여야 한다.

Example 2

다음 호출을 생각해보자.

```
solve(5, 2, {-1, 0, 0, 1, 1}, {{1, 2, 3, 4}, {4, 1, 2, 3}});
```

해당하는 트리는 다음과 같은 모양이다.



앞이 떨어진 기간이 2일 이상이라고 가정하자. 자원봉사자들의 기록에 따르면 앞 4가 떨어진 날이 서로 달라야 한다. (첫날과 마지막날) 이것은 모순이기 때문에, 함수의 리턴값은 1이어야 한다.

Constraints

- $2 \leq N \leq 10^5$.
- $1 \leq M \leq 5$.
- $\sum NM \leq 8 \times 10^5$.
- $F[0] = -1$. $1 \leq i \leq N - 1$ 에 대해서, $0 \leq F[i] \leq i - 1$.
- $1 \leq i \leq M - 1$ 에 대해서, 배열 $S[i]$ 은 $1, 2, \dots, N - 1$ 의 순열을 저장한다.
- F 에 저장된 값이 루트가 있는 트리를 나타내며, 노드 0이 루트 노드라는 것이 보장된다.

Subtasks

1. (11 points): $M = 1, N \leq 10, \sum N \leq 30$.
2. (14 points): $N \leq 10, \sum N \leq 30$.
3. (5 points): $M = 1, N \leq 1\,000, \sum N \leq 2\,000, F[i] = i - 1$.
4. (9 points): $M = 1, N \leq 1\,000, \sum N \leq 2\,000$.
5. (5 points): $N \leq 1\,000, \sum N \leq 2\,000, F[i] = i - 1$.
6. (11 points): $N \leq 1\,000, \sum N \leq 2\,000$.
7. (9 points): $M = 1, F[i] = i - 1$.
8. (11 points): $M = 1$.
9. (9 points): $F[i] = i - 1$.
10. (16 points): 추가적인 제약 조건이 없다.

Sample Grader

샘플 그레이더는 다음 양식으로 입력을 읽는다.

- Line 1: T

T 개의 테스트 케이스 각각에 대해서

- Line 1: $N \ M$
- Line 2: $F[1] \ F[2] \ \dots \ F[N - 1]$
- Line $3 + i \ (0 \leq i \leq M - 1)$: $S[i][0] \ S[i][1] \ S[i][2] \ \dots \ S[i][N - 2]$

샘플 그레이더는 여러분의 답을 다음 양식으로 출력한다.

각각의 테스트 케이스에 대해서

- Line 1: solve의 리턴값