

icpc International Collegiate Programming Contest



The 2026 ICPC Latin America Championship

Official Problem Set

March 7, 2026

This problem set contains 12 problems; pages are numbered from 1 to 19.

v1.0



icpc global sponsor
programming tools



icpc diamond
multi-regional sponsor

General Information

Unless otherwise stated, the following conditions hold for all problems.

Program Name

1. Your solution must be called `codename.c`, `codename.cpp`, `codename.java`, `codename.kt`, `codename.py3`, where `codename` is the capital letter which identifies the problem.

Input

1. The input must be read from standard input.
2. The input consists of a single test case, which is described using a number of lines that depends on the problem. No extra data appear in the input.
3. When a line of data contains several values, they are separated by *single* spaces. No other spaces appear in the input. There are no empty lines.
4. The English alphabet is used. There are no letters with tildes, accents, diaereses or other diacritical marks (ñ, Ã, é, Ì, ô, Û, ç, etcetera).
5. Every line, including the last one, has the usual end-of-line mark.

Output

1. The output must be written to standard output.
2. The result of the test case must appear in the output using a number of lines that depends on the problem. No extra data should appear in the output.
3. When a line of results contains several values, they must be separated by *single* spaces. No other spaces should appear in the output. There should be no empty lines.
4. The English alphabet must be used. There should be no letters with tildes, accents, diaereses or other diacritical marks (ñ, Ã, é, Ì, ô, Û, ç, etcetera).
5. Every line, including the last one, must have the usual end-of-line mark.

Problem A – Ants on a Ring

Author: Arthur Nascimento, Brasil

There are N ants on a circle. The circle has L spots numbered from 1 to L clockwise, with spot L adjacent to spot 1. Ants start at different spots, and want to reach different spots. To do so, in one second each ant may stay still or move around the circle to an adjacent spot (clockwise or counterclockwise).

No two ants can be at the same position on the circle at the same time, even between spots. For instance, suppose that during a second an ant moves clockwise from spot 1 to spot 2. During that second, other ants cannot do any of the following:

- Stay still at spot 2 (ants would meet at spot 2).
- Move counterclockwise from spot 3 to spot 2 (ants would meet at spot 2).
- Move counterclockwise from spot 2 to spot 1 (ants would meet between spots 1 and 2).

Determine whether it is possible for all ants to reach their targets, and if so, the minimum number of seconds required. That is, find the minimum t such that after t seconds, every ant can be at its target spot.

Input

The first line contains two integers N ($1 \leq N \leq 1000$) and L ($1 \leq L \leq 10^9$), indicating respectively the number of ants and the number of spots.

The second line contains N different integers A_1, A_2, \dots, A_N ($1 \leq A_i \leq L$ for $i = 1, 2, \dots, N$), where A_i is the initial spot of the i -th ant.

The third line contains N different integers B_1, B_2, \dots, B_N ($1 \leq B_i \leq L$ for $i = 1, 2, \dots, N$), such that B_i is the target spot of the i -th ant.

Output

Output a single line with an integer indicating the minimum time required for all ants to reach their targets, or the character “*” (asterisk) if it is impossible.

Sample Input 1

```
2 2
2 1
2 1
```

Sample Output 1

```
0
```

Explanation of Sample 1:

The two ants start at their target spots, so the answer is 0.

Sample Input 2

```
2 2
1 2
2 1
```

Sample Output 2

```
1
```

Explanation of Sample 2:

The two ants can move simultaneously either clockwise or counterclockwise, reaching their targets after just 1 second.

Sample Input 3

```
1 10
1
7
```

Sample Output 3

```
4
```

Sample Input 4

```
3 5
1 3 2
5 2 4
```

Sample Output 4

```
*
```

Sample Input 5

```
3 5
1 3 2
4 2 5
```

Sample Output 5

```
2
```

Explanation of Sample 5:

All three ants can move counterclockwise, with the second ant staying still for a second.

Problem B – Booksort

Author: João Victor Ayalla, Brasil

Beatriz always enjoyed reading, so she decided to open a bookstore to have books around her all day long. She wants to be sure that the books are properly organized, so she can attract many customers.

On a large shelf of the bookstore there are N stacks of books in a row, with the stacks numbered from 1 to N left to right. Stack i contains A_i books. Beatriz would like the numbers of books to be in non-decreasing order, that is, $A_i \leq A_{i+1}$ for $i = 1, 2, \dots, N - 1$, which might require some rearrangement of the books.

However, Beatriz is lazy and she really does not feel like organizing the books by herself. Then she asked her best friend Bernardo for help. They agreed that Beatriz will give Bernardo a sequence of commands. In each command Beatriz will specify two distinct stacks i and j , and Bernardo will take the $s = A_i + A_j$ books in those stacks, rearranging them as evenly as possible. This means that after performing the command, the number of books in those stacks will be updated in the following way:

$$A_i = \left\lfloor \frac{s}{2} \right\rfloor, \quad A_j = \left\lceil \frac{s}{2} \right\rceil.$$

Beatriz does not want to spend a lot of time with Bernardo moving around the books for her. She wants a sequence of at most 10^5 commands that yields the desired non-decreasing order. But, you know, Beatriz does not want to decide the commands by herself. Can you prepare any valid sequence of commands for her?

Input

The first line contains an integer N ($2 \leq N \leq 5000$) indicating the number of book stacks on the shelf. Each stack is identified by a distinct integer from 1 to N .

The second line contains N integers A_1, A_2, \dots, A_N ($1 \leq A_i \leq 10^5$), where A_i is the initial number of books in stack i .

Output

The first line must contain an integer k ($0 \leq k \leq 10^5$) indicating the number of commands to perform.

Each of the next k lines must describe a command with two integers i and j ($1 \leq i, j \leq N$ and $i \neq j$), representing that the books in stacks i and j are rearranged as described. After performing all the commands in the order in which they appear in your answer, the shelf must be sorted in non-decreasing order.

It can be proven that a valid answer exists under the given constraints. If there are multiple solutions, output any of them; there is no need to minimize k .

Sample Input 1

```
3
1 1 1
```

Sample Output 1

```
0
```

Explanation of Sample 1:

Since the shelf is initially sorted in non-decreasing order, an empty sequence of commands is a valid output.

Sample Input 2

```
3
1 1 1
```

Sample Output 2

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

Explanation of Sample 2:

Although the shelf is initially sorted, the output is valid because the shelf ends sorted after at most 10^5 commands. There is no need to minimize the number of commands.

Sample Input 3

```
5
14 7 13 8 15
```

Sample Output 3

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

Explanation of Sample 3:

The first command ($i = 2, j = 1$) changes the shelf from [14, 7, 13, 8, 15] to [11, 10, 13, 8, 15].

The second command ($i = 1, j = 4$) changes the shelf from [11, 10, 13, 8, 15] to [9, 10, 13, 10, 15].

The third command ($i = 3, j = 4$) changes the shelf from [9, 10, 13, 10, 15] to [9, 10, 11, 12, 15], which is sorted in non-decreasing order.

Problem C – Crop Circles

Author: Giovanna Kobus, Brasil

After a series of unfortunate events, dubious investments, and catastrophically failed marriages, Camila has decided to leave Earth for good. She does not have enough money to build a rocket, so her best hope is to be abducted by aliens. Camila asked an AI chatbot about it, and then she independently came to the conclusion that the best way to be abducted by aliens is by making a circle of crops.

The good news is that among Camila's poor investments there is a very large farm, which can be represented as the 2D plane. There are N sprinklers located on the farm, and each of them waters all plants within a radius that depends on its power. Camila's plan is to plant corn seeds in a perfect circle. Since the soil of the farm is very dry, the whole perimeter of Camila's circle must be watered by some sprinkler. Notice that Camila only needs to water the perimeter of the circle, while areas inside of it may remain dry.

To maximize her abduction chances, Camila's circle needs to be as large as possible. Since you are one of her few remaining friends, she asks for your help to determine what is the largest circle of corn she can plant on her farm, given the position and radius of each sprinkler.

Input

The first line contains an integer N ($1 \leq N \leq 40$) indicating the number of sprinklers.

Each of the next N lines describes a sprinkler with three integers X , Y and R ($-1000 \leq X, Y \leq 1000$ and $1 \leq R \leq 1000$), indicating that the sprinkler has coordinates (X, Y) and waters all plants within a radius of R , inclusive.

No two sprinklers have the same location.

Output

Output a single line with the maximum radius of a crop circle Camila can build. Notice that although the sprinklers have integer coordinates and radii, the coordinates and radius of the crop circle are not required to be integers. The output must have an absolute or relative error of at most 10^{-6} .

Sample Input 1

```
4
1 1 1
-1 1 1
-1 -1 1
1 -1 1
```

Sample Output 1

```
1
```

Explanation of Sample 1:

Camila can choose among five crop circles of radius 1. Four of them are centered at the sprinklers, while the remaining one is centered at $(0, 0)$.

Sample Input 2

```
1
3 1 415
```

Sample Output 2

```
415
```

Sample Input 3

```
4
1 0 2
-1 0 2
0 1 2
0 -1 2
```

Sample Output 3

```
2.57793547457
```

Problem D – Dropshipping

Author: Paulo Cezar Pereira Costa, Brasil

Drew P. Shipper received N customer requests for products sold at a store called EvenBuy, where all product prices are even integers. To fulfill the i -th request, Drew must purchase a specific product that costs A_i at EvenBuy, and Drew has already charged this price to the customer.

Drew is doing this for profit, based on a loyalty program that EvenBuy has: after Drew makes X full-price purchases, he gets a discount of 50% in his next single purchase. This discounted purchase does not count toward the X full-price purchases required to earn the next discount. Because Drew's customers paid him the full price upfront, he keeps the amount of the discount as his own profit whenever he purchases a discounted item.

Drew can decide the order of his purchases at EvenBuy to maximize his total profit. However, delivery times are directly related to the order of the purchases, and to avoid customer complaints, he cannot delay a purchase too much. In practice, Drew must fulfill the i -th request within his first $i + K$ purchases at EvenBuy.

Your task is to find the maximum total profit Drew can achieve. Each purchase fulfills exactly one request and each request must be fulfilled exactly once.

Input

The first line contains three integers N , X ($1 \leq N, X \leq 2 \times 10^5$) and K ($0 \leq K \leq 2 \times 10^5$), indicating respectively the number of customer requests, the number of full-price purchases required to get a 50% discount, and the delivery limit (the i -th request must be fulfilled within the first $i + K$ purchases).

The second line contains N even integers A_1, A_2, \dots, A_N ($2 \leq A_i \leq 10^9$), representing the prices of the requested products.

Output

Output a single line with an integer indicating the maximum total profit Drew can achieve.

Sample Input 1

```
3 1 0
6 4 14
```

Sample Output 1

```
2
```

Explanation of Sample 1:

Since $K = 0$, the i -th request must be fulfilled with the i -th purchase. As only the second purchase is discounted, Drew can only achieve a profit of $4/2 = 2$.

Sample Input 2

```
3 1 1
6 4 14
```

Sample Output 2

```
7
```

Explanation of Sample 2:

Now the i -th request must be fulfilled within the first $i + 1$ purchases. So the valid purchase orders are $[6, 4, 14]$, $[6, 14, 4]$, $[4, 6, 14]$ and $[14, 6, 4]$. The order $[6, 14, 4]$ is the best, as Drew can achieve a profit of $14/2 = 7$ on the second purchase.

Problem E – Eye Exam

Author: Giovanna Kobus, Brasil

During the warmup session of Programadores de América (PDA) in Santiago de Chile, Ezequiel notices something alarming: the scoreboard looks blurry. Worried that the same thing might happen during the actual contest, he decides to quickly take an eye exam.

The optometrist runs a series of tests to find Ezequiel’s true prescription, which is an integer x . Unfortunately, instead of carefully narrowing down the prescription, the optometrist uses the lenses in no particular order. Thus, in each test Ezequiel is shown two lenses with different correction degrees A and B ($A < B$), and he must tell his opinion about which lens is better. For each test Ezequiel answers:

- “A” if he thinks that A is strictly closer to x than B .
- “B” if he thinks that B is strictly closer to x than A .
- “E” if he thinks that A and B are equally close to x .

After running the tests in such a strange way, the optometrist does not know what to do with Ezequiel’s answers, so they need your help. Given the results of all tests, you must determine the minimum and maximum integer prescriptions that are consistent with the tests. If no integer prescription is consistent with the tests, you must report it.

Input

The first line contains an integer N ($1 \leq N \leq 1000$) indicating the number of tests.

Each of the next N lines describes a test with two integers A and B ($1 \leq A < B \leq 1000$), followed by an uppercase letter C (either “A”, “B” or “E”), where A and B are the correction degrees of the lenses and C is Ezequiel’s answer.

It is guaranteed that there is a finite number of integer prescriptions that are consistent with the tests.

Output

Output a single line with the character “*” (asterisk) if no integer prescription is consistent with the tests. Otherwise, output a single line with two integers indicating the minimum and maximum prescriptions that are consistent with the tests.

Sample Input 1

```
1
1 31 E
```

Sample Output 1

```
16 16
```

Explanation of Sample 1:

The only integer prescription that is equally close to 1 and 31 is 16.

Sample Input 2

```
2
1 31 A
1 31 B
```

Sample Output 2

```
*
```

Explanation of Sample 2:

No integer prescription is consistent with the tests, because it should be strictly closer to both 1 and 31.

Sample Input 3

```
3
1 5 B
1 11 A
1 5 B
```

Sample Output 3

```
4 5
```

Sample Input 4

```
1
3 4 E
```

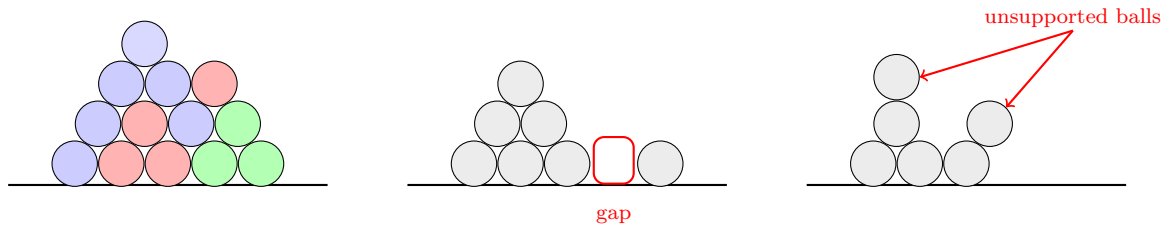
Sample Output 4

```
*
```

Problem F – Fun with Balls

Author: Roberto Sales, Brasil

Antonella has N colored balls, and she wants to build a *stable* pile of balls with them. For a pile to be stable, every ball must either lie on the ground, or be placed exactly on top of two other balls. Also, balls lying on the ground must be tightly packed, which means that there should be no empty space between them. The picture below shows three arrangements of balls; the one on the left is a stable pile, while the other two are failed attempts.



Antonella wants to build her stable pile in an incremental process. First, she will pick a ball and place it on the ground, forming a stable pile made of a single ball. Then, she will add the other $N - 1$ balls to the pile, one at a time, keeping the pile stable at every step. If there are multiple places where Antonella could add a ball, she will choose the highest place (relative to the ground). If there are still multiple options, she can choose any of them.

A set of balls in a stable pile is considered *connected* if, for any two balls s and t in the set, there is a sequence of balls $s = b_0, b_1, \dots, b_k = t$ such that b_{i-1} touches b_i for $i = 1, 2, \dots, k$. A *cluster* is a connected set of balls of the same color. The *size* of a cluster is the number of balls in it. The stable pile in the picture above has two red clusters of sizes 3 and 1, a green cluster of size 3, and a blue cluster of size 6.

Given the colors of the balls in the order in which Antonella will use them, you must tell the maximum size among all clusters of all possible stable piles that she can build.

Input

The first line contains an integer N ($1 \leq N \leq 150$) indicating the number of balls.

The second line contains N integers K_1, K_2, \dots, K_N ($1 \leq K_i \leq 150$ for $i = 1, 2, \dots, N$), where K_i is the color of the i -th ball that Antonella will add to her pile.

Output

Output a single line with an integer indicating the maximum size among all clusters of all possible stable piles that Antonella can build.

Sample Input 1

```
3
1 2 1
```

Sample Output 1

```
2
```

Explanation of Sample 1:

Antonella will place the first ball (color 1) on the ground. She will place the second ball (color 2) on the ground too, but it can be either on the left or on the right of the first ball. When inserting the third ball (color 1), she will place it on top of the first two balls. Regardless of the choice Antonella makes for the second ball, she will get a stable pile with a cluster of size 2 (color 1) and a cluster of size 1 (color 2). Thus, the maximum size among all clusters of all possible stable piles is 2.

Sample Input 2

```
3
1 1 1
```

Sample Output 2

```
3
```

Sample Input 3

```
4
1 5 5 1
```

Sample Output 3

```
2
```

Sample Input 4

```
6
1 2 2 1 2 3
```

Sample Output 4

```
3
```

Problem G – GATA-CAT

Author: Humberto Díaz Suárez, Puerto Rico

A DNA sequence is a molecular chain composed of the four nucleobases: cytosine (C), guanine (G), adenine (A), and thymine (T). Those letters can be used to encode a genetic sequence as text.

Previously, we hypothesized that felines owe their traits to what we call their *CAT degree*. It's the number of times that their genetic sequence contains the nucleobases C-A-T in that order, ignoring other nucleobases between them. For instance, the sequences “GACT”, “GCAT”, and “CCGAAGT” have CAT degrees 0, 1, and 4, respectively.

It turns out that was only half the picture. Researchers from Latin America were surprised to find healthy cats with below-average CAT degrees. Upon further study, they determined that there is a second factor regulating feline traits, now known as *GATA degree*. Similar to CAT degree, it's the number of times that a genetic sequence contains the nucleobases G-A-T-A in that order, ignoring other nucleobases between them.

Today you'll be crafting fresh DNA sequences in the lab. We will ask you for short genetic sequences (at most 500 nucleobases) having specific GATA and CAT degrees. Please write a program to assist in our research.

Input

The first line contains an integer Q ($1 \leq Q \leq 1000$) indicating the number of sequences that must be crafted.

Each of the next Q lines contains two integers G and C ($0 \leq G, C \leq 10^6$), representing respectively the required GATA and CAT degrees.

Output

For each request in the input, output a line with a non-empty string of at most 500 uppercase letters “C”, “G”, “A”, or “T”, having the specified GATA and CAT degrees.

It can be proven that a valid answer exists under the given constraints. If there are multiple solutions, output any of them; there is no need to minimize the length of the strings.

Sample Input 1

```
4
1 1
2 3
18 1
2 1
```

Sample Output 1

```
CGATA
ATGGCATATC
GGAAATTTTCAT
GATACAT
```

Sample Input 2

```
5
1 0
0 1
1 1
0 0
0 0
```

Sample Output 2

```
GATA
CAT
CGATA
T
TTTTTTTTTTT
```

Problem H – Holes and Tunnels

Author: Marcos Kolodny, Argentina

Path-Digging Animals (PDA) is a new cooperative game for two players that is gaining popularity among programmers. The game is played on a board having N holes and $N - 1$ bidirectional tunnels. The tunnels form an undirected tree, which means that there is a unique simple path between each pair of holes.

A *route* in PDA is a simple path between two different holes. Since tunnels are bidirectional, the direction of a route is not relevant: the (unique) route from hole U to hole V and the (unique) route from hole V to hole U are the same route.

Players use animal-shaped pieces for playing PDA. In each round of the game, each player independently chooses a route which is traversed by their animal, and both players score as many points as the number of tunnels that their routes have in common.

Right now Alicia and Bruno are at a board game party in Chile playing PDA, and they want to analyze their scoring possibilities. They would like to know, for each integer k from 1 to $N - 1$, how many ordered pairs of routes (A, B) there are such that Alicia and Bruno score exactly k points if Alicia chooses route A and Bruno chooses route B .

Input

The first line contains an integer N ($2 \leq N \leq 2 \times 10^5$) indicating the number of holes. Each hole is identified by a distinct integer from 1 to N .

Each of the next $N - 1$ lines contains two integers U and V ($1 \leq U, V \leq N$ and $U \neq V$), representing that there is a bidirectional tunnel between holes U and V . It is guaranteed that the tunnels form an undirected tree.

Output

Output a single line with $N - 1$ integers P_1, P_2, \dots, P_{N-1} , where P_k indicates the number of ordered pairs of routes (A, B) such that Alicia and Bruno score exactly k points if Alicia chooses route A and Bruno chooses route B . Because these numbers can be very large, output the remainder of dividing each of them by 998244353.

Sample Input 1

```
3
1 3
2 3
```

Sample Output 1

```
6 1
```

Explanation of Sample 1:

There are three possibilities for Alicia's route A : the route between holes 1 and 3 (formed by the single tunnel between these holes), the route between holes 2 and 3 (again a single tunnel), and the route between holes 2 and 1 (containing both tunnels). The same three possibilities are available for Bruno's route B . The table below shows the score for each combination. As can be seen, 6 ordered pairs of routes yield a score of 1 point, while 1 pair gives 2 points. Applying the modulo operation does not change these results.

		Bruno		
		1-3	2-3	2-3-1
Alicia	1-3	1	0	1
	2-3	0	1	1
	2-3-1	1	1	2

Problem I – Inversion Game

Author: Arthur Nascimento, Brasil

Evelyn and Todd play a game with a multiset S of integers, and an array v which is initially empty. After deciding who plays first, they take turns alternately. In each turn, the corresponding player chooses any element of S , appends it to the end of v , and removes it from S .

The game ends as soon as S is empty. At that moment, the number of inversions in v is counted, that is, the number of pairs of indices $i < j$ such that $v_i > v_j$. If there are an even number of inversions then Evelyn is the winner, while Todd wins if the number of inversions is odd.

Evelyn and Todd have been playing the game for quite some time, and now both play optimally. Thus, for a given multiset S , there are four possible outcomes:

- Evelyn wins, regardless of who plays first.
- Todd wins, regardless of who plays first.
- The first player wins, regardless of who they are.
- The second player wins, regardless of who they are.

Given S , your task is to find which of the four cases will occur.

Input

The first line contains an integer N ($1 \leq N \leq 10^5$) indicating the number of elements of the multiset S .

The second line contains N integers S_1, S_2, \dots, S_N ($1 \leq S_i \leq N$ for $i = 1, 2, \dots, N$), representing the elements of S .

Output

Output a single line with an uppercase letter indicating the outcome of the game, assuming that Evelyn and Todd play optimally. The letter must be:

- “E” if Evelyn wins;
- “T” if Todd wins;
- “F” if the first player wins; and
- “S” if the second player wins.

Sample Input 1

```
3
1 1 1
```

Sample Output 1

```
E
```

Explanation of Sample 1:

No matter how Evelyn and Todd choose the elements of $S = \{1, 1, 1\}$, the resulting array will be $v = [1, 1, 1]$. Since there are no inversions in v , Evelyn wins, regardless of who plays first.

Sample Input 2

```
3
3 1 2
```

Sample Output 2

```
S
```

Problem J – Jaime’s Palace

Author: Moroni Silverio Flores & Sarai Ramirez Gomez, Mexico

Churro, a chubby and energetic guinea pig, has just started a new job at a fancy restaurant called Jaime’s Palace. Since she is the newest employee, she is only responsible for washing and arranging the plates. Churro discovered that the most efficient way to keep the plates organized and easily accessible for the chef is by storing them in a single stack.

There are P plates in Jaime’s Palace, and K_i of them are used on the i -th day. Churro designed a rigid and highly logical system for managing the plates. On the i -th day she takes the top K_i plates from the stack and the chef uses each of them just once during the day. At the end of the day, Churro washes the K_i used plates and places them back on top of the stack in an arbitrary order.

Churro’s system is a real success. However, she is curious about the following questions. After repeating the above procedure for D days, can she be sure that there is a plate that was used at least t times? What is the maximum value of t that she can guarantee? Please help Churro to determine that value.

Input

The first line contains two integers P ($2 \leq P \leq 2000$) and D ($1 \leq D \leq 2000$), indicating respectively the number of plates and the number of days.

The second line contains D integers K_1, K_2, \dots, K_D ($1 \leq K_i \leq P$ for $i = 1, 2, \dots, D$), where K_i is the number of plates used on the i -th day.

Output

Output a single line with an integer indicating the maximum value of t such that after D days and for any reordering of the plates put back into the stack at the end of each day, there is a plate that was used at least t times.

Sample Input 1

```
10 3
1 1 2
```

Sample Output 1

```
3
```

Explanation of Sample 1:

On each of the first two days, Churro takes the top plate from the stack and places it back on top, so that plate is used twice. On the third day she takes the top two plates, so one of those plates is used three times, independently of the order in which the plates are placed back on top of the stack at the end of the third day.

Sample Input 2

```
10 4
5 3 5 2
```

Sample Output 2

```
3
```

Problem K – Kitten Greetings

Author: Martín Muñoz, Chile

Catarina loves all cats that live in her neighborhood. Her lifelong dream is to design a large cat-seeing circuit, so that every day she can go out and greet the cats while doing some exercise.

Catarina's neighborhood can be represented as the 2D plane, with the North-South direction aligned with the y -axis. A cat-seeing circuit that visits m cats consists of exactly m steps. Catarina chooses a starting position (x_0, y_0) and faces one of the four cardinal directions. For each step $i = 1, 2, \dots, m$ the following occurs:

- Catarina chooses a value $k_i > 0$ and walks k_i units from (x_{i-1}, y_{i-1}) straight ahead in her current direction, stopping at the location of a cat that she has not greeted during any previous step.
- Catarina greets the cat, taking some time to appreciate its beauty.
- Without turning around, Catarina walks another k_i units straight ahead in her current direction, stopping at a position (x_i, y_i) .
- Catarina turns 90° either clockwise or counterclockwise, facing again one of the four cardinal directions.

After completing all m steps, Catarina must be back to her starting position (x_0, y_0) , facing her initial direction. Notice that the length of the cat-seeing circuit is $\sum_{i=1}^m 2k_i$. When $m = 0$, the cat-seeing circuit has length 0.

Catarina knows the location of the N cats that live in her neighborhood. Surprisingly, no two cats have the same x -coordinate or the same y -coordinate. Your task is to determine the maximum length that a cat-seeing circuit can have.

Input

The first line contains an integer N ($1 \leq N \leq 4000$) indicating the number of cats.

Each of the next N lines describes a cat with two integers X and Y ($-10^8 \leq X, Y \leq 10^8$), representing that the cat has coordinates (X, Y) .

No two cats have the same x -coordinate or y -coordinate (they differ in both of them).

Output

Output a single line with an integer indicating the maximum length that a cat-seeing circuit can have.

Sample Input 1

```
5
1 2
2 1
0 0
-1 -2
-2 -1
```

Sample Output 1

```
0
```

Explanation of Sample 1:

In this case there is a circuit of length 16 that visits all the cats, but it is not a cat-seeing circuit because the cat at coordinates $(0, 0)$ is greeted twice.

Sample Input 2

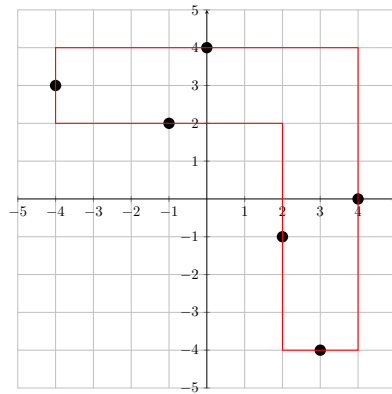
```
6
4 0
0 4
2 -1
-1 2
-4 3
3 -4
```

Sample Output 2

```
32
```

Explanation of Sample 2:

The picture below shows the location of the cats with small circles, together with a maximum-length cat-seeing circuit that visits all of them. The length of the circuit is 32.



Sample Input 3

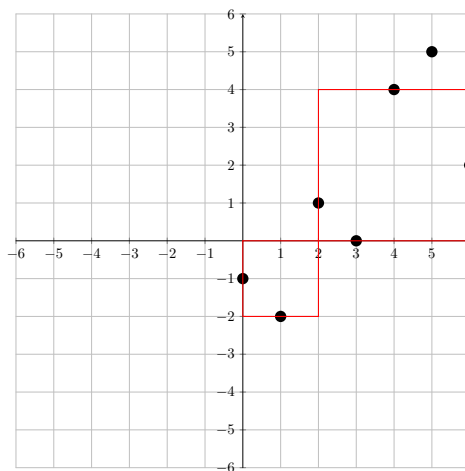
```
7
2 1
0 -1
5 5
3 0
4 4
6 2
1 -2
```

Sample Output 3

```
24
```

Explanation of Sample 3:

It is possible to visit $m = 6$ of the $N = 7$ cats with a cat-seeing circuit of length 24.



Problem L – Late and Disobedient

Author: Enrique Junchaya, Peru

Nathan is running late to the annual meeting of the Public Disobedience Association (PDA). He was driving there as fast as he could until he got stuck in front of a red light. As he is in a hurry, he wants to cross the street anyway (see Disclaimer). Of course, Nathan is not a monster, so he will cross the street only if there is enough space to cross it without hurting any pedestrian. At a given time T , Nathan has the “feeling” that he could cross the street safely, but Nathan’s feelings can be completely wrong!

The crosswalk has length L and can be modeled as a segment on the x -axis with endpoints at $x = 0$ and $x = L$. There are N pedestrians on the x -axis, not necessarily on the crosswalk. Each pedestrian has an initial position X and a constant velocity V (positive or negative); this means that at each time $t \geq 0$ the position of the pedestrian is $X + t \cdot V$.

Nathan’s car has width C and he can cross the street if there is a gap of width at least C between two consecutive pedestrians on the crosswalk or between a pedestrian and an endpoint of the crosswalk. He can also cross if there are no pedestrians on the crosswalk.

You will be given Q queries. Each query specifies an integer time $T \geq 0$ and you must decide whether Nathan can cross the street at that moment.

Disclaimer: The organization of Programadores de América (PDA) does not condone crossing red lights nor breaking the law. This is a fictional story and the behavior of Nathan does not represent the opinions of the organization of the competition.

Input

The first line contains three integers N , C and L ($1 \leq N \leq 1000$ and $1 \leq C \leq L \leq 10^9$), indicating respectively the number of pedestrians, the width of Nathan’s car and the length of the crosswalk.

Each of the next N lines describes a pedestrian with two integers X and V ($-10^9 \leq X, V \leq 10^9$ and $V \neq 0$), representing respectively the initial position and velocity of the pedestrian. No two pedestrians have the same initial position and velocity (they differ in at least one of them).

The next line contains an integer Q ($1 \leq Q \leq 2 \times 10^6$) denoting the number of queries.

Each of the next Q lines contains an integer T ($0 \leq T \leq 10^9$) indicating the time to be considered. Times are given in increasing order.

Output

Output a line for each query, with the uppercase letter “Y” if Nathan can cross the street at the corresponding time, and the uppercase letter “N” otherwise.

Sample Input 1

```
4 5 10
1 1
9 -1
-1 -1
11 1
3
0
3
4
```

Sample Output 1

```
Y
N
Y
```

Explanation of Sample 1:

In this sample there are $N = 4$ pedestrians, Nathan's car has width $C = 5$, and the crosswalk has length $L = 10$.

At time $T = 0$ the pedestrians are at positions $x_1 = 1$, $x_2 = 9$, $x_3 = -1$ and $x_4 = 11$, so Nathan can cross the street because there is a gap of width at least 5 between $x_1 = 1$ and $x_2 = 9$.

At time $T = 3$ the pedestrians are at positions $x_1 = 4$, $x_2 = 6$, $x_3 = -4$ and $x_4 = 14$, so Nathan cannot cross the street because no gap is wide enough.

Finally, at time $T = 4$ the pedestrians are at positions $x_1 = x_2 = 5$, $x_3 = -5$ and $x_4 = 15$, so Nathan can cross the street using either the gap between $x = 0$ and $x_1 = x_2 = 5$, or the gap between $x_1 = x_2 = 5$ and $x = L = 10$.