

September

The central square of Hangzhou is home to a famous ancient tree, which can be regarded as a rooted tree with N nodes, indexed from 0 to $N - 1$, with node 0 being the root.

A node with no children is called a **leaf node**. Every time the ancient tree sheds its leaves, it selects one leaf node at that time to delete, and it may shed leaves multiple times in the same day.

There are M volunteers (indexed from 0 to $M - 1$) responsible for guarding the ancient tree. Each of them independently records the leaf-shedding situation of this year, using the following method:

Every day, collect the indices of all newly fallen leaves (i.e. the indices of the nodes which are deleted on that day), and write them down in any order after the previous fallen leaves.

For example: On the first day, leaves 3 and 4 fall, so they write down 3, 4 or 4, 3. On the second day, leaves 1 and 2 fall, so they continue to write down 1, 2 or 2, 1. The final record may be any of (3, 4, 1, 2), (4, 3, 1, 2), (3, 4, 2, 1), or (4, 3, 2, 1).

The process lasts for K days, **with newly fallen leaves every day**, until only the root node remains.

While traveling, you happen to visit Hangzhou. It is now a cold winter. Looking up at the bare branches of the ancient tree, you can't help but imagine the beautiful sight of falling leaves.

You are very curious to know how many days you could have seen falling leaves this year, but you can only find the records of the M volunteers. Try to infer the maximum possible value of K from the records.

Implementation Details

You need to implement the following function:

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

- N : The number of nodes of the ancient tree.
- M : The number of volunteers.

- F : An integer array with a length of N . For $1 \leq i \leq N - 1$, $F[i]$ represents the index of the parent node of node i . $F[0]$ is always -1 .
- S : An array containing M arrays. Each element of S is an integer array with a length of $N - 1$. $S[i][j]$ represents The j -th index recorded by volunteer i (starting from 0).
- The function must return an integer that represents the maximum possible value of K (i.e. the feasible maximum number of leaf-falling days) according to the above rules.
- For each test case, the grader may call this function for more than once. Each call should be processed as a separately new scenario.

Note: **Since the function will be called more than once, contestants need to pay attention to the impact of the remaining data of the previous call on the current call, especially the state stored in global variables.**

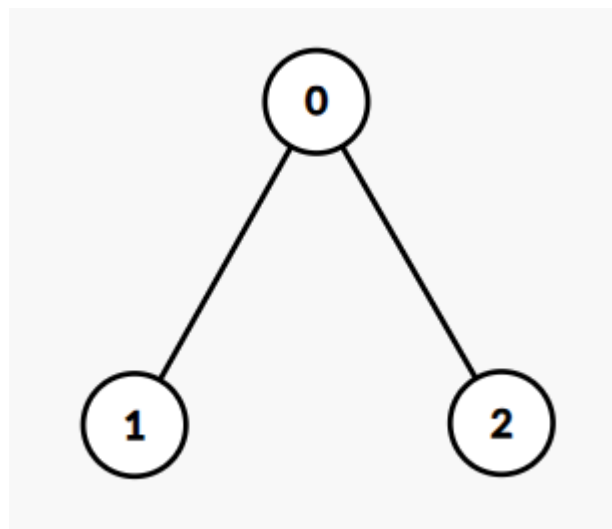
Examples

Example 1

Consider the following call:

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

The corresponding tree is shown below:



Leaves 1 and 2 may fall on the same day, or 1 may fall first on the first day, followed by 2 on the second day. The leaf-falling days lasts no more than 2 days.

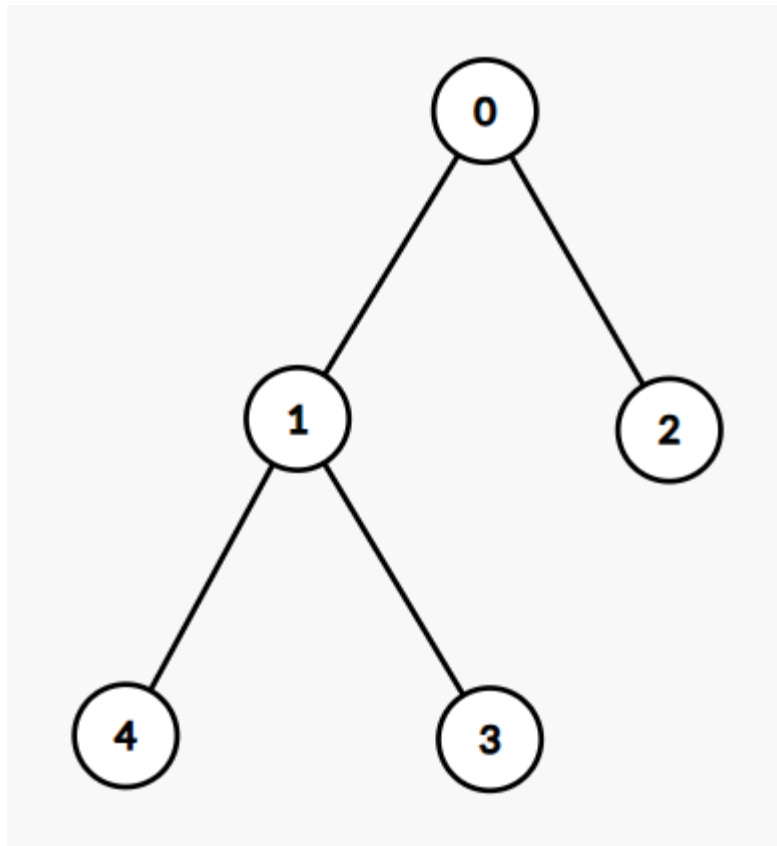
Therefore, the procedure should return 2.

Example 2

Consider the following call:

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

The corresponding tree is shown below:



Assuming there are at least 2 leaf-falling days, according to the volunteers' records, leaf 4 will fall on different days (the first and last), which is contradictory.

Therefore, the procedure should return 1.

Constraints

- $2 \leq N \leq 10^5$.
- $1 \leq M \leq 5$.
- $\sum NM \leq 8 \times 10^5$.
- $F[0] = -1$. For $1 \leq i \leq N - 1$, $0 \leq F[i] \leq i - 1$.
- For $1 \leq i \leq M - 1$, array $S[i]$ is a permutation of $1, 2, \dots, N - 1$.
- It is guaranteed that F describes a rooted tree with node 0 being the root.

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): No additional constraints.

Sample Grader

The sample grader reads the input in the following format:

- Line 1: T

For each of the following T test cases:

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

The sample grader prints your answers in the following format:

For each test cases:

- Line 1: the return value of solve