COCI 2018/2019

Round #2, November 17th, 2018

Tasks

Task	Time limit	Memory limit	Score
Preokret	1 s	64 MB	50
Kocka	1 s	64 MB	70
Deblo	1 s	64 MB	90
Maja	2 s	64 MB	110
Sunčanje	4 s	256 MB	130
Total			450

If they continue playing like this, soon the basketball players of the world's best league, the NBA, will put a ball in the basket every second. So there will no longer be any defense, tactics nor basketball.
Let's imagine observing one of these future matches between Team A and Team B. We know how many points both Team A and Team B have scored and the exact second when it happened. Within a second, it will not be possible to score more than one point.
King James is observing the task input and wants to answer the following two questions:

1. How many points have been scored during the first half-time, that is in the first half of the game, if we know that the entire game lasts 4 × 12 minutes?
2. How many "turnarounds" have happened during the match, i.e. how many times has a team come from a losing situation (has strictly fewer points scored than the other team) to a leading one (has strictly more points scored than the other team)?

INPUT

The first line contains a positive integer A ($1 \le A \le 2879$), the number of points Team A has scored.
In the following A lines there are positive integers As ($1 \le As \le 2880$), the seconds in which Team A was scoring points ordered from the smallest to the largest number.
In the $(A + 2)^{th}$ line there is a positive integer B ($1 \le B \le 2879$), the number of points Team B has scored. In the following B lines there are positive integers Bs ($1 \le Bs \le 2880$), the seconds in which Team B was scoring points ordered from the smallest to the largest number.

OUTPUT

In the first line print an integer value, the answer to the first question from the text of the task.
In the second line print an integer value, the answer to the second question from the text of the task.

SCORING

The correct output of the first line is worth 2 points, and the correct output of the second one is worth 3 points. If you do not know how to solve some part of the task, print anything in the corresponding line.

SAMPLE TESTS

input

```
3
10
1400
1500
2
7
2000
```

input

```
6
15
30
35
55
60
2065
7
20
25
```

input

```
11
1402
1412
1428
1430
1441
1444
1453
1483
1485
```

	40 45 50 2070 2075	1489 1490 9 1403 1405 1409 1435 1459 1460 1461 1487 1495
output	**output**	**output**
3 1	10 5	8 2

Clarification of the second example:

Results’ progress „Team A : Team B“	Turnaround (YES/NO)
1:0 (15th second, 1st half-time)	NO
1:1 (20th second, 1st half-time)	NO
1:2 (25th second, 1st half-time)	YES (Team B)
2:2 (30th second, 1st half-time)	NO
3:2 (35th second, 1st half-time)	YES (Team A)
3:3 (40th second, 1st half-time)	NO
3:4 (45th second, 1st half-time)	YES (Team B)
3:5 (50th second, 1st half-time)	NO
4:5 (55th second, 1st half-time)	NO
5:5 (60th second, 1st half-time)	NO
6:5 (2065th second, 2nd half-time)	YES (Team A)
6:6 (2070th second, 2nd half-time)	NO
6:7 (2075th second, 2nd half-time)	YES (Team B)

I'm again in the cube!
I'm again in the cube!

While watching the kids' playground in the early morning hours, the author of this task caught sight of an interesting object: a cube made out of metal bars, composed of many unit-sized cubes made out of metal bars.

While observing the cube, an interesting problem came to his mind. Here follows the two-dimensional version of the problem, since nobody likes problems involving 3D objects:

You're given $N \times N$ matrix (*square* for reference). Some of the fields in the square are blocked and some are empty. The author was watching the square from each of its 4 sides. Firstly, he looked at the square from its left side, and for each of its N rows he wrote how many empty field there were in the row in front of the first blocked field he could see. If there were no blocked fields in a row, he wrote down the number -1. Then he repeated the same procedure looking at the square from its right, top and bottom side, in that order.

By doing so, he wrote $4N$ numbers in total, as he wrote N numbers for each side of the square. However, unknown villains destroyed his square and the only thing left were the numbers he had written down. The author of the task wonders if those numbers make any sense, i.e. if it is possible to form a square for which the same sequence of numbers will be obtained by doing the described procedure.

INPUT

The first line contains a positive integer N ($1 \leq N \leq 100\,000$), dimension of the square.
The second line contains N integers L_i ($-1 \leq L_i < N$), numbers obtained by watching the square from its **left** side, in order from 1st to Nth row.
The second line contains N integers R_i ($-1 \leq R_i < N$), numbers obtained by watching the square from its **right** side, in order from 1st to Nth row.
The second line contains N integers U_i ($-1 \leq U_i < N$), numbers obtained by watching the square from its **top** side, in order from 1st to Nth column.
The second line contains N integers D_i ($-1 \leq D_i < N$), numbers obtained by watching the square from its **bottom** side, in order from 1st to Nth column.

OUTPUT

If it is possible to from a square which satisfies the given conditions, print “DA” (Croatian for yes, without quotation marks), otherwise print “NE” (Croatian for no).

SCORING

In test cases worth 40% of total points, it will hold that $N \leq 1000$.

SAMPLE TESTS

input

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

output

```
DA
```

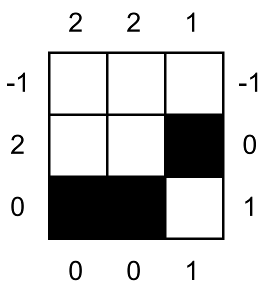
input

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

output

```
NE
```

Clarification of the first test sample:



About thirty years ago, young Krešo participated for the first time in the national informatics competition. Similar to today, the opening of the competition consisted of a series of speakers who tried to demonstrate the importance of this event to the contestants through motivational messages. The audience, with enthusiasm, began applauding every couple of seconds, but Krešo was irritated by one sentence. Namely, one of the speakers claimed he was more appreciative of the logical operation *AND* than the logical operation *OR* because, regardless of the winner's identity, to him both Mirko **and** Slavko were winners of the national competition, instead of the winner being Mirko **or** Slavko. Krešo went mad, got up and started explaining to the audience that this is an operation known as *exclusive OR* (popular *XOR*). After having given his lecture, he gave the distinguished speaker the next task to verify his understanding.

There is a given tree consisting of N nodes, where each node is assigned a value. The value of a path on that tree is defined as the exclusive *OR* of all nodes' values on that path. Your task is to determine the sum of the values of all paths of the tree, including paths containing only one node.

Thirty years later, Krešo has finally persuaded the authors of the COCI tasks to include this task in one of the rounds. Help us return Krešo's faith in the future of competitive programming.

Note: *Exclusive OR* ($\oplus$) is a binary operation that is applied separately on each pair of corresponding bits of its two operands so that some bit in the result is set to 1 if and only if that bit in exactly one operand is set to 1.

INPUT

The first line contains a positive integer N ($1 \le N \le 100\,000$) that denotes the number of nodes in the tree.
The second line contains N integers v_i ($0 \le v_i \le 3\,000\,000$) separated by space, i^{th} value representing the value of the i^{th} node.
The following *(N-1)* lines contain two numbers a_j and b_j ($1 \le a_j$, $b_j \le N$) that indicate that there is an edge between the nodes a_j and b_j.

OUTPUT

Print the required sum of values for all tree paths.

SCORING

In test cases worth 30% of total points, N will be less than or equal to 200.
In test cases worth 50% of total points, N will be less than or equal to 1000.
In test cases worth additional 20% of total points, each node x = 1, 2, ... $N-1$ will be connected to node $x + 1$.

SAMPLE TESTS

input	input	input

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

output

```
10
```

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

output

```
64
```

```
6
5 4 1 3 3 3
3 1
3 5
4 3
4 2
2 6
```

output

```
85
```

Clarification of the first sample test:

- The value of the path from node 1 to node 1 is 1
- The value of the path from node 1 to node 2 is $1 \oplus 2 = 3$
- The value of the path from node 1 to node 3 is $1 \oplus 2 \oplus 3 = 0$
- The value of the path from node 2 to node 2 is 2
- The value of the path from node 2 to node 3 is $2 \oplus 3 = 1$
- The value of the path from node 3 to node 3 is 3

The sum of all paths is 1 + 3 + 0 + 2 + 1 + 3 = 10.

Maja the Bee pollinates flowers in a magical meadow. The meadow can be represented as a matrix of N rows and M columns. In i^{th} row and j^{th} column there are $C_{i,j}$ unpollinated flowers.

Maja will start her journey from her hive, which is located in the field in the A^{th} row and B^{th} column. In several steps, she will visit some fields of the meadow and then return back to her hive. From each field, Maja can move to one of its adjacent cells in one of the following directions: left, right, up or down. Also, Maja will never leave the meadow. Each time Maja flies over some field, she pollinates all unpollinated flowers growing on the field. But the meadow is magical! As soon as Maja leaves the field (i, j), all the pollinated flowers will disappear and $C_{i,j}$ new unpollinated flowers will grow on that field.

Since Maja can't fly forever, she will get tired after K steps and gladly tell her adventurous story to her bee friends. What is the maximal number of flowers Maja can pollinate if she makes exactly K steps and ends her journey back at her hive?

INPUT

The first line contains positive integers N, M ($2 \le N, M \le 100$), A ($1 \le A \le N$), B ($1 \le B \le M$) and K ($2 \le K \le 10^9$). K will always be even.
N lines follows, each containing M integers describing amount of flowers $C_{i,j}$ ($0 \le C_{i,j} \le 10^9$) located in i^{th} row and j^{th} column.
The field containing the hive won't have any flowers on it.

OUTPUT

Print the number from the task statement.

SCORING

In test cases worth 40% of total points, it will hold $K \le 10\,000$.

SAMPLE TESTS

input

```
2 2 1 1 2
0 1
2 10
```

output

```
2
```

input

```
2 2 1 1 4
0 5
5 10
```

output

```
20
```

input

```
3 3 2 2 6
5 1 0
1 0 3
1 3 3
```

output

```
15
```

Clarification of sample tests:

In the first sample Maja starts from the field (1, 1), flyes to the field below, pollinates 2 flowers there, and returns back to the hive.

In the second sample Maja start from the field (1, 1) and can pollinate flowers moving as follows: she moves right, then down, then up and then left. Notice that Maja visited the field (1, 2) twice, each time pollinating 5 flowers on that field.

Little Slavko dreamed of an unusual dream. One sunny morning, N white rectangles climbed one by one on the rectangular roof of Slavko's house. They were preparing for an exotic trip to Hawaii - sunbathing. Each rectangle chose a place on the roof and lay in such a way that its sides were parallel to the edges of the roof. It is possible that some rectangles overlapped parts of other rectangles that have previously lain down on Slavko's roof. For each rectangle its length A_i, height B_i and distances from the left and bottom edges of the roof, X_i and Y_i, respectively, are known.

After sunset, rectangles climbed down the roof and went to sleep dreaming of beautiful Hawaii beaches and their bodies tanned to yellow due to the sun exposure. However, the next morning they spotted a problem! Only parts of rectangles that had been directly exposed to the sun became yellow. In other words, if parts of a rectangle were covered by some other rectangle, then those parts didn't change colour from white to yellow.

Sadly, rectangles that did not change the colour entirely were forced to cancel the trip.

Write a program that will determine for each rectangle if it is going to Hawaii or not.

INPUT

The first line contains a positive integer N ($1 \le N \le 100\ 000$), number of rectangles.
Each of the next N lines contains four integers X_i, Y_i ($0 \le X_i, Y_i \le 10^9$), A_i and B_i ($1 \le A_i, B_i \le 10^9$), describing rectangles in the order they were climbing and lying down on the roof. X_i represents distance from the left edge of the roof, Y_i the distance from the bottom edge of the roof, A_i the length and B_i the height of the i^{th} rectangle.

OUTPUT

You have to print N lines. In i^{th} line print “DA” (Croatian for yes, without quotation marks) if i^{th} rectangle will go to Hawaii, otherwise print “NE” (Croatian for no).

SCORING

In test cases worth 10% of total points, it will hold that $N \le 10\ 000$.

SAMPLE TESTS

input

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

output

```
NE
DA
NE
DA
DA
```

input

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

output

```
NE
NE
DA
```

Clarification of the first sample:
The first and the third rectangle are not entirely exposed to the sun, meaning they won't change colour entirely and won't go to Hawaii. Other rectangles are exposed to the sun entirely.