Codeforces Round #257 (Div. 2)

A. Jzzhu and Children

time limit per test: 1 second

memory limit per test: 256 megabytes

input: standard input

output: standard output

There are n children in Jzzhu's school. Jzzhu is going to give some candies to them. Let's number all the children from 1 to n. The i-th child wants to get at least a_i candies.

Jzzhu asks children to line up. Initially, the i-th child stands at the i-th place of the line. Then Jzzhu start distribution of the candies. He follows the algorithm:

1. Give m candies to the first child of the line.
2. If this child still haven't got enough candies, then the child goes to the end of the line, else the child go home.
3. Repeat the first two steps while the line is not empty.

Consider all the children in the order they go home. Jzzhu wants to know, which child will be the last in this order?

Input

The first line contains two integers n, m $(1 \le n \le 100; 1 \le m \le 100)$. The second line contains n integers $a_1, a_2, ..., a_n$ $(1 \le a_i \le 100)$.

Output

Output a single integer, representing the number of the last child.

Sample test(s)

input

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

output

```
4
```

input

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

output

```
6
```

Note

Let's consider the first sample.

Firstly child 1 gets 2 candies and go home. Then child 2 gets 2 candies and go to the end of the line. Currently the line looks like [3, 4, 5, 2] (indices of the children in order of the line). Then child 3 gets 2 candies and go home, and then child 4 gets 2 candies and goes to the end of the line. Currently the line looks like [5, 2, 4]. Then child 5 gets 2 candies and goes home. Then child 2 gets two candies and goes home, and finally child 4 gets 2 candies and goes home.

Child 4 is the last one who goes home.

B. Jzzhu and Sequences

time limit per test: 1 second
memory limit per test: 256 megabytes
input: standard input
output: standard output

Jzzhu has invented a kind of sequences, they meet the following property:

$$f_1 = x;\ \ f_2 = y;\ \ \forall\ i\ (i \geq 2),\ f_i = f_{i-1} + f_{i+1}.$$

You are given x and y, please calculate f_n modulo 1000000007 $(10^9 + 7)$.

Input

The first line contains two integers x and y $(|x|, |y| \leq 10^9)$. The second line contains a single integer n $(1 \leq n \leq 2 \cdot 10^9)$.

Output

Output a single integer representing f_n modulo 1000000007 $(10^9 + 7)$.

Sample test(s)

input

```
2 3
3
```

output

```
1
```

input

```
0 -1
2
```

output

```
1000000006
```

Note

In the first sample, $f_2 = f_1 + f_3$, $3 = 2 + f_3$, $f_3 = 1$.

In the second sample, $f_2 = -1$; -1 modulo $(10^9 + 7)$ equals $(10^9 + 6)$.

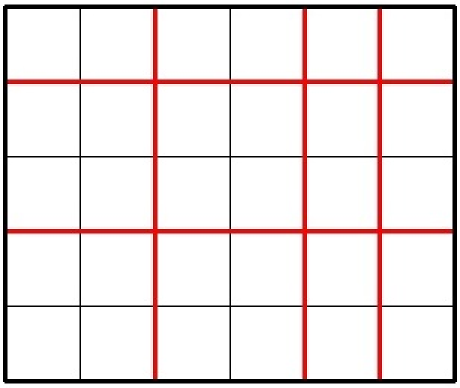
C. Jzzhu and Chocolate

time limit per test: 1 second
memory limit per test: 256 megabytes
input: standard input
output: standard output

Jzzhu has a big rectangular chocolate bar that consists of $n \times m$ unit squares. He wants to cut this bar exactly k times. Each cut must meet the following requirements:

- each cut should be straight (horizontal or vertical);
- each cut should go along edges of unit squares (it is prohibited to divide any unit chocolate square with cut);
- each cut should go inside the whole chocolate bar, and all cuts must be distinct.

The picture below shows a possible way to cut a 5×6 chocolate for 5 times.



Imagine Jzzhu have made k cuts and the big chocolate is splitted into several pieces. Consider the smallest (by area) piece of the chocolate, Jzzhu wants this piece to be as large as possible. What is the maximum possible area of smallest piece he can get with exactly k cuts? The area of a chocolate piece is the number of unit squares in it.

Input

A single line contains three integers n, m, k $(1 \le n, m \le 10^9; 1 \le k \le 2 \cdot 10^9)$.

Output

Output a single integer representing the answer. If it is impossible to cut the big chocolate k times, print `-1`.

Sample test(s)

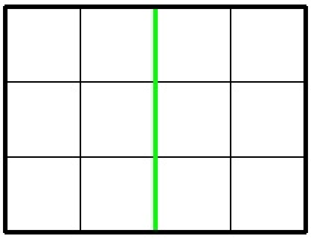
input
```
3 4 1
```
output
```
6
```

input
```
6 4 2
```
output
```
8
```

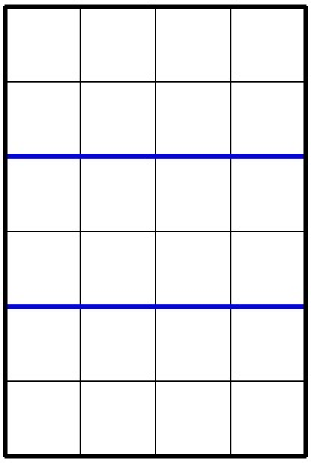
input
```
2 3 4
```
output
```
-1
```

Note

In the first sample, Jzzhu can cut the chocolate following the picture below:



In the second sample the optimal division looks like this:



In the third sample, it's impossible to cut a 2×3 chocolate 4 times.

D. Jzzhu and Cities

time limit per test: 2 seconds

memory limit per test: 256 megabytes

input: standard input

output: standard output

Jzzhu is the president of country A. There are n cities numbered from 1 to n in his country. City 1 is the capital of A. Also there are m roads connecting the cities. One can go from city u_i to v_i (and vise versa) using the i-th road, the length of this road is x_i. Finally, there are k train routes in the country. One can use the i-th train route to go from capital of the country to city s_i (and vise versa), the length of this route is y_i.

Jzzhu doesn't want to waste the money of the country, so he is going to close some of the train routes. Please tell Jzzhu the maximum number of the train routes which can be closed under the following condition: the length of the shortest path from every city to the capital mustn't change.

Input

The first line contains three integers n, m, k $(2 \le n \le 10^5;\ 1 \le m \le 3 \cdot 10^5;\ 1 \le k \le 10^5)$.

Each of the next m lines contains three integers u_i, v_i, x_i $(1 \le u_i, v_i \le n;\ u_i \ne v_i;\ 1 \le x_i \le 10^9)$.

Each of the next k lines contains two integers s_i and y_i $(2 \le s_i \le n;\ 1 \le y_i \le 10^9)$.

It is guaranteed that there is at least one way from every city to the capital. Note, that there can be multiple roads between two cities. Also, there can be multiple routes going to the same city from the capital.

Output

Output a single integer representing the maximum number of the train routes which can be closed.

Sample test(s)

input

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

output

```
2
```

input

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

output

```
2
```

E. Jzzhu and Apples

time limit per test: 1 second
memory limit per test: 256 megabytes
input: standard input
output: standard output

Jzzhu has picked n apples from his big apple tree. All the apples are numbered from 1 to n. Now he wants to sell them to an apple store.

Jzzhu will pack his apples into groups and then sell them. Each group must contain two apples, and the greatest common divisor of numbers of the apples in each group must be greater than 1. Of course, each apple can be part of at most one group.

Jzzhu wonders how to get the maximum possible number of groups. Can you help him?

Input

A single integer n $(1 \le n \le 10^5)$, the number of the apples.

Output

The first line must contain a single integer m, representing the maximum number of groups he can get. Each of the next m lines must contain two integers — the numbers of apples in the current group.

If there are several optimal answers you can print any of them.

Sample test(s)

input

```
6
```

output

```
2
6 3
2 4
```

input

```
9
```

output

```
3
9 3
2 4
6 8
```

input

```
2
```

output

```
0
```