White Stone Creek

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	1024 megabytes

There are n piles of stones, each pile initially containing exactly one stone.

You can perform some operations on them: choose a position p ($1 \le p \le n$), then:

- If $p = 1$, take one stone from pile 2 and put it into pile 1 (pile 2 must have at least 1 stone).
- If $p = n$, take one stone from pile $n-1$ and put it into pile n (pile $n-1$ must have at least 1 stone).
- If $2 \le p \le n-1$, take one stone from pile $p-1$ and one stone from pile $p+1$, and put them into pile p (pile $p-1$ and pile $p+1$ must each have at least 1 stone).

What is the minimum number of operations required so that, for all $1 \le i \le n$, pile i contains exactly a_i stones? It can be proved that when $\sum a_i = n$, there always exists a valid finite sequence of operations.In addition, sometimes you also need to output the sequence of operations.

Input

The first line contains two integers n, typ ($3 \le n \le 2 \times 10^5$, typ $\in \{0, 1\}$). typ indicates whether you need to output the sequence of operations.

The next line contains n integers, representing a_i. It is guaranteed that $0 \le a_i \le n$ and $\sum a_i = n$.

Output

If typ $= 0$, output a single integer, the answer.

If typ $= 1$, you need to output two lines. The first line contains the answer ans. The second line contains ans integers, where the i-th integer represents the position chosen for the i-th operation.

It is guaranteed that when typ $= 1$**,** ans **does not exceed** 2×10^5**.**

Examples

standard input	standard output
5 1 0 2 1 2 0	3 3 2 4
10 0 0 0 5 2 2 1 0 0 0 0	103