Hong Kong Olympiad in Informatics 2013
IOI/NOI Team Formation Test

Task Overview

Task	CPU time limit	Score	Type
Fun Enumeration	1 second	100	Batch
A game with Laura	1 second	100	Batch
Polary's Busiest Day	1 second	100	Batch
The Forgotten Triangle	1 second	100	Interactive

Notice:
C++ programmers should be aware that using C++ streams (cin/cout) may lead to I/O bottlenecks and substantially slower performance.

C/C++ programmers should use "%lld" for 64-bit integers I/O.

Some test cases are grouped. You may need to pass all test cases in that group to get points.

Fun Enumeration

Problem

Recently there is a new real-time strategy game released - the *Age of Fun Enumeration*. Polary is a fan of the game. He is studying the following scenario.

The game starts with s unit of resources and no villagers. At each second, every villager will produce p unit of resources, and the produced resources are gathered at the end of the second. The player may also produce a single villager by using up c unit of resources. To do this, he needs to have at least c unit of resources in stock at the beginning of the second, and the villager will be produced at the end of the second. Notice that the player may produce at most 1 villager every second.

Polary is wondering how long it takes to have at least e unit of resources in stock, if played optimally. Help him.

Input

Four integers s, e, c, p on the first line.

Output

Output the minimum time on a single line. The input guarantees that it fits in a 64-bit signed integer.

Sample test

Input	Output
3 20 2 2	7

Input	Output
4 10 2 3	4

Input	Output
2 91 1 5	7

Constraints

In 50% test cases, $1 \leq s, e, c, p \leq 5000$
In all test cases, $1 \leq c \leq s < e \leq 2 \times 10^9, 1 \leq p \leq 2 \times 10^9$

A Game with Laura

Problem

Despite the heavy workload Dr. Jones needed to deal with in the hospital, once he went back home, he still squeezed part of his sleeping time to play with his daughter Laura:

"My lovely daughter, let's play a game!"
"What's the game, Dad?"
"I'm going to write 9 numbers on a piece of paper."

Then Dr. Jones wrote "10 6 8 -9 25 4 3 7 20' on the paper.

"I'll also write 1 to 5 in a random order."

He wrote "4 1 3 2 5" and then he continued to say:

"Can you see that the five numbers 6 -9 4 3 7 belongs to part of the long sequence I've written?"

"Yea. But what's the point of this?"
"Here we've got 4 1 3 2 5. The second number 1 is the smallest, so as the -9 there. The fourth number 2 is the second-smallest, so as the 3 there..."
"Oh, I see! The ranks of the numbers 6 -9 4 3 7 are exactly the same as 4 1 3 2 5! Am I right, Dad?"
"You're so smart! Can you find another 5-number subsequence that exactly has the same characteristic?"
"Hmm... Ah! 6 -9 4 3 20!"
"Bingo!" said Dr. Jones, while stroking his daughter's hair.

Formally, given a sequence a of n numbers and a permutation p of 1 to 5 , you should find a subsequence of a: $a_{k_0}, a_{k_1}, a_{k_2}, a_{k_3}, a_{k_4}$, such that $a_{k_i} < a_{k_j}$ if and only if $p_i < p_j$.

Input

The first line contains a single integer, n. The second line contains a sequence of n integers, the sequence a. The third contains another sequence of 5 integers p_0, p_1, p_2, p_3, p_4, a permutation of $\{1, 2, 3, 4, 5\}$.

Output

Print the required subsequence in a single line separated by single space. If multiple solutions exist, you may output any one of them. If a solution does not exist, output '-1' (without quotes).

Sample test

Input	Output
9 10 6 8 -9 25 4 3 7 20 4 1 3 2 5	6 -9 4 3 7

Input	Output
7 1 2 3 4 5 6 7 5 4 3 2 1	-1

Constraints

In 30% of the test cases, $n \leq 40$
In 70% of the test cases, $n \leq 300$
In all the test cases, $|a_i| \leq 10^9, 5 \leq n \leq 1000$

Polary's busiest day

Problem

There are n parties in the town today. Polary wants to attend all of them. The i-th party starts at time S_i and ends at E_i. According to the traditions in the town, one needs to stay at the party for at least half of the party time **continuously** to be counted as 'attended'.

Of course, Polary cannot attend two parties at the same time. However, he can join or leave a party at any time, and show up at another party immediately after he attended the previous one.

Can you help Polary to determine whether he can attend all parties or not?

Input

The first line contains a single integer, n.
In the next n lines, each line contains two integers S_i and E_i.

Output

Print 'YES' if Polary can attend all parties, otherwise, print 'NO'.

Sample test

Input	Output
3 1 5 3 6 5 6	YES

Input	Output
3 1 6 2 5 3 4	NO

Input	Output
3 1 6 3 5 4 5	YES

Explanation

In the first sample, Polary can attend the first party at $[1, 3]$, the second party at $[3, 4.5]$ and the last one at $[5, 5.5]$.

Constraints

In 50% of the test cases, $n \leq 10$
In all the test cases, $1 \leq n \leq 10^5, 0 \leq S_i < E_i \leq 10^9$

The Forgotten Triangle

Problem

Once upon a time, there was a **non-obtuse** triangle on the 2 dimensional plane. However, after years, its actual position was lost. As a adventurer, you are going to find the area of the triangle.

Formally speaking, this is a non-obtuse triangle T with **integer** coordinates (x_0, y_0), (x_1, y_1), (x_2, y_2) and it has **strictly positive area**. It is also known that the coordinates are bounded by B. That is, all x_i, y_i are integers in the range $-B$ to B.

You can make **queries** to gain information. To make a query, you specify the coordinates of an arbitrary triangle P. The answer of the query is 1 if the common area of triangle T and P is nonzero, otherwise the answer is 0.

Your task is to find $2 \times area(T)$ by making queries.

Input and Output

Interactions are done using standard I/O. Your program should first read the integer B. To make a query, print

```
Q X0 Y0 X1 Y1 X2 Y2
```

to standard output. Notice that `X0, Y0, X1, Y1, X2, Y2` should fit in a signed 32-bit integer. Then your program should read the result (0 or 1) from standard input. Finally, your program should print a line

```
A R
```

to standard output, where `R` is twice the area of triangle T. Notice that `R` should fit in a signed 64-bit integer. Then your program should terminate.

Your program should flush the output after writing every line to it. C/C++ users can use `fflush(stdout);` Pascal users can use `flush(Output);`

Sample

Input	Output	Explanation
1		$B = 1$
	`Q 0 0 0 1 -1 0`	Making query using triangle P with coordinates $(0,0), (0,1), (-1,0)$
0		T does not share common area with P
	`Q -1 1 0 1 -1 0`	Making query using triangle P with coordinates $(-1,1), (0,1), (-1,0)$
1		T shares a strictly positive common area with P
	`A 1`	The program knows T must be $(-1,1), (0,1), (-1,0)$, with area 0.5, hence answers 1

Hints

To help you test your program, we provide a program that helps you check whether two triangle T and P share a positive common area. You can find a program called `intersect` in the specific folder.

To use the program, type

```
./intersect x0 y0 x1 y1 x2 y2 x3 y3 x4 y4 x5 y5
```

in the terminal. The program writes 1 to stdout if the triangle with vertices (x0, y0), (x1, y1), (x2, y2) shares a positive common area with the triangle with vertices (x3, y3), (x4, y4), (x5, y5), otherwise it prints 0 to stdout.

Scoring

There are different subtasks in this problem. You should use at most Q queries at each subtask. The coordinates of T is within the range $-B$ to B (inclusively).

Subtask 1 (25 points)

- $B = 50$
- $Q = 50000$

Subtask 2 (15 points)

- $B = 2000$
- $Q = 50000$

Subtask 3 (15 points)

- $B = 5000$
- $Q = 50000$

Subtask 4 (45 points)

- $B = 10^9$
- $Q = 350$