

The problem of finding the next term of a given sequence of numbers is usually proposed in QI tests. We want to construct a method and a codification that allow us to know all the sequence from the first  $N$  terms.

Let  $S = (S_i)_{i \in \mathbb{N}}$  denote a sequence of real numbers whose  $i$ -order term is  $S_i$ . We codify a constant sequence with the following operator:

$$S = [n] \quad \text{meaning that} \quad S_i = n \quad \forall i \in \mathbb{N},$$

where  $n \in \mathbb{Z}$ . We also define the following operators on a given sequence of numbers  $S = (S_i)_{i \in \mathbb{N}}$ :

$$V = [m + S] \quad \text{meaning that} \quad V_i = \begin{cases} m & , i = 1 \\ V_{i-1} + S_{i-1} & , i > 1 \end{cases};$$

$$V = [m * S] \quad \text{meaning that} \quad V_i = \begin{cases} m * S_1 & , i = 1 \\ V_{i-1} * S_i & , i > 1 \end{cases};$$

where  $m \in \mathbb{N}$ . For example we have the following codifications:

$$\begin{aligned} [2 + [1]] &= 2, 3, 4, 5, 6 \dots & [1 + [2 + [1]]] &= 1, 3, 6, 10, 15, 21, 28, 36 \dots \\ [2 * [1 + [2 + [1]]]] &= 2, 6, 36, 360, 5400, 113400 \dots & [2 * [5 + [-2]]] &= 10, 30, 30, -30, 90, -450, 3150 \dots \end{aligned}$$

Given a sequence of  $N$  integer numbers and an integer  $M$ , the problem is to write the codification that generate the sequence and have at most  $M$  operators. We have that  $2 \leq N \leq 51$  and  $1 \leq M \leq 50$ .

## Input

The input file contains several test cases. For each of them, the program input is a single line containing  $M$  followed by the list of first terms of the sequence. The terms of the given sequence are positive (in the interval  $[1, 200000]$ ) or negative integers (in the interval  $[-200000, -1]$ ), and their number  $N$  can differ but it is always greater than  $M$ .

## Output

For each test case, the program output is a single line containing the codification without any space. If there exists no solution with at most  $M$  operators, the output must be '[0]'.

### Examples

Input	Output
2 2 3 4	[2+[1]]
3 1 3 6 10 15	[1+[2+[1]]]
4 2 6 36 360 5400 113400	[2*[1+[2+[1]]]]

## Sample Input

```
3 10 30 30 -30 90 -450 3150
2 2 6 36 360 5400 113400
```

## Sample Output

```
[2*[5+[-2]]]
[0]
```