Sequence Equation

by zemen

Problem Submissions Leaderboard Discussions Editorial

You are given a sequence of n integers, $p(1), p(2), \ldots, p(n)$. Each element in the sequence is distinct and satisfies $1 \leq p(x) \leq n$. For *each* x where $1 \leq x \leq n$, find any integer y such that $p(p(y)) \equiv x$ and print the value of y on a new line.

Input Format

The first line contains an integer, n, denoting the number of elements in the sequence.
The second line contains n space-separated integers denoting the respective values of $p(1), p(2), \ldots, p(n)$.

Constraints

- $1 \leq n \leq 50$
- $1 \leq p(x) \leq 50$, where $1 \leq x \leq n$.
- Each element in the sequence is distinct.

Output Format

For each x from 1 to n, print an integer denoting any valid y satisfying the equation $p(p(y)) \equiv x$ on a new line.

Sample Input 0

```
3
2 3 1
```

Sample Output 0

```
2
3
1
```

Explanation 0

Given the values of $p(1) = 2$, $p(2) = 3$, and $p(3) = 1$, we calculate and print the following values for each x from 1 to n:

1. $x = 1 \equiv p(3) = p(p(2)) = p(p(y))$, so we print the value of $y = 2$ on a new line.
2. $x = 2 \equiv p(1) = p(p(3)) = p(p(y))$, so we print the value of $y = 3$ on a new line.
3. $x = 3 \equiv p(2) = p(p(1)) = p(p(y))$, so we print the value of $y = 1$ on a new line.

Submissions: 3269
Max Score: 20
Difficulty: Easy

Rate This Challenge:

More

Current Buffer (saved locally, editable) C#

```
using System;
using System.Collections.Generic;
using System.IO;
using System.Linq;

namespace Solution {
class Solution {
    static void Main(string[] args) {
        /* Enter your code here. Read input from STDIN. Print output to STDOUT */

         int n = int.Parse(Console.ReadLine());
            int[] a = Array.ConvertAll(Console.ReadLine().Split(' '), e => int.Parse(e));
            //3
             //int[] a = { 2, 3, 1 };

             Dictionary<int, int> indices = new Dictionary<int, int>();
             for (int i = 0; i < a.Length; i++)
             {
                 indices[a[i]] = i;
             }

            for (int i = 1; i <= a.Length; i++)
            {

                //int pos = Array.IndexOf(a, i) + 1;
                //int indice_pos = Array.IndexOf(a, pos);
                //Console.WriteLine(indice_pos + 1);
                int pos = indices[i] + 1;
                int indice_pos = indices[pos];
                Console.WriteLine(indice_pos + 1);
            }

    }
  }
}

```

Line: 31 Col: 14

Upload Code as File Test against custom input Run Code Submit Code

Congrats, you solved this challenge!

- ✔ Test Case #0
- ✔ Test Case #1
- ✔ Test Case #2
- ✔ Test Case #3
- ✔ Test Case #4
- ✔ Test Case #5
- ✔ Test Case #6
- ✔ Test Case #7
- ✔ Test Case #8
- ✔ Test Case #9
- ✔ Test Case #10

Next Challenge