

# Tasks from Sasha

Input file:            **standard input**  
Output file:          **standard output**  
Time limit:          5 seconds  
Memory limit:        512 megabytes

Sasha recently moved into a multi-story building. The building has a total of  $n$  floors, numbered from 1 to  $n$ . There is exactly one resident living on each floor. Between the floors, there are  $n - 1$  staircases, and these staircases are not necessarily built between adjacent floors. It is known that for every floor except the first, there is exactly one staircase leading to a lower floor. For the  $i$ -th floor ( $2 \leq i \leq n$ ), this staircase leads to the floor numbered  $p_i$ .

Sasha plans to solve  $k$  tasks numbered from 1 to  $k$ . For the task with number  $i$ , Sasha has calculated that it is most optimal to solve it on the floor numbered  $x_i$ . Since the tasks are different from each other, all values of  $x_i$  are distinct.

It is boring to solve tasks alone, so for each task, Sasha wants to invite at least one resident to solve it together. However, the residents of the building really dislike climbing stairs and are only willing to descend to the floors where the tasks will be solved. Therefore, Sasha can invite a resident from floor  $j$  to solve task  $i$  only if there is a way to reach floor  $x_i$  from floor  $j$ , using several, possibly zero, staircases that each time lead to a floor with a lower number. Thus, a resident from floor  $j$  can solve task  $i$  only if  $j = x_i$  or  $p_j = x_i$ , or  $p_{p_j} = x_i$ , and so on.

Residents really dislike descending the stairs unnecessarily. Therefore, if Sasha invites a certain set of people to solve a task, they will only be willing to gather to solve the task on the **highest floor that they all can descend to**. For example, if there is a staircase from floor 3 to floor 2, then Sasha cannot invite residents from floors 2 and 3 to solve the task on floor 1, as all residents can gather on a higher floor.

Sasha does not like to appear intrusive, so he will invite a resident from each floor to solve no more than one task. At the same time, he may choose not to invite residents from some floors to solve tasks.

Sasha has one more favorite task that he will not share with anyone except you. But for him to tell you about it, you need to help him count the number of different ways to invite residents to solve tasks with him, ensuring that all restrictions are met. Two ways are considered different if at least one task is solved by a different set of residents.

## Input

The first line contains two integers  $n$  and  $k$  ( $3 \leq n \leq 10^6$ ,  $1 \leq k \leq \min(n, 2000)$ ) — the number of floors in the building and the number of tasks.

The second line contains  $k$  integers  $x_1, x_2, \dots, x_k$  ( $1 \leq x_i \leq n$ ) — the floors on which Sasha will solve the tasks. It is guaranteed that all  $x_i$  are distinct.

The third line contains  $n - 1$  integers  $p_2, p_3, \dots, p_n$  ( $1 \leq p_i < i$ ), where  $p_i$  describes the number of the floor to which the staircase leads down from floor  $i$ .

## Output

Output one number — the number of different ways to invite residents of the building to solve tasks together with Sasha, modulo 998 244 353.

## Examples

standard input	standard output
3 1 1 1 1	5
6 2 2 5 1 2 3 4 5	12
7 3 2 7 1 1 1 2 2 3 3	62

## Note

In the first example, Sasha has five ways to invite residents to solve tasks:

- only with the resident on floor 1;
- with residents on floors 1 and 2;
- with residents on floors 1 and 3;
- with residents on floors 1, 2, and 3;
- with residents on floors 2 and 3;

Sasha cannot invite the resident from floor 2 alone to solve the task, as then the highest floor where all residents willing to solve the task could gather would be floor 2, while Sasha wants to solve the task on floor 1.

In the second example, two different suitable ways to invite residents to solve tasks could be as follows:

- Invite residents from floors 2 and 6 to solve the first task, and the resident from floor 5 to solve the second task.
- Invite the resident from floor 2 to solve the first task, and residents from floors 5 and 6 to solve the second task.

## Scoring

The tests for this problem consist of nine groups. Points for each group are awarded only if all tests in the group and all tests in some of the previous groups are passed. Note that passing the tests from the statement is not required for some groups. **Offline verification** means that the results of testing your solution on this group will only be available after the competition ends. The final score for each group is equal to the maximum score obtained for that group of tests across all submitted solutions.

Group	Points	Additional constraints		Required groups	Comment
		$n$	$k$		
0	0	–	–	–	Tests from the statement.
1	12	$n \leq 10$	$k \leq 10$	0	
2	13	$n \leq 500$	$k \leq 500$	0, 1	
3	9	–	$k = 1$	–	
4	10	–	–	–	$p_i = i - 1$
5	13	–	–	4	Each floor is connected to no more than two floors with a higher number
6	14	$n \leq 200\,000$	$k \leq 500$	0 – 2	
7	11	–	$k \leq 500$	0 – 3, 6	
8	10	–	$k \leq 1000$	0 – 3, 6, 7	
9	8	–	–	0 – 8	<b>Offline verification</b>