Some 3-SUMs

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	256 megabytes

For an array of integers a, a *triplet* of elements is defined by 3 distinct indices (a_i, a_j, a_k), and their order does not matter.

Consider the triplets from a that sum to zero.

For example, the array $a = [1, 1, 1, 2, -2, 0]$ has 4 triplets that sum to zero: (a_1, a_2, a_5), (a_1, a_3, a_5), (a_2, a_3, a_5), and (a_4, a_5, a_6).

Given an integer k, construct an array of integers a such that exactly k triplets of elements from a sum to zero.

Your array must have size at most 5000.

Input

The single line of the input contains an integer k ($0 \le k \le 10^9$) — the required number of triplets that sum to zero.

Output

Output two lines.

On the first line, output a single integer n ($0 \le n \le 5000$) — the size of a.

On the second line, output n space-separated integers $a_1, \ldots, a_n$ ($-10^9 \le a_i \le 10^9$).

If there are multiple arrays satisfying these conditions, output any of them.

It can be shown that a valid answer always exists.

Examples

standard input	standard output
0	4 1 2 3 4
1	5 -1 1 -2 2 -3
4	6 1 1 1 2 -2 0

Note

In sample 1, no triples in the array a sum to zero.

In sample 2, the only triple that sums to zero is (a_2, a_4, a_5).