A. Mr. Kitayuta's Gift

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

Mr. Kitayuta has kindly given you a string s consisting of lowercase English letters. You are asked to insert exactly one lowercase English letter into s to make it a palindrome. A *palindrome* is a string that reads the same forward and backward. For example, "`noon`", "`testset`" and "`a`" are all palindromes, while "`test`" and "`kitayuta`" are not.

You can choose any lowercase English letter, and insert it to any position of s, possibly to the beginning or the end of s. You have to insert a letter even if the given string is already a palindrome.

If it is possible to insert one lowercase English letter into s so that the resulting string will be a palindrome, print the string after the insertion. Otherwise, print "`NA`" (without quotes, case-sensitive). In case there is more than one palindrome that can be obtained, you are allowed to print any of them.

Input

The only line of the input contains a string s ($1 \le |s| \le 10$). Each character in s is a lowercase English letter.

Output

If it is possible to turn s into a palindrome by inserting one lowercase English letter, print the resulting string in a single line. Otherwise, print "`NA`" (without quotes, case-sensitive). In case there is more than one solution, any of them will be accepted.

Examples

input
```
revive
```
output
```
reviver
```

input
```
ee
```
output
```
eye
```

input
```
kitayuta
```
output
```
NA
```

Note

For the first sample, insert 'r' to the end of "`revive`" to obtain a palindrome "`reviver`".

For the second sample, there is more than one solution. For example, "`eve`" will also be accepted.

For the third sample, it is not possible to turn "`kitayuta`" into a palindrome by just inserting one letter.

B. Mr. Kitayuta's Colorful Graph

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

Mr. Kitayuta has just bought an undirected graph consisting of n vertices and m edges. The vertices of the graph are numbered from 1 to n. Each edge, namely edge i, has a color c_i, connecting vertex a_i and b_i.

Mr. Kitayuta wants you to process the following q queries.

In the i-th query, he gives you two integers — u_i and v_i.

Find the number of the colors that satisfy the following condition: the edges of that color connect vertex u_i and vertex v_i directly or indirectly.

Input

The first line of the input contains space-separated two integers — n and m ($2 \le n \le 100, 1 \le m \le 100$), denoting the number of the vertices and the number of the edges, respectively.

The next m lines contain space-separated three integers — a_i, b_i ($1 \le a_i < b_i \le n$) and c_i ($1 \le c_i \le m$). Note that there can be multiple edges between two vertices. However, there are no multiple edges of the same color between two vertices, that is, if $i \ne j$, $(a_i, b_i, c_i) \ne (a_j, b_j, c_j)$.

The next line contains a integer — q ($1 \le q \le 100$), denoting the number of the queries.

Then follows q lines, containing space-separated two integers — u_i and v_i ($1 \le u_i, v_i \le n$). It is guaranteed that $u_i \ne v_i$.

Output

For each query, print the answer in a separate line.

Examples

input

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

output

```
2
1
0
```

input

```
5 7
1 5 1
2 5 1
3 5 1
4 5 1
1 2 2
2 3 2
3 4 2
5
1 5
5 1
2 5
1 5
1 4
```

output

```
1
1
1
1
2
```

Note

Let's consider the first sample.

The figure above shows the first sample.

- Vertex 1 and vertex 2 are connected by color 1 and 2.
- Vertex 3 and vertex 4 are connected by color 3.
- Vertex 1 and vertex 4 are not connected by any single color.

C. Mr. Kitayuta, the Treasure Hunter

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

The Shuseki Islands are an archipelago of 30001 small islands in the Yutampo Sea. The islands are evenly spaced along a line, numbered from 0 to 30000 from the west to the east. These islands are known to contain many treasures. There are n gems in the Shuseki Islands in total, and the i-th gem is located on island p_i.

Mr. Kitayuta has just arrived at island 0. With his great jumping ability, he will repeatedly perform jumps between islands to the east according to the following process:

- First, he will jump from island 0 to island d.
- After that, he will continue jumping according to the following rule. Let l be the length of the previous jump, that is, if his previous jump was from island $prev$ to island cur, let $l = cur - prev$. He will perform a jump of length $l - 1$, l or $l + 1$ to the east. That is, he will jump to island $(cur + l - 1)$, $(cur + l)$ or $(cur + l + 1)$ (if they exist). The length of a jump must be positive, that is, he cannot perform a jump of length 0 when $l = 1$. If there is no valid destination, he will stop jumping.

Mr. Kitayuta will collect the gems on the islands visited during the process. Find the maximum number of gems that he can collect.

Input

The first line of the input contains two space-separated integers n and d ($1 \le n, d \le 30000$), denoting the number of the gems in the Shuseki Islands and the length of the Mr. Kitayuta's first jump, respectively.

The next n lines describe the location of the gems. The i-th of them ($1 \le i \le n$) contains a integer p_i ($d \le p_1 \le p_2 \le \ldots \le p_n \le 30000$), denoting the number of the island that contains the i-th gem.

Output

Print the maximum number of gems that Mr. Kitayuta can collect.

Examples

input

```
4 10
10
21
27
27
```

output

```
3
```

input

```
8 8
9
19
28
36
45
55
66
78
```

output

```
6
```

input

```
13 7
8
8
9
16
17
17
18
21
23
24
24
26
30
```

output

```
4
```

Note

In the first sample, the optimal route is 0 → 10 (+1 gem) → 19 → 27 (+2 gems) → ...

In the second sample, the optimal route is 0 → 8 → 15 → 21 → 28 (+1 gem) → 36 (+1 gem) → 45 (+1 gem) → 55 (+1 gem) → 66 (+1 gem) → 78 (+1 gem) → ...

In the third sample, the optimal route is 0 → 7 → 13 → 18 (+1 gem) → 24 (+2 gems) → 30 (+1 gem) → ...

D. Mr. Kitayuta's Technology

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

Shuseki Kingdom is the world's leading nation for innovation and technology. There are n cities in the kingdom, numbered from 1 to n.

Thanks to Mr. Kitayuta's research, it has finally become possible to construct teleportation pipes between two cities. A teleportation pipe will connect two cities unidirectionally, that is, a teleportation pipe from city x to city y cannot be used to travel from city y to city x. The transportation within each city is extremely developed, therefore if a pipe from city x to city y and a pipe from city y to city z are both constructed, people will be able to travel from city x to city z instantly.

Mr. Kitayuta is also involved in national politics. He considers that the transportation between the m pairs of city (a_i, b_i) ($1 \le i \le m$) is important. He is planning to construct teleportation pipes so that for each important pair (a_i, b_i), it will be possible to travel from city a_i to city b_i by using one or more teleportation pipes (but not necessarily from city b_i to city a_i). Find the minimum number of teleportation pipes that need to be constructed. So far, no teleportation pipe has been constructed, and there is no other effective transportation between cities.

Input

The first line contains two space-separated integers n and m ($2 \le n \le 10^5, 1 \le m \le 10^5$), denoting the number of the cities in Shuseki Kingdom and the number of the important pairs, respectively.

The following m lines describe the important pairs. The i-th of them ($1 \le i \le m$) contains two space-separated integers a_i and b_i ($1 \le a_i, b_i \le n, a_i \ne b_i$), denoting that it must be possible to travel from city a_i to city b_i by using one or more teleportation pipes (but not necessarily from city b_i to city a_i). It is guaranteed that all pairs (a_i, b_i) are distinct.

Output

Print the minimum required number of teleportation pipes to fulfill Mr. Kitayuta's purpose.

Examples

input

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

output

```
3
```

input

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

output

```
4
```

Note

For the first sample, one of the optimal ways to construct pipes is shown in the image below:

For the second sample, one of the optimal ways is shown below:

E. Mr. Kitayuta vs. Bamboos

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

Mr. Kitayuta's garden is planted with n bamboos. (Bamboos are tall, fast-growing tropical plants with hollow stems.) At the moment, the height of the i-th bamboo is h_i meters, and it grows a_i meters at the end of each day.

Actually, Mr. Kitayuta hates these bamboos. He once attempted to cut them down, but failed because their stems are too hard. Mr. Kitayuta have not given up, however. He has crafted Magical Hammer with his intelligence to drive them into the ground.

He can use Magical Hammer at most k times during each day, due to his limited Magic Power. Each time he beat a bamboo with Magical Hammer, its height decreases by p meters. If the height would become negative by this change, it will become 0 meters instead (it does not disappear). In other words, if a bamboo whose height is h meters is beaten with Magical Hammer, its new height will be $max(0, h - p)$ meters. It is possible to beat the same bamboo more than once in a day.

Mr. Kitayuta will fight the bamboos for m days, starting today. His purpose is to minimize the height of the tallest bamboo after m days (that is, m iterations of "Mr. Kitayuta beats the bamboos and then they grow"). Find the lowest possible height of the tallest bamboo after m days.

Input

The first line of the input contains four space-separated integers n, m, k and p ($1 \le n \le 10^5, 1 \le m \le 5000, 1 \le k \le 10, 1 \le p \le 10^9$). They represent the number of the bamboos in Mr. Kitayuta's garden, the duration of Mr. Kitayuta's fight in days, the maximum number of times that Mr. Kitayuta beat the bamboos during each day, and the power of Magic Hammer, respectively.

The following n lines describe the properties of the bamboos. The i-th of them ($1 \le i \le n$) contains two space-separated integers h_i and a_i ($0 \le h_i \le 10^9, 1 \le a_i \le 10^9$), denoting the initial height and the growth rate of the i-th bamboo, respectively.

Output

Print the lowest possible height of the tallest bamboo after m days.

Examples

input

```
3 1 2 5
10 10
10 10
15 2
```

output

```
17
```

input

```
2 10 10 1000000000
0 10
0 10
```

output

```
10
```

input

```
5 3 3 10
9 5
9 2
4 7
9 10
3 8
```

output

```
14
```