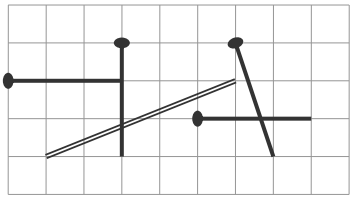
MATHEMATICAL GRAMMAR SCHOOL CUP
June, 30, 2016

TASK 1. PIN

Time limit: 0.5 sec

Memory limit: 512 MB



As you may already know, the double line in the picture is electrified bare conductor. It is unsafe to touch it, like every metal element exposed to the conductor. But, there are metal pins scattered around it.

Input

The total number of pins and the conductor is given in the first line of the standard input. In the following lines, each pin is represented on a separate line at the standard input with two pairs of numbers: *x* and *y* coordinates at the ends. Finally, the ends of the conductor are described in the last line at the standard input the same way. All given coordinates are integer numbers in range of [0, 10000]. There are no more than 1000 pins.

Output

Write on the standard output a single line with one number – the number of pins that are safer to touch with bare hands (meaning pins which don't have electrical connection to the conductor).

Example (represented by a picture above)

Input

```
5
5 2 8 2
3 4 3 1
7 1 6 4
3 3 0 3
6 3 1 1
```

Output

```
2
```

MATHEMATICAL GRAMMAR SCHOOL CUP
June, 30, 2016

TASK 2. SEPARATION

Time limit: 0.5 sec

Memory limit: 64 MB

A segment with length *n* cm must be separated on *m* fragments, which are of the length *a* cm, *b* cm or *c* cm. Write a program **`separation`**, which calculates in how many different ways the segment can be separated.

Input

In the first row of the standard input, the positive integers *a*, *b*, *c*, *m* and *n* ($a < b < c < 100$; $m*a < n < m*c$; $n < 10^{15}$) are given.

Output

Write on the standard output a single line with one number – the number of ways for separating of the segment.

Example

Input

```
2 7 9 20 100
```

Output

```
2
```

Explanation: The segment can be separated in two different ways:
1 way: 8 fragments to 2 cm, 12 fragments to 7 cm, 0 fragments to 9 cm;
2 way: 10 fragments to 2 cm, 5 fragments to 7 cm, 5 fragments to 9 cm.

MATHEMATICAL GRAMMAR SCHOOL CUP
June, 30, 2016

TASK 3. STUTTER

Time limit: 2 sec

Memory limit: 2 MB

Given an integer array with *N* elements such as that every element occurs 2 times except two elements which occur only once, how do you find as fast as possible (with respect to time and memory limit) the elements, which happen to occur only once in the array? Write a program **`stutter`**, able to solve this situation.

Input

The following lines are read from the standard input:

- line 1: the positive integer number *N* (array dimension)
- Each of the rest *N* lines contains one number of an array, no order at all. Each of these numbers, except the two that you need, occurs exactly twice. On the contrary, each of "unique" numbers occurs exactly once.

Output

Write one line containing the found unique numbers, increasingly ordered and space separated, to the standard output.

Constraints

Of course, number *N* is even, not less than two and not greater than one million (10^6). The numbers in the array are positive integers with no more than 18 decimal digits.

Evaluation

In test examples for 20 points no element value exceeds 100000. In other tests for 30 points more, element values do not exceed one milliard (10^9), while *N* is no more than 200000.

Example

Input

```
10
14
7
21
9
14
17
17
7
19
9
```

Output

19 21

MATHEMATICAL GRAMMAR SCHOOL CUP
June, 30, 2016

Task 4. PARKING LOT

Time limit: 1 sec

Memory limit: 64 MB

We have found an archive of used parking tickets from a parking lot. There are entrance and exit time written on each ticket. We know that the parking lot was sometimes full. You should determinate what was the total amount of time when the parking lot was full.

Input

First line: N – number of tickets ($0 < N < 100\,000$)

Next N lines: A, B – integer numbers representing entrance and exit time of each car ($0 < A < 10^9$, $0 < B < 10^9$, $A < B$)

Output

Single integer number – the total amount of time when the parking lot was full

Example

Input

4
10 20
20 30
15 50
40 100

Output

25

Explanation of the example

The maximal number of cars on the parking lot was 2 (when an exit time of a ticket is equal to the entrance time of another ticket, then we consider appropriate two cars was never on the parking lot together). This means that capacity of the parking lot is at least 2 cars, but since we know that the parking lot was sometimes full, it follows that the capacity of the parking lot is exactly 2 cars. The parking lot was full in the period 15 to 30 and in the period 40 to 50. The resulting amount of time is 30 – 15 + 50 – 40.