A. Wet Shark and Odd and Even

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

Today, Wet Shark is given n integers. Using any of these integers no more than once, Wet Shark wants to get maximum possible even (divisible by 2) sum. Please, calculate this value for Wet Shark.

Note, that if Wet Shark uses no integers from the n integers, the sum is an even integer 0.

Input

The first line of the input contains one integer, n ($1 \le n \le 100\,000$). The next line contains n space separated integers given to Wet Shark. Each of these integers is in range from 1 to 10^9, inclusive.

Output

Print the maximum possible even sum that can be obtained if we use some of the given integers.

Examples

input

```
3
1 2 3
```

output

```
6
```

input

```
5
999999999 999999999 999999999 999999999 999999999
```

output

```
3999999996
```

Note

In the first sample, we can simply take all three integers for a total sum of 6.

In the second sample Wet Shark should take any four out of five integers $999\,999\,999$.

B. Wet Shark and Bishops

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

Today, Wet Shark is given n bishops on a 1000 by 1000 grid. Both rows and columns of the grid are numbered from 1 to 1000. Rows are numbered from top to bottom, while columns are numbered from left to right.

Wet Shark thinks that two bishops attack each other if they share the same diagonal. Note, that this is the only criteria, so two bishops may attack each other (according to Wet Shark) even if there is another bishop located between them. Now Wet Shark wants to count the number of pairs of bishops that attack each other.

Input

The first line of the input contains n ($1 \le n \le 200\,000$) — the number of bishops.

Each of next n lines contains two space separated integers x_i and y_i ($1 \le x_i, y_i \le 1000$) — the number of row and the number of column where i-th bishop is positioned. It's guaranteed that no two bishops share the same position.

Output

Output one integer — the number of pairs of bishops which attack each other.

Examples

input

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

output

```
6
```

input

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

output

```
0
```

Note

In the first sample following pairs of bishops attack each other: $(1, 3)$, $(1, 5)$, $(2, 3)$, $(2, 4)$, $(3, 4)$ and $(3, 5)$. Pairs $(1, 2)$, $(1, 4)$, $(2, 5)$ and $(4, 5)$ do not attack each other because they do not share the same diagonal.

C. Wet Shark and Flowers

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

There are n sharks who grow flowers for Wet Shark. They are all sitting around the table, such that sharks i and $i + 1$ are neighbours for all i from 1 to $n - 1$. Sharks n and 1 are neighbours too.

Each shark will grow some number of flowers s_i. For i-th shark value s_i is random integer equiprobably chosen in range from l_i to r_i. Wet Shark has it's favourite prime number p, and he really likes it! If for any pair of **neighbouring** sharks i and j the product $s_i \cdot s_j$ is divisible by p, then Wet Shark becomes happy and gives 1000 dollars to each of these sharks.

At the end of the day sharks sum all the money Wet Shark granted to them. Find the expectation of this value.

Input

The first line of the input contains two space-separated integers n and p ($3 \le n \le 100\,000$, $2 \le p \le 10^9$) — the number of sharks and Wet Shark's favourite prime number. It is guaranteed that p is prime.

The i-th of the following n lines contains information about i-th shark — two space-separated integers l_i and r_i ($1 \le l_i \le r_i \le 10^9$), the range of flowers shark i can produce. Remember that s_i is chosen equiprobably among all integers from l_i to r_i, inclusive.

Output

Print a single real number — the expected number of dollars that the sharks receive in total. You answer will be considered correct if its absolute or relative error does not exceed 10^{-6}.

Namely: let's assume that your answer is a, and the answer of the jury is b. The checker program will consider your answer correct, if $\frac{|a-b|}{\max(1, b)} \le 10^{-6}$.

Examples

input

```
3 2
1 2
420 421
420420 420421
```

output

```
4500.0
```

input

```
3 5
1 4
2 3
11 14
```

output

```
0.0
```

Note

A prime number is a positive integer number that is divisible only by 1 and itself. 1 is not considered to be prime.

Consider the first sample. First shark grows some number of flowers from 1 to 2, second sharks grows from 420 to 421 flowers and third from 420420 to 420421. There are eight cases for the quantities of flowers (s_0, s_1, s_2) each shark grows:

1. $(1, 420, 420420)$: note that $s_0 \cdot s_1 = 420$, $s_1 \cdot s_2 = 176576400$, and $s_2 \cdot s_0 = 420420$. For each pair, 1000 dollars will be awarded to each shark. Therefore, each shark will be awarded 2000 dollars, for a total of 6000 dollars.
2. $(1, 420, 420421)$: now, the product $s_2 \cdot s_0$ is not divisible by 2. Therefore, sharks s_0 and s_2 will receive 1000 dollars, while shark s_1 will receive 2000. The total is 4000.
3. $(1, 421, 420420)$: total is 4000
4. $(1, 421, 420421)$: total is 0.
5. $(2, 420, 420420)$: total is 6000.
6. $(2, 420, 420421)$: total is 6000.
7. $(2, 421, 420420)$: total is 6000.
8. $(2, 421, 420421)$: total is 4000.

The expected value is $\frac{6000+4000+4000+0+6000+6000+6000+4000}{8} = 4500$.

In the second sample, no combination of quantities will garner the sharks any money.

D. Rat Kwesh and Cheese

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

Wet Shark asked Rat Kwesh to generate three positive real numbers x, y and z, from 0.1 to 200.0, inclusive. Wet Krash wants to impress Wet Shark, so all generated numbers will have **exactly one** digit after the decimal point.

Wet Shark knows Rat Kwesh will want a lot of cheese. So he will give the Rat an opportunity to earn a lot of cheese. He will hand the three numbers x, y and z to Rat Kwesh, and Rat Kwesh will pick one of the these twelve options:

1. $a_1 = x^{y^z}$;
2. $a_2 = x^{z^y}$;
3. $a_3 = (x^y)^z$;
4. $a_4 = (x^z)^y$;
5. $a_5 = y^{x^z}$;
6. $a_6 = y^{z^x}$;
7. $a_7 = (y^x)^z$;
8. $a_8 = (y^z)^x$;
9. $a_9 = z^{x^y}$;
10. $a_{10} = z^{y^x}$;
11. $a_{11} = (z^x)^y$;
12. $a_{12} = (z^y)^x$.

Let m be the maximum of all the a_i, and c be the smallest index (from 1 to 12) such that $a_c = m$. Rat's goal is to find that c, and he asks you to help him. Rat Kwesh wants to see how much cheese he gets, so he you will have to print the expression corresponding to that a_c.

Input

The only line of the input contains three space-separated real numbers x, y and z ($0.1 \le x, y, z \le 200.0$). Each of x, y and z is given with exactly one digit after the decimal point.

Output

Find the maximum value of expression among x^{y^z}, x^{z^y}, $(x^y)^z$, $(x^z)^y$, y^{x^z}, y^{z^x}, $(y^x)^z$, $(y^z)^x$, z^{x^y}, z^{y^x}, $(z^x)^y$, $(z^y)^x$ and print the corresponding expression. If there are many maximums, print the one that comes first in the list.

x^{y^z} should be outputted as x^y^z (without brackets), and $(x^y)^z$ should be outputted as `(x^y)^z` (quotes for clarity).

Examples

input
```
1.1 3.4 2.5
```
output
```
z^y^x
```

input
```
2.0 2.0 2.0
```
output
```
x^y^z
```

input
```
1.9 1.8 1.7
```
output
```
(x^y)^z
```

E. Wet Shark and Blocks

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

There are b blocks of digits. Each one consisting of the same n digits, which are given to you in the input. Wet Shark must choose **exactly one** digit from each block and concatenate all of those digits together to form one large integer. For example, if he chooses digit 1 from the first block and digit 2 from the second block, he gets the integer 12.

Wet Shark then takes this number modulo x. Please, tell him how many ways he can choose one digit from each block so that he gets exactly k as the final result. As this number may be too large, print it modulo $10^9 + 7$.

Note, that the number of ways to choose some digit in the block is equal to the number of it's occurrences. For example, there are 3 ways to choose digit 5 from block 3 5 6 7 8 9 5 1 1 1 1 5.

Input

The first line of the input contains four space-separated integers, n, b, k and x ($2 \le n \le 50\,000, 1 \le b \le 10^9, 0 \le k < x \le 100, x \ge 2$) — the number of digits in one block, the number of blocks, interesting remainder modulo x and modulo x itself.

The next line contains n space separated integers a_i ($1 \le a_i \le 9$), that give the digits contained in each block.

Output

Print the number of ways to pick exactly one digit from each blocks, such that the resulting integer equals k modulo x.

Examples

input

```
12 1 5 10
3 5 6 7 8 9 5 1 1 1 1 5
```

output

```
3
```

input

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

output

```
0
```

input

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

output

```
6
```

Note

In the second sample possible integers are 22, 26, 62 and 66. None of them gives the remainder 1 modulo 2.

In the third sample integers 11, 13, 21, 23, 31 and 33 have remainder 1 modulo 2. There is exactly one way to obtain each of these integers, so the total answer is 6.