A. Drinks Choosing

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

Old timers of Summer Informatics School can remember previous camps in which each student was given a drink of his choice on the vechorka (late-evening meal). Or may be the story was more complicated?

There are n students living in a building, and for each of them the favorite drink a_i is known. So you know n integers $a_1, a_2, \dots, a_n$, where a_i ($1 \le a_i \le k$) is the type of the favorite drink of the i-th student. The drink types are numbered from 1 to k.

There are infinite number of drink sets. Each set consists of **exactly two** portions of the same drink. In other words, there are k types of drink sets, the j-th type contains two portions of the drink j. The available number of sets of each of the k types is infinite.

You know that students will receive the minimum possible number of sets to give all students exactly one drink. Obviously, the number of sets will be exactly $\lceil \frac{n}{2} \rceil$, where $\lceil x \rceil$ is x rounded up.

After students receive the sets, they will distribute their portions by their choice: each student will get exactly one portion. Note, that if n is odd then one portion will remain unused and the students' teacher will drink it.

What is the maximum number of students that can get their favorite drink if $\lceil \frac{n}{2} \rceil$ sets will be chosen optimally and students will distribute portions between themselves optimally?

Input

The first line of the input contains two integers n and k ($1 \le n, k \le 1\,000$) — the number of students in the building and the number of different drinks.

The next n lines contain student's favorite drinks. The i-th line contains a single integer from 1 to k — the type of the favorite drink of the i-th student.

Output

Print exactly one integer — the maximum number of students that can get a favorite drink.

Examples

input
```
5 3
1
3
1
1
2
```
output
```
4
```

input
```
10 3
2
1
3
2
3
3
1
3
1
2
```
output
```
9
```

Note

In the first example, students could choose three sets with drinks 1, 1 and 2 (so they will have two sets with two drinks of the type 1 each and one set with two drinks of the type 2, so portions will be $1, 1, 1, 1, 2, 2$). This way all students except the second one will get their favorite drinks.

Another possible answer is sets with drinks 1, 2 and 3. In this case the portions will be $1, 1, 2, 2, 3, 3$. Then all the students except one will gain their favorite drinks. The only student that will not gain the favorite drink will be a student with $a_i = 1$ (i.e. the first, the third or the fourth).

B. Sport Mafia

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

Each evening after the dinner the SIS's students gather together to play the game of Sport Mafia.

For the tournament, Alya puts candies into the box, which will serve as a prize for a winner. To do that, she performs n actions. The first action performed is to put a single candy into the box. For each of the remaining moves she can choose from two options:

- the first option, in case the box contains at least one candy, is to take **exactly one candy out and eat it**. This way the number of candies in the box decreased by 1;
- the second option is to put candies in the box. In this case, Alya will put 1 more candy, than she put in the previous time.

Thus, if the box is empty, then it can only use the second option.

For example, one possible sequence of Alya's actions look as follows:

- put one candy into the box;
- put two candies into the box;
- eat one candy from the box;
- eat one candy from the box;
- put three candies into the box;
- eat one candy from the box;
- put four candies into the box;
- eat one candy from the box;
- put five candies into the box;

This way she will perform 9 actions, the number of candies at the end will be 11, while Alya will eat 4 candies in total.

You know the total number of actions n and the number of candies at the end k. You need to find the total number of sweets Alya ate. That is the number of moves of the first option. It's guaranteed, that for the given n and k the answer always exists.

Please note, that during an action of the first option, Alya takes out and eats exactly one candy.

Input

The first line contains two integers n and k ($1 \le n \le 10^9$; $0 \le k \le 10^9$) — the total number of moves and the number of candies in the box at the end.

It's guaranteed, that for the given n and k the answer exists.

Output

Print a single integer — the number of candies, which Alya ate. Please note, that in this problem there aren't multiple possible answers — the answer is unique for any input data.

Examples

input
```
1 1
```
output
```
0
```

input
```
9 11
```
output
```
4
```

input
```
5 0
```
output
```
3
```

input
```
3 2
```
output
```
1
```

Note

In the first example, Alya has made one move only. According to the statement, the first move is always putting one candy in the box. Hence Alya ate 0 candies.

In the second example the possible sequence of Alya's actions looks as follows:

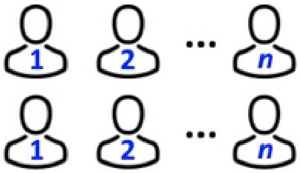
- put 1 candy,
- put 2 candies,
- eat a candy,
- eat a candy,
- put 3 candies,
- eat a candy,
- put 4 candies,
- eat a candy,
- put 5 candies.

This way, she will make exactly $n = 9$ actions and in the end the box will contain $1 + 2 - 1 - 1 + 3 - 1 + 4 - 1 + 5 = 11$ candies. The answer is 4, since she ate 4 candies in total.

C. Basketball Exercise

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

Finally, a basketball court has been opened in SIS, so Demid has decided to hold a basketball exercise session. $2 \cdot n$ students have come to Demid's exercise session, and he lined up them into two rows of the same size (there are exactly n people in each row). Students are numbered from 1 to n in each row in order from left to right.



Now Demid wants to choose a team to play basketball. He will choose players from left to right, and the index of each chosen player (excluding the first one **taken**) will be strictly greater than the index of the previously chosen player. To avoid giving preference to one of the rows, Demid chooses students in such a way that no consecutive chosen students belong to the same row. The first student can be chosen among all $2n$ students (there are no additional constraints), and a team can consist of any number of students.

Demid thinks, that in order to compose a perfect team, he should choose students in such a way, that the total height of all chosen students is maximum possible. Help Demid to find the maximum possible total height of players in a team he can choose.

Input

The first line of the input contains a single integer n ($1 \le n \le 10^5$) — the number of students in each row.

The second line of the input contains n integers $h_{1,1}, h_{1,2}, \ldots, h_{1,n}$ ($1 \le h_{1,i} \le 10^9$), where $h_{1,i}$ is the height of the i-th student in the first row.

The third line of the input contains n integers $h_{2,1}, h_{2,2}, \ldots, h_{2,n}$ ($1 \le h_{2,i} \le 10^9$), where $h_{2,i}$ is the height of the i-th student in the second row.

Output

Print a single integer — the maximum possible total height of players in a team Demid can choose.

Examples

input

```
5
9 3 5 7 3
5 8 1 4 5
```

output

```
29
```

input

```
3
1 2 9
10 1 1
```

output

```
19
```

input

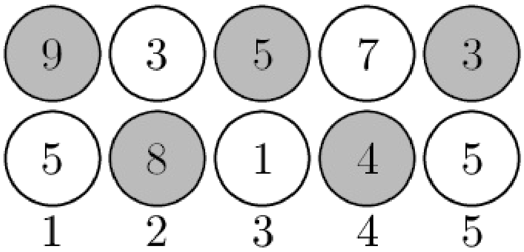
```
1
7
4
```

output

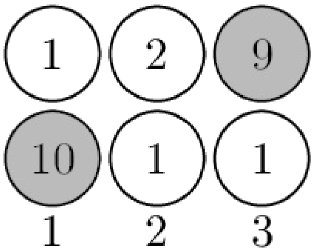
```
7
```

Note

In the first example Demid can choose the following team as follows:



In the second example Demid can choose the following team as follows:



D1. Submarine in the Rybinsk Sea (easy edition)

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

This problem differs from the next one only in the presence of the constraint on the equal length of all numbers $a_1, a_2, \dots, a_n$*. Actually, this problem is a subtask of the problem D2 from the same contest and the solution of D2 solves this subtask too.*

A team of SIS students is going to make a trip on a submarine. Their target is an ancient treasure in a sunken ship lying on the bottom of the Great Rybinsk sea. Unfortunately, the students don't know the coordinates of the ship, so they asked Meshanya (who is a hereditary mage) to help them. He agreed to help them, but only if they solve his problem.

Let's denote a function that alternates digits of two numbers $f(a_1 a_2 \dots a_{p-1} a_p, b_1 b_2 \dots b_{q-1} b_q)$, where $a_1 \dots a_p$ and $b_1 \dots b_q$ are digits of two integers written in the decimal notation without leading zeros.

In other words, the function $f(x, y)$ alternately shuffles the digits of the numbers x and y by writing them from the lowest digits to the older ones, starting with the number y. The result of the function is also built from right to left (that is, from the lower digits to the older ones). If the digits of one of the arguments have ended, then the remaining digits of the other argument are written out. Familiarize with examples and formal definitions of the function below.

For example:

$$f(1111, 2222) = 12121212$$

$$f(7777, 888) = 7787878$$

$$f(33, 44444) = 4443434$$

$$f(555, 6) = 5556$$

$$f(111, 2222) = 2121212$$

Formally,

- if $p \ge q$ then $f(a_1 \dots a_p, b_1 \dots b_q) = a_1 a_2 \dots a_{p-q+1} b_1 a_{p-q+2} b_2 \dots a_{p-1} b_{q-1} a_p b_q$;
- if $p < q$ then $f(a_1 \dots a_p, b_1 \dots b_q) = b_1 b_2 \dots b_{q-p} a_1 b_{q-p+1} a_2 \dots a_{p-1} b_{q-1} a_p b_q$.

Mishanya gives you an array consisting of n integers a_i. **All numbers in this array are of equal length (that is, they consist of the same number of digits).** Your task is to help students to calculate $\sum_{i=1}^{n} \sum_{j=1}^{n} f(a_i, a_j)$ modulo $998\,244\,353$.

Input

The first line of the input contains a single integer n ($1 \le n \le 100\,000$) — the number of elements in the array. The second line of the input contains n integers $a_1, a_2, \dots, a_n$ ($1 \le a_i \le 10^9$) — the elements of the array. All numbers $a_1, a_2, \dots, a_n$ are of equal length (that is, they consist of the same number of digits).

Output

Print the answer modulo $998\,244\,353$.

Examples

input

```
3
12 33 45
```

output

```
26730
```

input

```
2
123 456
```

output

```
1115598
```

input

```
1
1
```

output

```
11
```

input

```
5
1000000000 1000000000 1000000000 1000000000 1000000000
```

output

```
265359409
```

D2. Submarine in the Rybinsk Sea (hard edition)

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

This problem differs from the previous one only in the absence of the constraint on the equal length of all numbers $a_1, a_2, \dots, a_n$.

A team of SIS students is going to make a trip on a submarine. Their target is an ancient treasure in a sunken ship lying on the bottom of the Great Rybinsk sea. Unfortunately, the students don't know the coordinates of the ship, so they asked Meshanya (who is a hereditary mage) to help them. He agreed to help them, but only if they solve his problem.

Let's denote a function that alternates digits of two numbers $f(a_1 a_2 \dots a_{p-1} a_p, b_1 b_2 \dots b_{q-1} b_q)$, where $a_1 \dots a_p$ and $b_1 \dots b_q$ are digits of two integers written in the decimal notation without leading zeros.

In other words, the function $f(x, y)$ alternately shuffles the digits of the numbers x and y by writing them from the lowest digits to the older ones, starting with the number y. The result of the function is also built from right to left (that is, from the lower digits to the older ones). If the digits of one of the arguments have ended, then the remaining digits of the other argument are written out. Familiarize with examples and formal definitions of the function below.

For example:

$$f(1111, 2222) = 12121212$$

$$f(7777, 888) = 7787878$$

$$f(33, 44444) = 4443434$$

$$f(555, 6) = 5556$$

$$f(111, 2222) = 2121212$$

Formally,

- if $p \ge q$ then $f(a_1 \dots a_p, b_1 \dots b_q) = a_1 a_2 \dots a_{p-q+1} b_1 a_{p-q+2} b_2 \dots a_{p-1} b_{q-1} a_p b_q$;
- if $p < q$ then $f(a_1 \dots a_p, b_1 \dots b_q) = b_1 b_2 \dots b_{q-p} a_1 b_{q-p+1} a_2 \dots a_{p-1} b_{q-1} a_p b_q$.

Mishanya gives you an array consisting of n integers a_i, your task is to help students to calculate $\sum_{i=1}^{n} \sum_{j=1}^{n} f(a_i, a_j)$ modulo $998\,244\,353$.

Input

The first line of the input contains a single integer n ($1 \le n \le 100\,000$) — the number of elements in the array. The second line of the input contains n integers $a_1, a_2, \dots, a_n$ ($1 \le a_i \le 10^9$) — the elements of the array.

Output

Print the answer modulo $998\,244\,353$.

Examples

input

```
3
12 3 45
```

output

```
12330
```

input

```
2
123 456
```

output

```
1115598
```

E. OpenStreetMap

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input
output: standard output

Seryozha conducts a course dedicated to building a map of heights of Stepanovo recreation center. He laid a rectangle grid of size $n \times m$ cells on a map (rows of grid are numbered from 1 to n from north to south, and columns are numbered from 1 to m from west to east). After that he measured the average height of each cell above Rybinsk sea level and obtained a matrix of heights of size $n \times m$. The cell (i, j) lies on the intersection of the i-th row and the j-th column and has height $h_{i,j}$.

Seryozha is going to look at the result of his work in the browser. The screen of Seryozha's laptop can fit a subrectangle of size $a \times b$ of matrix of heights ($1 \le a \le n$, $1 \le b \le m$). Seryozha tries to decide how the weather can affect the recreation center — for example, if it rains, where all the rainwater will gather. To do so, he is going to find the cell having minimum height among all cells that are shown on the screen of his laptop.

Help Seryozha to calculate the sum of heights of such cells for all possible subrectangles he can see on his screen. In other words, you have to calculate the sum of minimum heights in submatrices of size $a \times b$ with top left corners in (i, j) over all $1 \le i \le n - a + 1$ and $1 \le j \le m - b + 1$.

Consider the sequence $g_i = (g_{i-1} \cdot x + y) \bmod z$. You are given integers g_0, x, y and z. By miraculous coincidence, $h_{i,j} = g_{(i-1) \cdot m + j - 1}$ ($(i - 1) \cdot m + j - 1$ is the index).

Input

The first line of the input contains four integers n, m, a and b ($1 \le n, m \le 3\,000$, $1 \le a \le n$, $1 \le b \le m$) — the number of rows and columns in the matrix Seryozha has, and the number of rows and columns that can be shown on the screen of the laptop, respectively.

The second line of the input contains four integers g_0, x, y and z ($0 \le g_0, x, y < z \le 10^9$).

Output

Print a single integer — the answer to the problem.

Example

input

```
3 4 2 1
1 2 3 59
```

output

```
111
```

Note

The matrix from the first example:

1	5	13	29
2	7	17	37
18	39	22	47

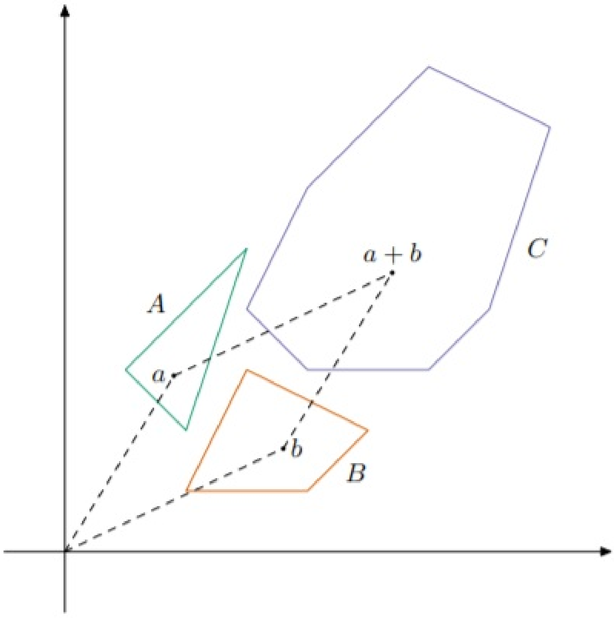
F. Geometers Anonymous Club

time limit per test: 2 seconds
memory limit per test: 256 megabytes
input: standard input

output: standard output

Denis holds a Geometers Anonymous Club meeting in SIS. He has prepared n convex polygons numbered from 1 to n for the club. He plans to offer members of the club to calculate Minkowski sums of these polygons. More precisely, he plans to give q tasks, the i-th of them asks to calculate the sum of Minkowski of polygons with indices from l_i to r_i inclusive.

The sum of Minkowski of two sets A and B is the set $C = \{a + b : a \in A, b \in B\}$. It can be proven that if A and B are convex polygons then C will also be a convex polygon.



Sum of two convex polygons

To calculate the sum of Minkowski of p polygons ($p > 2$), you need to calculate the sum of Minkowski of the first $p - 1$ polygons, and then calculate the sum of Minkowski of the resulting polygon and the p-th polygon.

For the convenience of checking answers, Denis has decided to prepare and calculate the number of vertices in the sum of Minkowski for each task he prepared. Help him to do it.

Input

The first line of the input contains one integer n — the number of convex polygons Denis prepared ($1 \le n \le 100\,000$).

Then n convex polygons follow. The description of the i-th polygon starts with one integer k_i — the number of vertices in the i-th polygon ($3 \le k_i$). The next k_i lines contain two integers x_{ij}, y_{ij} each — coordinates of vertices of the i-th polygon in counterclockwise order ($|x_{ij}|, |y_{ij}| \le 10^9$).

It is guaranteed, that there are no three consecutive vertices lying on the same line. The total number of vertices over all polygons does not exceed $300\,000$.

The following line contains one integer q — the number of tasks ($1 \le q \le 100\,000$). The next q lines contain descriptions of tasks. Description of the i-th task contains two integers l_i and r_i ($1 \le l_i \le r_i \le n$).

Output

For each task print a single integer — the number of vertices in the sum of Minkowski of polygons with indices from l_i to r_i.

Example

input

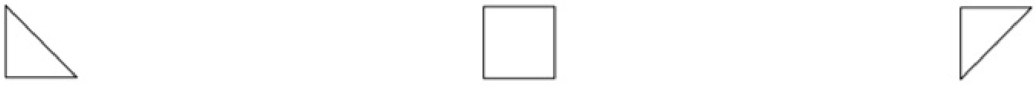
```
3
3
0 0
1 0
0 1
4
1 1
1 2
0 2
0 1
3
2 2
1 2
2 1
3
1 2
2 3
1 3
```

output

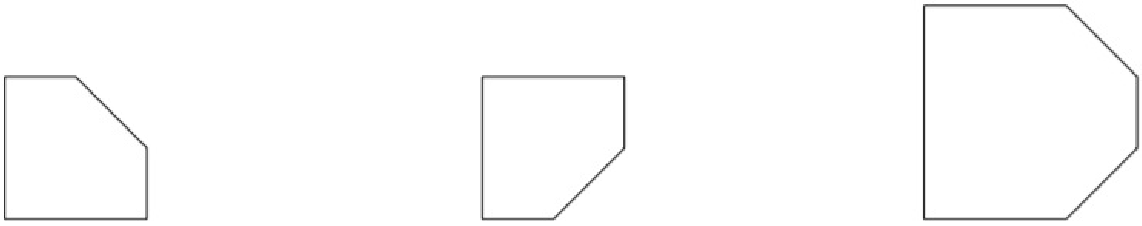
```
5
5
6
```

Note

Description of the example:



First, second and third polygons from the example



Minkowski sums of the first and second, the second and third and all polygons correspondingly