Codeforces Round #317 [AimFund Thanks-Round] (Div. 1)

A. Lengthening Sticks

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

You are given three sticks with positive integer lengths of a, b, and c centimeters. You can increase length of some of them by some positive integer number of centimeters (different sticks can be increased by a different length), but in total by at most l centimeters. In particular, it is allowed not to increase the length of any stick.

Determine the number of ways to increase the lengths of some sticks so that you can form from them a non-degenerate (that is, having a positive area) triangle. Two ways are considered different, if the length of some stick is increased by different number of centimeters in them.

Input

The single line contains 4 integers a, b, c, l ($1 \le a, b, c \le 3 \cdot 10^5$, $0 \le l \le 3 \cdot 10^5$).

Output

Print a single integer — the number of ways to increase the sizes of the sticks by the total of at most l centimeters, so that you can make a non-degenerate triangle from it.

Sample test(s)

input
1 1 1 2
output
4

input
1 2 3 1
output
2

input
10 2 1 7
output
0

Note

In the first sample test you can either not increase any stick or increase any two sticks by 1 centimeter.

In the second sample test you can increase either the first or the second stick by one centimeter. Note that the triangle made from the initial sticks is degenerate and thus, doesn't meet the conditions.

B. Minimization

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

You've got array A, consisting of n integers and a positive integer k. Array A is indexed by integers from 1 to n.

You need to permute the array elements so that value

$$\sum_{i=1}^{n-k} |A[i] - A[i+k]|$$

became minimal possible. In particular, it is allowed not to change order of elements at all.

Input

The first line contains two integers n, k ($2 \le n \le 3 \cdot 10^5$, $1 \le k \le min(5000, n - 1)$).

The second line contains n integers $A[1], A[2], ..., A[n]$ ($-10^9 \le A[i] \le 10^9$), separate by spaces — elements of the array A.

Output

Print the minimum possible value of the sum described in the statement.

Sample test(s)

input
3 2 1 2 4
output
1

input
5 2 3 -5 3 -5 3
output
0

input
6 3 4 3 4 3 2 5
output
3

Note

In the first test one of the optimal permutations is $1\ 4\ 2$.

In the second test the initial order is optimal.

In the third test one of the optimal permutations is $2\ 3\ 4\ 4\ 3\ 5$.

C. CNF 2

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

'In Boolean logic, a formula is in conjunctive normal form (CNF) or clausal normal form if it is a conjunction of clauses, where a clause is a disjunction of literals' (cited from https://en.wikipedia.org/wiki/Conjunctive_normal_form)

In the other words, CNF is a formula of type $(v_{11} \mid v_{12} \mid \ldots \mid v_{1k_1}) \& (v_{21} \mid v_{22} \mid \ldots \mid v_{2k_2}) \& \ldots \& (v_{l1} \mid v_{l2} \mid \ldots \mid v_{lk_l})$, where $\&$ represents a logical "AND" (conjunction), $\mid$ represents a logical "OR" (disjunction), and v_{ij} are some boolean variables or their negations. Each statement in brackets is called a *clause*, and v_{ij} are called *literals*.

You are given a CNF containing variables $x_1, \ldots, x_m$ and their negations. We know that each variable occurs in at most two clauses (with negation and without negation in total). Your task is to determine whether this CNF is *satisfiable*, that is, whether there are such values of variables where the CNF value is true. If CNF is satisfiable, then you also need to determine the values of the variables at which the CNF is true.

It is guaranteed that each variable occurs at most once in each clause.

Input

The first line contains integers n and m $(1 \le n, m \le 2 \cdot 10^5)$ — the number of clauses and the number variables, correspondingly.

Next n lines contain the descriptions of each clause. The i-th line first contains first number k_i $(\boldsymbol{k_i \ge 1})$ — the number of literals in the i-th clauses. Then follow space-separated literals v_{ij} $(1 \le |v_{ij}| \le m)$. A literal that corresponds to v_{ij} is $x_{|v_{ij}|}$ either with negation, if v_{ij} is negative, or without negation otherwise.

Output

If CNF is not satisfiable, print a single line "NO" (without the quotes), otherwise print two strings: string "YES" (without the quotes), and then a string of m numbers zero or one — the values of variables in satisfying assignment in the order from x_1 to x_m.

Sample test(s)

input
```
2 2
2 1 -2
2 2 -1
```
output
```
YES
11
```

input
```
4 3
1 1
1 2
3 -1 -2 3
1 -3
```
output
```
NO
```

input
```
5 6
2 1 2
3 1 -2 3
4 -3 5 4 6
2 -6 -4
1 5
```
output
```
YES
100010
```

Note

In the first sample test formula is $(x_1 \mid \neg x_2) \& (x_2 \mid \neg x_1)$. One of possible answer is $x_1 = TRUE, x_2 = TRUE$.

D. Campus

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

Oscolcovo city has a campus consisting of n student dormitories, n universities and n military offices. Initially, the i-th dormitory belongs to the i-th university and is assigned to the i-th military office.

Life goes on and the campus is continuously going through some changes. The changes can be of four types:

1. University a_j merges with university b_j. After that all the dormitories that belonged to university b_j are assigned to to university a_j, and university b_j disappears.
2. Military office c_j merges with military office d_j. After that all the dormitories that were assigned to military office d_j, are assigned to military office c_j, and military office d_j disappears.
3. Students of university x_j move in dormitories. Lets k_{x_j} is the number of dormitories that belong to this university at the time when the students move in. Then the number of students in each dormitory of university x_j increases by k_{x_j} (note that the more dormitories belong to the university, the more students move in each dormitory of the university).
4. Military office number y_j conducts raids on all the dormitories assigned to it and takes all students from there.

Thus, at each moment of time each dormitory is assigned to exactly one university and one military office. Initially, all the dormitory are empty.

Your task is to process the changes that take place in the campus and answer the queries, how many people currently live in dormitory q_j.

Input

The first line contains two integers, n and m $(1 \le n, m \le 5 \cdot 10^5)$ — the number of dormitories and the number of queries, respectively.

Next m lines contain the queries, each of them is given in one of the following formats:

- «U a_j b_j» — merging universities;
- «M c_j d_j» — merging military offices;
- «A x_j» — students of university x_j moving in the dormitories;
- «Z y_j» — a raid in military office y_j;
- «Q q_j» — a query asking the number of people in dormitory q_j.

All the numbers in the queries are positive integers and do not exceed n. It is guaranteed that at the moment of the query the universities and military offices, that are present in the query, exist.

Output

In the i-th line print the answer to the i-th query asking the number of people in the dormitory.

Sample test(s)

input

```
2 7
A 1
Q 1
U 1 2
A 1
Z 1
Q 1
Q 2
```

output

```
1
0
2
```

input

```
5 12
U 1 2
M 4 5
A 1
Q 1
A 3
A 4
Q 3
Q 4
Z 4
Q 4
A 5
Q 5
```

output

```
2
1
1
0
```

```
1
```

Note

Consider the first sample test:

- In the first query university 1 owns only dormitory 1, so after the query dormitory 1 will have 1 student.
- After the third query university 1 owns dormitories 1 and 2.
- The fourth query increases by 2 the number of students living in dormitories 1 and 2 that belong to university number 1. After that 3 students live in the first dormitory and 2 students live in the second dormitory.
- At the fifth query the number of students living in dormitory 1, assigned to the military office 1, becomes zero.

E. Geometric Progressions

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

Geometric progression with the first element a and common ratio b is a sequence of numbers $a, ab, ab^2, ab^3, ...$

You are given n integer geometric progressions. Your task is to find the smallest integer x, that is the element of all the given progressions, or else state that such integer does not exist.

Input

The first line contains integer $(1 \le n \le 100)$ — the number of geometric progressions.

Next n lines contain pairs of integers a, b $(1 \le a, b \le 10^9)$, that are the first element and the common ratio of the corresponding geometric progression.

Output

If the intersection of all progressions is empty, then print -1, otherwise print the remainder of the minimal positive integer number belonging to all progressions modulo 1000000007 $(10^9 + 7)$.

Sample test(s)

input
2 2 2 4 1

output
4

input
2 2 2 3 3

output
-1

Note

In the second sample test one of the progressions contains only powers of two, the other one contains only powers of three.