Problem Statement

A tree is an object that consists of some locations and some connections. The locations are called vertices and the connections between vertices are called edges. Each edge connects some pair of vertices. The number of vertices is always one greater than the number of edges. The entire tree is connected: the edges connect the vertices in such a way that one can walk from any vertex to any other vertex by following a sequence of edges.

You are given the description of a tree with n+1 vertices and n edges: a tree with n elements. The vertices of our tree are numbered 0 through n. For each i between 0 and n-1, inclusive, there is an edge between vertices i+1 and tree[i]. (The constraints guarantee that these edges always form a valid tree.)

You are now going to destroy the tree by removing one of its vertices (and all edges that go from that vertex to some other vertex). When you do so, the tree may sometimes fall apart: there will be more than one connected component.

Compute and return the largest number of components you can get by removing a single vertex of your tree.

Definition

Class:

TreeAndVertex

Method:

get

Parameters:

int[] tree

Returns:

int

Method signature:

(be sure your method is public)

Limits

Time limit (s):

2.000

Memory limit (MB):

256

Notes

- The number n is not given explicitly. Instead, you can determine it as the number of elements in tree.

Constraints

- tree will contain between 1 and 99 elements, inclusive.

- For each valid i tree[i] will be between 0 and i, inclusive.

Examples

0)

{0,0,0}

Returns: 3

Here are four vertices and three edges 0-1, 0-2, 0-3. One can remove vertex 0 and get three connected components.

1)

{0, 1, 2, 3}

Returns: 2

Here are five vertices and four edges 0-1, 1-2, 2-3, 3-4. One can remove any vertex, except 0 and 4, to get two connected components.

2)

{0, 0, 2, 2}

Returns: 3

3)

{0, 0, 0, 1, 1, 1}

Returns: 4

4)

{0, 1, 2, 0, 1, 5, 6, 1, 7, 4, 2, 5, 5, 8, 6, 2, 14, 12, 18, 10, 0, 6, 18, 2, 20, 11, 0, 11, 7, 12, 17, 3, 18, 31, 14, 34, 30, 11, 9}

Returns: 5