NTU PK 2019

National Taiwan University

2019/12/21

Problem	Problem Name	Time Limit	Memory Limit
A	Cool Problem	2 s	512 MB
B	Big Array	12 s	512 MB
C	Sequence	5 s	512 MB
D	A PSP Problem	1 s	512 MB
E	Another Big Array	5 s	512 MB
F	Banana	2 s	512 MB
G	Yet Another Array Problem	1 s	512 MB
H	Hanhan's Cafe	1 s	512 MB
I	Escape	3 s	512 MB
J	Palindrome	4 s	512 MB

This page is intentionally left blank.

A. Cool Problem

Problem ID: Cool

One day, Yen-Jen comes up with a cool problem, but he and his friend YP can not solve it.

Given a connected undirected graph with N vertices (vertices are indexed from 1 to N) and M edges. The i^{th} edge connects vertices u_i and v_i, the weight of the i^{th} edge is w_i, and the color of the i^{th} edge is c_i. The weights of edges with same color are distinct. Note that there might be multiple edges between two vertices, but there are no self cycles in the graph.

Now, you need to find a spanning tree in the graph, such that it contain exactly k_0 edges with color 0 and exactly k_1 edges with color 1. You also need to minimize the total weight in the spanning tree.

Input

The first line of the input contains four integers N, M, k_0, k_1 which is mentioned in the description.

In the next M line, each line contains four integers v_i, u_i, w_i, c_i which is mentioned in the description.

- $2 \leq N \leq 50$
- $N - 1 \leq M \leq 210$
- $1 \leq v_i, u_i \leq N, v_i \neq u_i$
- $0 \leq c_i \leq 2$
- $1 \leq w_i \leq 70$
- $0 \leq k_0, k_1 \leq N - 1$
- $\forall i \neq j$, if $c_i = c_j$, then $w_i \neq w_j$ holds.
- The graph is connected

Output

If there are no spanning tree with exactly k_0 edges with color 0 and k_1 edges with color 1, output `-1` in one line.

Otherwise, output the minimum total weight of the spanning tree.

Sample Input 1	Sample Output 1
4 5 1 1 1 2 3 0 2 3 4 1 3 4 5 2 4 3 6 1 1 4 7 0	12

Sample Input 2	Sample Output 2
2 2 0 0 1 2 2 1 1 2 2 0	-1

B. Big Array

Problem ID: Array1

You have an $N \times M$ array. Initially, all the elements in the array are zero.

Then, there are K operations. In the i^{th} operation, two integers s_i, t_i are given, which means that a_{s_i,t_i} is set to 1.

Now, please count the number of rectangles such that all the elements in the rectangle are zero. That is, you need to count the pairs $((x_1, y_1), (x_2, y_2))$ such that $1 \le x_1 \le x_2 \le N, 1 \le y_1 \le y_2 \le M$, and $\sum_{i=x_1}^{x_2} \sum_{j=y_1}^{y_2} a_{i,j} = 0$.

Input

The first line of the input contains three integers N, M, K which is mentioned above.

Then, in the next K lines, the i^{th} line contains two integers s_i, t_i which is mentioned above.

- $1 \le N, M \le 10^5$
- $0 \le K \le 10^5$
- $1 \le s_i \le N$
- $1 \le t_i \le M$
- all (s_i, t_i) are distinct

Output

Output the number mentioned in the description. Since the number might be large, output it modulo 998244353.

Sample Input 1	Sample Output 1
2 2 1 1 1	5

This page is intentionally left blank.

C. Sequence

Problem ID: Sequence

Given initial values $a_1, a_2, \cdots, a_M$ and sequences b_j, c_j. We can define an infinite integer sequence by $a_i = \prod_{j=1}^{M} \left(a_{i-b_j} + c_j\right)$. Please calculate the N-th number in the sequence a_N.

Since the answer can be very large, you only need to output the answer modulo P.

Input

The first line contains 3 positive integers N, M, P.

The second line contains M nonnegative integers $a_1, a_2, \cdots, a_M$.

The third line contains M positive integers $b_1, b_2, \cdots, b_M$.

The fourth line contains M nonnegative integers $c_1, c_2, \cdots, c_M$.

- $1 \leq M < N \leq 10^6$
- $M \leq 150$
- $1 \leq P \leq 3 \times 10^{18}$
- $0 \leq a_i < P$
- $1 \leq b_1 \leq b_2 \leq \cdots \leq b_M \leq M$
- $0 \leq c_i < P$

Output

Print one integer in one line — a_N modulo P.

Note

You can use the 128-bit builtin integer types `__int128` and `unsigned __int128`. However, arithmetic operations (especially division and modulo) on these types are slow, so you may need to reduce the number of operations in order to complete this task.

Besides, there is a little hint: the time complexity of the solution is $O(NM)$.

Sample Input 1	Sample Output 1
8 4 100000 1 2 3 4 2 2 3 3 0 1 0 0	64320

D. A PSP Problem

Problem ID: APSP

There are n cities in a country, and there are m roads that connect them. All the roads have the same length (magically) and each of them connects two cities bidirectionally. The whole country is connected. That is, for each pair of cities, there is always at least one path connecting them. You have n friends (magically), the i-th friend lives in the i-th city. Once in a while, the u-th friend who lives at city u wants to visit another friend at city v and play PSP. So this friend drives from city u to city v using any shortest possible path.

Now, you really want to join your friends, so you plan to surprise your friend u by telling u that you know which exact path u will take. You think by doing so, u will feel happy and then invite you to v's place to play PSP.

Thus, you will like to know for how many number of (u, v) pairs, there is exactly one possible shortest path? Since this problem is about playing PSP on all-pairs shortest path, you decide to call it a PSP problem.

Input

The first line contains two integers n and m, denoting the number of cities and roads, respectively.

In the next m lines, the i-th line contains two integers x_i and y_i, indicating that there is a road connecting city x_i and city y_i.

- $2 \leq n \leq 100$,
- $n - 1 \leq m \leq \frac{n(n-1)}{2}$.
- For all i, $1 \leq x_i, y_i \leq n$ and $x_i \neq y_i$.
- All $\{x_i, y_i\}$ sets within an input file are unique.
- The whole country is connected.

Output

Please output the number of (u, v) pairs where $1 \leq u, v \leq n$ and $u \neq v$, such that the shortest path from u to v is unique.

Sample Input 1	Sample Output 1
5 4 1 2 2 3 3 4 4 5	20

Sample Input 2	Sample Output 2
6 6 1 2 2 3 3 4 4 5 6 5 1 6	24

E. Another Big Array

Problem ID: Array2

Given an array $a_1, a_2, \ldots, a_N$ with length N and another array $b_1, b_2, \ldots, b_M$ with length M, you need to construct an integer array $c_1, c_2, \ldots, c_M$ with length M that meets the following restrictions:

- $0 \le c_i \le N$.
- The occurrence of every non-zero value must be less then or equal to 1.

After you construct the array c, do the following operations:

- For every non-zero c_i, change a_{c_i} to b_i.

Then, calculate the maximum subarray of array a. The maximum subarray sum of the array a is $\max\limits_{1 \le l \le r \le N} (\sum_{i=l}^{r} a_i)$. What is the maximum possible value of the maximum subarray sum of a, after you choose the array c optimally?

Input

The first line of the input contains two integers N, M denotes the length of array a and b.

In the next line, there are N integers $a_1, a_2, \ldots, a_N$.

In the next line, there are N integers $b_1, b_2, \ldots, b_M$.

- $1 \le N \le 10^5$
- $0 \le M \le 10^5$
- $-10^9 \le a_i, b_i \le 10^9$

Output

Output the maximum possible value of the maximum subarray sum of a.

Sample Input 1	Sample Output 1

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

F. Banana

Problem ID: Banana

Yen-Jen has a big banana with length L.

As Yen-Jen's friend, YP has already made N cuts, the i^{th} cut is made at a_i meters from the left of the banana. It is guaranteed that $1 \leq a_1 < a_2 < \cdots < a_N < L$ holds, and all a_i are integers.

Now, Yen-Jen wants to make exactly K more cuts (It is guaranteed that Yen-Jen can make K more cuts). He wants to make cut on $b_1, b_2, \ldots, b_K$ meters from the left of the banana. The following conditions must hold:

- $1 \leq b_1 < b_2 \cdots < b_K < L$
- b_i is an integer
- all the elements in a, b must be distinct

After Yen-Jen selects b, he will calculate the difference between the maximum length of the banana and the minimum length of the banana, let's denote the value as V.

Now, Yen-Jen wants to select b optimally, such that V is as small as possible. You need to tell Yen-Jen the minimum possible value of V.

Input

The first line of the input contains an integer T denotes the number of test cases in this input file.

For each test case, the first line contains three integers N, K, L which is mentioned above.

Then, in the next line, there are N space-separated integers $a_1, a_2, \ldots, a_N$ which is mentioned above

- $1 \le T \le 4 \times 10^5$
- $1 \le N \le 4 \times 10^5$
- $0 \le K \le 10^{15}$
- $1 \le L \le 10^{15}$
- $N + K < L$
- $1 \le a_1 < a_2 < \cdots < a_N < L$
- The summation of N in the T test cases is less then or equal to 4×10^5.

Output

For each test case, output the minimum possible V mentioned in the description.

Sample Input 1	Sample Output 1
2 3 2 12 2 8 10 3 2 12 2 7 10	0 2

G. Yet Another Array Problem

Problem ID: Array3

Given D, mod, find the number of array a with length N satisfying the following conditions:

- $1 \leq N$
- $1 \leq a_1 < a_2 < \cdots < a_N \leq D$
- Define array b with length N as follows: $b_1 = a_1$, $\forall i > 1, b_i = b_{i-1} \oplus a_i$, where $\oplus$ is bitwise exclusive-or function. After construct array b, the condition $b_1 < b_2 < \cdots < b_{N-1} < b_N$ must holds.

Since the number might be too large, you need to output the number modulo mod.

Input

The first line of the input contains an integer T denotes the number of test cases in the input.

For each test case, there's one line contain two integers D, mod, which is mentioned above.

- $1 \leq T \leq 100$
- $1 \leq N, mod \leq 10^9$

Output

For each test case, output the number mentioned in the description.

Sample Input 1	Sample Output 1
2 1 100 2 1	1 0

This page is intentionally left blank.

H. Hanhan's Cafe

Problem ID: Restaurant

Hanhan is thinking of starting a new café. The café will sell M different kinds of drink and N different kinds of cake. Now, Hanhan is considering what should the price of these items be set to. Let x_i be the final price of the i-th drink and y_i be the final price of the i-th cake. Naturally, each x_i and each y_i should be set to a non-negative integer. In other words, $x_1, \ldots, x_M \geq 0$ and $y_1, \ldots, y_N \geq 0$.

To better determine the pricing, he conducts a survey in which he asks Q potential customers questions. For the i-th person, he asks, "What is the maximum price you will pay for a set consists of one a_i-th drink and one b_i-th cake?" and gets the response r_i.

Because Hanhan puts customers first, he want to make sure the pricing will meet everyone's demand. That is, $x_{a_i} + y_{b_i} \leq r_i$ for each i. Yet, He still wants to maximum the profit, so he will try to make $S = \sum_{i=1}^{M} x_i + \sum_{j=1}^{N} y_j$ to be as large as possible.

Help Hanhan to find a possible pricing that maximizes S and satisfies all the constraints.

Input

The first line contains three positive integers M, N, Q. Then there are Q lines, the i-th line consists of three integers a_i, b_i, r_i.

- $1 \leq M, N \leq 300$
- $1 \leq Q \leq 10000$
- $1 \leq a_i \leq M$
- $1 \leq b_i \leq N$
- $0 \leq r_i \leq 10^5$

Output

If the maximum of S does not exist, output a single line consists of only one integer -1.

Or else, output three lines. In the first line, output a single integer $S_{\max}$, which represents the maximum possible value of S. Then, output any solution that achieves this maximum. In the second line, output M non-negative integers $x_1, \ldots, x_M$ separated by spaces. In the third line, output N non-negative integers $y_1, \ldots, y_N$ separated by spaces. The solution should satisfy $x_1, \ldots, x_M, y_1, \ldots, y_N \leq 10^6$. It is guaranteed that an optimal solution that satisfies this additional constraint exists.

Sample Input 1	Sample Output 1
2 2 4 1 1 3 1 2 2 2 1 5 2 2 6	7 0 3 2 2

Sample Input 2	Sample Output 2
4 3 9 1 1 3 4 2 9 2 2 8 2 1 3 3 2 4 3 3 5 1 2 2 4 3 6 2 2 10	15 2 2 4 5 1 0 1

Sample Input 3	Sample Output 3
2 3 3 1 1 10 2 2 20 2 1 30	-1

I. Escape

Problem ID: Escape

Meow got trapped in a D-dimensional space!

The space which Meow got trapped in is a D-dimensional Euclidean space. Moreover, there are N surveillance cameras in the space to prevent Meow from escaping.

Meow is trying to escape from this weird high-dimensional space. To escape, Meow needs to destroy some of the surveillance cameras. However, Meow must first destroy a camera on a vertex of the convex hull of the N cameras' position. Otherwise, the action will be captured by the cameras, and Meow will fail to escape. (The vertices of a convex hull of a point set A is the set of all points x such that $x \in A$ and the convex hull of A and the convex hull of $A \setminus \{x\}$ are different.)

Meow also discovered that the positions of surveillance cameras can be calculated by the following C program:

```
#include <stdint.h>

const uint64_t kTable[3] = {0x9e3779b185ebca87, 0xc2b2ae3d27d4eb4f,
   0x165667b19e3779f9};
uint64_t Mix(uint64_t a, uint64_t b) {
  a += b * kTable[1];
  a = (a << 31) | (a >> 33);
  return a * kTable[0];
}

uint64_t PRNG(uint64_t a, uint64_t b) {
  uint64_t v1 = Mix(-kTable[0], a);
  uint64_t v2 = Mix(kTable[1], b);
  uint64_t ret = ((v1 << 18) | (v1 >> 46)) + ((v2 << 7) | (v2 >> 57)
     );
  ret ^= ret >> 33;
  ret *= kTable[1];
  ret ^= ret >> 29;
  ret *= kTable[2];
  ret ^= ret >> 32;
  return ret;
}

```

```
int pts[20000][500];
void Generate(int N, int D, uint64_t seed) {
  seed += N * 500 + D;
  for (int i = 0; i < N; i++, seed = PRNG(seed, 880301)) {
    for (int j = 0; j < D; j++) pts[i][j] = PRNG(seed, j ^ seed) &
        65535;
  }
}
```

After calling `Generate(N, D, seed)`, the positions of surveillance cameras are placed in the `pts` array. The i-th camera is placed at $(\text{pts}[i][0], \text{pts}[i][1], \cdots, \text{pts}[i][D-1])$ $(0 \le i < N)$.

Since Meow is not familiar with high-dimensional spaces, please write a program to help Meow to calculate the number of vertices of the convex hull formed by the positions of the surveillance cameras.

Input

The first line contains 3 integers N, D, seed — the number of surveillance cameras, the dimensions of the space, and the input seed of the program. To get the positions of the surveillance cameras, simply pass the three numbers into `Generate` function.

- $3 \le N \le \max(480, 40 \times D)$
- $1 \le D \le 500$
- $0 \le \text{seed} < 1000$

Output

Print one integer — the number of vertices of the convex hull formed by the positions of the surveillance cameras.

Sample Input 1	Sample Output 1
`10 2 213`	`6`

J. Palindrome

Problem ID: Palindrome

Given a string S with length N. You need to perform Q operations on it. Each operation is of one of the following three kinds:

1. Given a character c, add c to the front of S.
2. Given a character c, add c to the back of S.
3. Given an integer k, count the number of substring T in S, such that T is a palindrome, and the length of T is less then or equal to k.

A string $a = a_0a_1 \ldots a_{m-1}$ with length m is a substring of string $b = b_0b_1 \ldots b_{n-1}$ with length n, if there is an index L, such that $a_i = b_{i+L}, \forall i \in [0, m-1]$.

A string $s = s_0s_1 \ldots s_{n-1}$ is a palindrome if $s_i = s_{n-i-1}$ holds for $i \in [0, n-1]$.

Input

The first line of the input contains two integers N, Q which is mentioned in the description.

The second line contains a string S consists of lowercase letters with length N.

In the next Q lines, each line represents an operation.

The first integer *type* in each line denotes the type of the operation.

If $type = 1$, then one lowercase letter c follows. You need to add c to the front of S.

If $type = 2$, thne one lowercase letter c follows. You need to add c to the back of S.

If $type = 3$, then one integer k follows. You need to count the number of substring T in S, such that T is a palindrome, and the length of T is less then or equal to k.

- $1 \leq N \leq 10^5$
- $1 \leq Q \leq 4 \times 10^5$
- The length of S is N, and S only contains lowercase letter.
- $1 \leq type \leq 3$
- c is a lowercase letter
- $1 \leq k \leq 5 \times 10^5$

Output

For each $type = 3$ operation, output the number mentioned in the description.

Sample Input 1	Sample Output 1
3 6 aab 3 1 3 2 1 a 3 4 2 z 3 500000	3 4 7 8

Sample Input 2	Sample Output 2
2 2 0 0 1 2 2 1 1 2 2 0	-1