HackerRank Sample Test 00:11:28 to test end

Missing Term in an AP

Task: Find the missing term in an Arithmetic Progression.

An Arithmetic Progression is defined as one in which there is a constant difference between the consecutive terms of a given series of numbers. You are provided with consecutive elements of an Arithmetic Progression. There is however one hitch: Exactly one term from the original series is missing from the set of numbers which have been given to you. The rest of the given series is the same as the original AP. Find the missing term.

Input Format

The first line contains an Integer N, which is the number of terms which will be provided as input.
This is followed by N consecutive Integers, with a space between each pair of integers. All of these are on one line, and they are in AP (other than the point where an integer is missing).

Output Format

One Number which is the missing integer from the series.

Sample Input

5
1 3 5 9 11

Sample Output

7

Explanation

You are provided with 5 integers. As you can can observe, they have been picked from a series, in which the starting term is 1 and the common difference is 2. The only abberration, i.e. the missing term (7), occurs between 5 and 9. This is the missing element which you need to find.

Constraints

3 <= N <= 2500

All integers in the series will lie in the range [-10^6,+10^6].

YOUR ANSWER

Draft saved 08:12 pm

Python 2

Click here to know how to read from STDIN and write to STDOUT

```
1  # Enter your code here. Read input from STDIN. Print
   output to STDOUT
2  # Time complexity: O(n)
3  n = int(raw_input())
4  nums = map(int, raw_input().split())
5  diff = (nums[-1] - nums[0]) / n
6  cur = nums[0]
7  for i in range(0,n):
8      if cur!= nums[i]:
9          print cur
10         break
11     cur += diff
12
```

Line: 5 Col: 32

Test against custom input

Run Code

Submit code & Continue

Download sample testcases *The input/output files have Unix line endings. Do not use Notepad to edit them on windows.*

Status: Compiled successfully. All available test cases passed!

Testcase 1: *Success*

Your Output

```
21
```

Expected Output

```
21
```

Testcase 2: *Success*

Your Output

```
Output hidden
```

Testcase 3: *Success*

Your Output

```
Output hidden
```

Testcase 4: *Success*

Your Output

```
Output hidden
```

Testcase 5: *Success*

Your Output

```
Output hidden
```

Testcase 6: *Success*

Your Output

```
Output hidden
```

Testcase 7: *Success*

Your Output

```
Output hidden
```