https://course.acciojob.com/idle?question=954ccbee-ad1c-4426-ae83-6a181746b3bc

- MEDIUM
- Max Score: 40 Points

Array Nesting

You are given an integer array `nums` of length `n` where `nums` is a permutation of the numbers in the range `[0, n - 1]`.

You should build a set `s[k] = {nums[k], nums[nums[k]], nums[nums[nums[k]]], ... }` subjected to the following rule:

- The first element in `s[k]` starts with the selection of the element `nums[k]` of `index = k`.
- The next element in `s[k]` should be `nums[nums[k]]`, and then `nums[nums[nums[k]]]`, and so on.
- We stop adding right before a duplicate element occurs in `s[k]`.

Return the longest length of a set `s[k]`.

Input Format

The first line of input contains integer `n` representing the size of array `nums` respectively.

The second line of input contains `n` space-separated integers representing the elements of array `nums`.

Output Format

The only line of output contains a single integer representing the longest length of a set `s[k]`.

Example 1

Input

```
7
5 4 0 3 1 6 2
```

Output

```
4
```

Explanation

nums[0] = 5, nums[1] = 4, nums[2] = 0, nums[3] = 3, nums[4] = 1, nums[5] = 6, nums[6] = 2. One of the longest sets s[k]:

s[0] = {nums[0], nums[5], nums[6], nums[2]} = {5, 6, 2, 0}

Example 2

Input

```
3
0 1 2
```

Output

```
1
```

Explanation

Same explanation as example 1.

Constraints:

`1 <= nums.length <= 10^5`

`0 <= nums[i] < nums.length`

All the values of `nums` are unique.

Topic Tags

- **DFS**
- **Arrays**

My code

```
// n java
import java.util.*;

class Solution {
    static int length(int[] arr,int i,int visit[])
    {
        int ans=1;
        visit[i]=1;
        int f=arr[i];
        i=arr[i];

        while(arr[i]!=f)
            {
                visit[i]=1;
```

```
                                ans++;
                                i=arr[i];
                        }
                return ans;
        }
    public int arrayNesting(int[] arr) {
        // Write your code here
                int n=arr.length;
                int ans=0;
                int visit[]=new int[n];
             for(int i=0;i<n;i++)
                         {
                                 if(visit[i]!=1)
                                 {
                                         int t=length(arr,i,visit) ;
                                 if(t>ans)
                                  ans=t;
                                 }
                         }

                return ans;
    }
}

public class Main{
    public static void main(String[] args) throws Exception {
        Scanner sc = new Scanner(System.in);
        int n = sc.nextInt();
        int[] arr = new int[n];
        for (int i = 0; i < n; i++) {
            arr[i] = sc.nextInt();
        }
        sc.close();
        Solution Obj = new Solution();
        System.out.println(Obj.arrayNesting(arr));
    }
}
```