Task 1: Labels (`labels`)

Today is the first day of work for Charles the Courier. He has been tasked with delivering N packages where each package has a (not necessarily unique) label number between 1 and N inclusive. At the end of each day, he is required to report a sequence A of N integers $A_1, \ldots, A_N$ where A_i is the label number of the ith delivered package.

A mathematician at heart, Charles decides to use delta encoding to save on memory space and records a sequence D of $N-1$ integers $D_1, \ldots, D_{N-1}$ instead, where $D_i = A_{i+1} - A_i$.

After delivering all the packages, Charles realises that he does not know how to recover A from D. Your task today is to help him, or state that it is not possible to uniquely recover A.

Input

Your program must read from standard input.

The first line contains a single integer N, the total number of packages.

The second line contains $N-1$ space-separated integers, $D_1, \ldots, D_{N-1}$. D_i represents the difference between the label numbers of the $(i+1)^{th}$ and i^{th} delivered package.

Output

Your program must print to standard output.

If it is possible to uniquely recover A from D, your output should contain N space-separated integers, the sequence A.

Otherwise, your output should contain a single integer on a single line, the integer -1.

Implementation Note

As the input lengths for subtasks 4 and 5 may be very large, you are recommended to use C++ with fast input routines to solve this problem.

C++ and Java source files containing fast input/output templates have been provided in the attachment. You are strongly recommended to use these templates.

If you are implementing your solution in Java, please name your file `Labels.java` and place your main function inside `class Labels`.

Subtasks

The maximum execution time on each instance is 1.0s, and the maximum memory usage on each instance is 1GiB. For all testcases, the input will satisfy the following bounds:

- $2 \leq N \leq 3 \times 10^5$

- $1 \leq A_i \leq N$

- $-N < D_i < N$

Your program will be tested on input instances that satisfy the following restrictions:

Subtask	Marks	Additional Constraints
1	7	$N = 2$
2	15	$2 \leq N \leq 6$
3	25	$2 \leq N \leq 10^3$
4	18	$-1 \leq D_i \leq 1$
5	35	-

Sample Testcase 1

This testcase is valid for subtasks 2, 3, and 5 only.

Input	Output
5 1 3 -2 1	1 2 5 3 4

Sample Testcase 1 Explanation

We are able to uniquely recover $A = [1, 2, 5, 3, 4]$.

This is consistent with D since:

$A_2 - A_1 = 2 - 1 = \quad 1 = D_1$

$A_3 - A_2 = 5 - 2 = \quad 3 = D_2$

$A_4 - A_3 = 3 - 5 = -2 = D_3$

$A_5 - A_4 = 4 - 3 = \quad 1 = D_4$

Sample Testcase 2

This testcase is valid for subtasks 2, 3 and 5 only.

Input	Output
5 2 2 -3 1	1 3 5 2 3

Sample Testcase 2 Explanation

We are able to uniquely recover $A = [1, 3, 5, 2, 3]$. Note that label numbers can appear more than once.

Sample Testcase 3

This testcase is valid for all subtasks.

Input	Output
2 0	-1

Sample Testcase 3 Explanation

We are unable to uniquely recover A since we could have either $A = [1, 1]$ or $A = [2, 2]$.