CODEFORCES β
Sponsored by Telegram

Codeforces Round #462 (Div. 2)

A. A Compatible Pair

time limit per test: 1 second
memory limit per test: 256 megabytes
input: standard input
output: standard output

Nian is a monster which lives deep in the oceans. Once a year, it shows up on the land, devouring livestock and even people. In order to keep the monster away, people fill their villages with red colour, light, and cracking noise, all of which frighten the monster out of coming.

Little Tommy has n lanterns and Big Banban has m lanterns. Tommy's lanterns have brightness $a_1, a_2, ..., a_n$, and Banban's have brightness $b_1, b_2, ..., b_m$ respectively.

Tommy intends to hide one of his lanterns, then Banban picks one of Tommy's non-hidden lanterns and one of his own lanterns to form a pair. The pair's brightness will be the product of the brightness of two lanterns.

Tommy wants to make the product as small as possible, while Banban tries to make it as large as possible.

You are asked to find the brightness of the chosen pair if both of them choose optimally.

Input

The first line contains two space-separated integers n and m ($2 \le n, m \le 50$).

The second line contains n space-separated integers $a_1, a_2, ..., a_n$.

The third line contains m space-separated integers $b_1, b_2, ..., b_m$.

All the integers range from -10^9 to 10^9.

Output

Print a single integer — the brightness of the chosen pair.

Examples

input	Copy
2 2 20 18 2 14	

output
252

input	Copy
5 3 -1 0 1 2 3 -1 0 1	

output
2

Note

In the first example, Tommy will hide 20 and Banban will choose 18 from Tommy and 14 from himself.

In the second example, Tommy will hide 3 and Banban will choose 2 from Tommy and 1 from himself.

B. A Prosperous Lot

time limit per test: 1 second
memory limit per test: 256 megabytes
input: standard input
output: standard output

Apart from Nian, there is a daemon named Sui, which terrifies children and causes them to become sick. Parents give their children money wrapped in red packets and put them under the pillow, so that when Sui tries to approach them, it will be driven away by the fairies inside.

Big Banban is hesitating over the amount of money to give out. He considers loops to be lucky since it symbolizes unity and harmony.

He would like to find a positive integer n not greater than 10^{18}, such that there are exactly k *loops* in the decimal representation of n, or determine that such n does not exist.

A *loop* is a planar area enclosed by lines in the digits' decimal representation written in Arabic numerals. For example, there is one loop in digit 4, two loops in 8 and no loops in 5. Refer to the figure below for all exact forms.

Input

The first and only line contains an integer k ($1 \leq k \leq 10^6$) — the desired number of loops.

Output

Output an integer — if no such n exists, output -1; otherwise output any such n. In the latter case, your output should be a **positive** decimal integer not exceeding 10^{18}.

Examples

input	Copy
2	

output
462

input	Copy
6	

output
8080

C. A Twisty Movement

time limit per test: 1 second
memory limit per test: 256 megabytes
input: standard input
output: standard output

A dragon symbolizes wisdom, power and wealth. On Lunar New Year's Day, people model a dragon with bamboo strips and clothes, raise them with rods, and hold the rods high and low to resemble a flying dragon.

A performer holding the rod low is represented by a 1, while one holding it high is represented by a 2. Thus, the line of performers can be represented by a sequence $a_1, a_2, ..., a_n$.

Little Tommy is among them. He would like to choose an interval $[l, r]$ $(1 \le l \le r \le n)$, then reverse $a_l, a_{l+1}, ..., a_r$ so that the length of the longest non-decreasing subsequence of the new sequence is maximum.

A non-decreasing subsequence is a sequence of indices $p_1, p_2, ..., p_k$, such that $p_1 < p_2 < ... < p_k$ and $a_{p_1} \le a_{p_2} \le ... \le a_{p_k}$. The length of the subsequence is k.

Input

The first line contains an integer n $(1 \le n \le 2000)$, denoting the length of the original sequence.

The second line contains n space-separated integers, describing the original sequence $a_1, a_2, ..., a_n$ $(1 \le a_i \le 2, i = 1, 2, ..., n)$.

Output

Print a single integer, which means the maximum possible length of the longest non-decreasing subsequence of the new sequence.

Examples

input	Copy
4 1 2 1 2	

output
4

input	Copy
10 1 1 2 2 2 1 1 2 2 1	

output
9

Note

In the first example, after reversing $[2, 3]$, the array will become $[1, 1, 2, 2]$, where the length of the longest non-decreasing subsequence is 4.

In the second example, after reversing $[3, 7]$, the array will become $[1, 1, 1, 1, 2, 2, 2, 2, 1]$, where the length of the longest non-decreasing subsequence is 9.

D. A Determined Cleanup

time limit per test: 1 second
memory limit per test: 256 megabytes
input: standard input
output: standard output

In order to put away old things and welcome a fresh new year, a thorough cleaning of the house is a must.

Little Tommy finds an old polynomial and cleaned it up by taking it modulo another. But now he regrets doing this...

Given two integers p and k, find a polynomial $f(x)$ with non-negative integer coefficients strictly less than k, whose remainder is p when divided by $(x + k)$. That is, $f(x) = q(x) \cdot (x + k) + p$, where $q(x)$ is a polynomial (not necessarily with integer coefficients).

Input

The only line of input contains two space-separated integers p and k ($1 \le p \le 10^{18}$, $2 \le k \le 2\,000$).

Output

If the polynomial does not exist, print a single integer -1, or output two lines otherwise.

In the first line print a non-negative integer d — the number of coefficients in the polynomial.

In the second line print d space-separated integers $a_0, a_1, ..., a_{d-1}$, describing a polynomial $f(x) = \sum_{i=0}^{d-1} a_i \cdot x^i$ fulfilling the given requirements. Your output should satisfy $0 \le a_i < k$ for all $0 \le i \le d - 1$, and $a_{d-1} \neq 0$.

If there are many possible solutions, print any of them.

Examples

input	Copy
46 2	

output
7 0 1 0 0 1 1 1

input	Copy
2018 214	

output
3 92 205 1

Note

In the first example, $f(x) = x^6 + x^5 + x^4 + x = (x^5 - x^4 + 3x^3 - 6x^2 + 12x - 23) \cdot (x + 2) + 46$.

In the second example, $f(x) = x^2 + 205x + 92 = (x - 9) \cdot (x + 214) + 2018$.

E. A Colourful Prospect

time limit per test: 1 second
memory limit per test: 256 megabytes
input: standard input
output: standard output

Firecrackers scare Nian the monster, but they're wayyyyy too noisy! Maybe fireworks make a nice complement.

Little Tommy is watching a firework show. As circular shapes spread across the sky, a splendid view unfolds on the night of Lunar New Year's eve.

A wonder strikes Tommy. How many regions are formed by the circles on the sky? We consider the sky as a flat plane. A region is a connected part of the plane with positive area, whose bound consists of parts of bounds of the circles and is a curve or several curves without self-intersections, and that does not contain any curve other than its boundaries. Note that exactly one of the regions extends infinitely.

Input

The first line of input contains one integer n ($1 \le n \le 3$), denoting the number of circles.

The following n lines each contains three space-separated integers x, y and r ($-10 \le x, y \le 10$, $1 \le r \le 10$), describing a circle whose center is (x, y) and the radius is r. No two circles have the same x, y and r at the same time.

Output

Print a single integer — the number of regions on the plane.

Examples

input Copy

```
3
0 0 1
2 0 1
4 0 1
```

output

```
4
```

input Copy

```
3
0 0 2
3 0 2
6 0 2
```

output

```
6
```

input Copy

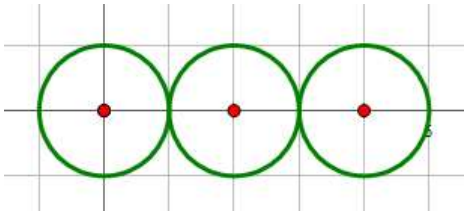
```
3
0 0 2
2 0 2
1 1 2
```

output

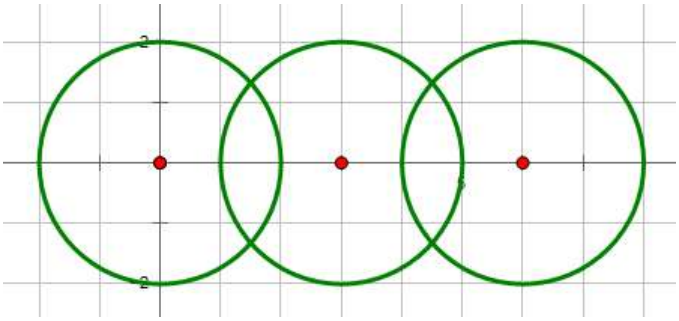
```
8
```

Note

For the first example,



For the second example,



For the third example,