

Covering a Tree

Input file: **standard input**
Output file: **standard output**
Time limit: 1 second
Memory limit: 1024 megabytes

Given a tree with n nodes, you can use several line segments of arbitrary lengths to cover all the edges of the tree, with the following requirements:

1. Each edge must be covered exactly once;
2. Each line segment must start from a leaf node and end at one of its ancestor nodes.

You can choose any number of line segments such that these segments can cover the entire tree according to the above requirements, but you need to minimize the maximum length of the segments. Your task is to find this minimum value.

Input

The first line contains an integer T ($1 \leq T \leq 10^5$), representing the number of test cases.

For each test case, the first line contains an integer n ($2 \leq n \leq 2 \cdot 10^5$), representing the total number of nodes in the tree.

The second line contains $n - 1$ integers p_2, p_3, \dots, p_n ($1 \leq p_i < i$), where p_i represents the index of the parent node of node i .

It is guaranteed that the sum of n across all test cases does not exceed $2 \cdot 10^5$.

Output

For each test case, output a single integer on a new line, representing the minimum possible maximum length of the line segments.

Example

standard input	standard output
2	3
8	7
1 2 3 2 5 1 7	
8	
1 2 3 4 5 6 7	

Note

For the first sample test case, the illustration is as follows. The red, green, and blue segments represent three line segments with lengths of 2, 3, and 2, respectively.

