Sequence Equation ★

Given a sequence of n integers, $p(1), p(2), \ldots, p(n)$ where each element is distinct and satisfies $1 \leq p(x) \leq n$. For each x where $1 \leq x \leq n$, that is x increments from 1 to n, find any integer y such that $p(p(y)) \equiv x$ and keep a history of the values of y in a return array.

Example

$p = [5, 2, 1, 3, 4]$

Each value of x between 1 and 5, the length of the sequence, is analyzed as follows:

1. $x = 1 \equiv p[3], p[4] = 3$, so $p[p[4]] = 1$
2. $x = 2 \equiv p[2], p[2] = 2$, so $p[p[2]] = 2$
3. $x = 3 \equiv p[4], p[5] = 4$, so $p[p[5]] = 3$
4. $x = 4 \equiv p[5], p[1] = 5$, so $p[p[1]] = 4$
5. $x = 5 \equiv p[1], p[3] = 1$, so $p[p[3]] = 5$

The values for y are $[4, 2, 5, 1, 3]$.

Function Description

Complete the permutationEquation function in the editor below.

permutationEquation has the following parameter(s):

- int p[n]: an array of integers

Returns

- int[n]: the values of y for all x in the arithmetic sequence 1 to n

Input Format

The first line contains an integer n, the number of elements in the sequence.

The second line contains n space-separated integers $p[i]$ where $1 \leq i \leq n$.

Constraints

- $1 \leq n \leq 50$
- $1 \leq p[i] \leq 50$, where $1 \leq i \leq n$.
- Each element in the sequence is distinct.

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.

Sample Input 1

```
5
4 3 5 1 2
```

Sample Output 1

```
1
3
5
4
2
```

Change Theme Language Python 3

```
# Complete the permutationEquation function below.
#
# The function is expected to return an INTEGER_ARRAY.
# The function accepts INTEGER_ARRAY p as parameter.
#

def permutationEquation(p):
    # Write your code here
    d = {val: idx+1 for idx, val in enumerate(p)}

    result = []
    for i in range(1, len(p)+1):
        px = d[i]
        y = d[px]
        result.append(y)
    return result

if __name__ == '__main__':
    fptr = open(os.environ['OUTPUT_PATH'], 'w')

    n = int(input().strip())

    p = list(map(int, input().rstrip().split()))

    result = permutationEquation(p)
```

EMACS Line: 26 Col: 1

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

You have earned 20.00 points!

You are now 1254.8 points away from the 6th star for your problem solving badge.

7% 945.2/2200

Congratulations

You solved this challenge. Would you like to challenge your friends?

Next Challenge

- Test case 0
- Test case 1
- Test case 2
- Test case 3
- Test case 4
- Test case 5
- Test case 6

Compiler Message

```
Success
```

Input (stdin) Download

```
3
2 3 1
```

Expected Output Download

```
2
3
1
```