Problem A. TreeScript

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

TreeScript is a programming language developed for maintaining tree structures. In this problem, we will learn how to create rooted trees in TreeScript.

In TreeScript, all tree nodes are stored in memory. Each tree node has a number and the address of its parent node, and both are immutable, so they have to be determined when creating the node. In particular, the address of the root node's parent node is empty.

In order to access these nodes, the address of a node can be stored in a register. If there are m registers, the registers can be written as $r[0], r[1], ..., r[m-1]$.

Now let's learn the node creation statement:

$$r[i] = \text{create}(r[j], k);$$

where k is the node number, i and j are the indices of the registers, where $0 \leq i, j < m$ and $i = j$ is possible. The effect of this statement is that a node numbered k is created, whose parent address is stored in $r[j]$, and then the new node's address is stored in $r[i]$. Once each node has been created correctly, you do not need to store the address of any more nodes; they will automatically execute the pre-defined instructions. For reasons of space, we will learn about them later.

To check your learning, you need to create a rooted tree of size n. At first, the system will automatically create the root node for you and store it in $r[0]$. So you only need to execute $n-1$ additional creation instructions to create the tree.

As you know, registers are very expensive, so you need to find the minimum value of the registers you need.

Input

There are multiple test cases.

The first line of the input contains one integer T ($1 \leq T \leq 10^5$) - the number of test cases.

For each test case,

The first line contains one integer n ($2 \leq n \leq 2 \times 10^5$) - the size of the tree.

The second line contains n integers $p_1, p_2, \cdots, p_n$, where node p_i is the parent node of node i and $1 \leq p_i < i$. Specially, $p_1 = 0$ and it means 1 is the root of the tree.

The sum of n over all test cases does not exceed 2×10^5.

Output

For each test case, output the answer in one line.

Example

standard input	standard output
2 3 0 1 2 7 0 1 2 2 1 4 1	1 2