Problem A. Cumulative Sum Query

Time limit	7000 ms
Mem limit	1572864 kB
Code length Limit	50000 B
OS	Linux

William Macfarlane wants to look at an array.

You are given a list of **N** numbers and **Q** queries. Each query is specified by two numbers *i* and *j*; the answer to each query is the sum of every number between the range [i, j] (inclusive).

Note: the query ranges are specified using 0-based indexing.

Input

The first line contains **N**, the number of integers in our list (**N <= 100,000**). The next line holds **N** numbers that are guaranteed to fit inside an integer. Following the list is a number **Q** (**Q <= 10,000**). The next **Q** lines each contain two numbers **i** and **j** which specify a query you must answer (**0 <= i, j <= N-1**).

Output

For each query, output the answer to that query on its own line in the order the queries were made.

Example

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

Problem B. Kuriyama Mirai's Stones

Time limit	2000 ms
Mem limit	262144 kB

Kuriyama Mirai has killed many monsters and got many (namely n) stones. She numbers the stones from 1 to n. The cost of the i-th stone is v_i. Kuriyama Mirai wants to know something about these stones so she will ask you two kinds of questions:

1. She will tell you two numbers, l and r $(1 \le l \le r \le n)$, and you should tell her $\sum_{i=l}^{r} v_i$.
2. Let u_i be the cost of the i-th cheapest stone (the cost that will be on the i-th place if we arrange all the stone costs in non-decreasing order). This time she will tell you two numbers, l and r $(1 \le l \le r \le n)$, and you should tell her $\sum_{i=l}^{r} u_i$.

For every question you should give the correct answer, or Kuriyama Mirai will say "fuyukai desu" and then become unhappy.

Input

The first line contains an integer n $(1 \le n \le 10^5)$. The second line contains n integers: $v_1, v_2, ..., v_n$ $(1 \le v_i \le 10^9)$ — costs of the stones.

The third line contains an integer m $(1 \le m \le 10^5)$ — the number of Kuriyama Mirai's questions. Then follow m lines, each line contains three integers $type$, l and r $(1 \le l \le r \le n;\ 1 \le type \le 2)$, describing a question. If $type$ equal to 1, then you should output the answer for the first question, else you should output the answer for the second one.

Output

Print m lines. Each line must contain an integer — the answer to Kuriyama Mirai's question. Print the answers to the questions in the order of input.

Sample 1

Input	Output
6 6 4 2 7 2 7 3 2 3 6 1 3 4 1 1 6	24 9 28

Sample 2

Input	Output
4 5 5 2 3 10 1 2 4 2 1 4 1 1 1 2 1 4 2 1 2 1 1 1 1 3 3 1 1 3 1 4 4 1 2 2	10 15 5 15 5 5 2 12 3 5

Note

Please note that the answers to the questions may overflow 32-bit integer type.

Problem C. Smallest Difference

Time limit	1000 ms
Mem limit	262144 kB
OS	Windows

You are given an array a consists of n elements, find the maximum number of elements you can select from the array such that the absolute difference between any two of the chosen elements is ≤ 1.

Input

The first line contains an integer T $(1 \leq T \leq 100)$, in which T is the number of test cases.

The first line of each test case consist of an integer n $(2 \leq n \leq 10^4)$, in which n is size of the array a

The a line follow containing n elements $a_1, a_2, \ldots, a_n$ $(1 \leq a_i \leq 10^4)$, giving the array a.

Output

For each test case, print a single line containing the maximum number of elements you can select from the array such that the absolute difference between any two of the chosen elements is ≤ 1.

Sample 1

Input	Output
2 3 1 2 3 5 2 2 3 4 5	2 3

Problem D. Spy Detected!

Time limit 2000 ms
Mem limit 262144 kB

You are given an array a consisting of n $(n \geq 3)$ positive integers. It is known that in this array, all the numbers except one are the same (for example, in the array $[4, 11, 4, 4]$ all numbers except one are equal to 4).

Print the index of the element that does not equal others. The numbers in the array are numbered from one.

Input

The first line contains a single integer t $(1 \leq t \leq 100)$. Then t test cases follow.

The first line of each test case contains a single integer n $(3 \leq n \leq 100)$ — the length of the array a.

The second line of each test case contains n integers $a_1, a_2, \ldots, a_n$ $(1 \leq a_i \leq 100)$.

It is guaranteed that all the numbers except one in the a array are the same.

Output

For each test case, output a single integer — the index of the element that is not equal to others.

Sample 1

Input	Output
4 4 11 13 11 11 5 1 4 4 4 4 10 3 3 3 3 10 3 3 3 3 3 3 20 20 10	2 1 5 3

Problem E. Odd Queries

Time limit 2000 ms
Mem limit 262144 kB

You have an array $a_1, a_2, \ldots, a_n$. Answer q queries of the following form:

- If we change all elements in the range $a_l, a_{l+1}, \ldots, a_r$ of the array to k, will the sum of the entire array be odd?

Note that queries are **independent** and do not affect future queries.

Input

Each test contains multiple test cases. The first line contains the number of test cases t ($1 \le t \le 10^4$). The description of the test cases follows.

The first line of each test case consists of 2 integers n and q ($1 \le n \le 2 \cdot 10^5$; $1 \le q \le 2 \cdot 10^5$) — the length of the array and the number of queries.

The second line of each test case consists of n integers a_i ($1 \le a_i \le 10^9$) — the array a.

The next q lines of each test case consists of 3 integers l, r, k ($1 \le l \le r \le n; 1 \le k \le 10^9$) — the queries.

It is guaranteed that the sum of n over all test cases doesn't exceed $2 \cdot 10^5$, and the sum of q doesn't exceed $2 \cdot 10^5$.

Output

For each query, output "`YES`" if the sum of the entire array becomes odd, and "`NO`" otherwise.

You can output the answer in any case (upper or lower). For example, the strings "`yEs`", "`yes`", "`Yes`", and "`YES`" will be recognized as positive responses.

Sample 1

Input	Output
2 5 5 2 2 1 3 2 2 3 3 2 3 4 1 5 5 1 4 9 2 4 3 10 5 1 1 1 1 1 1 1 1 1 1 3 8 13 2 5 10 3 8 10 1 10 2 1 9 100	YES YES YES NO YES NO NO NO NO YES

Note

For the first test case:

- If the elements in the range $(2, 3)$ would get set to 3 the array would become $\{2, 3, 3, 3, 2\}$, the sum would be $2 + 3 + 3 + 3 + 2 = 13$ which is odd, so the answer is "`YES`".
- If the elements in the range $(2, 3)$ would get set to 4 the array would become $\{2, 4, 4, 3, 2\}$, the sum would be $2 + 4 + 4 + 3 + 2 = 15$ which is odd, so the answer is "`YES`".
- If the elements in the range $(1, 5)$ would get set to 5 the array would become $\{5, 5, 5, 5, 5\}$, the sum would be $5 + 5 + 5 + 5 + 5 = 25$ which is odd, so the answer is "`YES`".
- If the elements in the range $(1, 4)$ would get set to 9 the array would become $\{9, 9, 9, 9, 2\}$, the sum would be $9 + 9 + 9 + 9 + 2 = 38$ which is even, so the answer is "`NO`".
- If the elements in the range $(2, 4)$ would get set to 3 the array would become $\{2, 3, 3, 3, 2\}$, the sum would be $2 + 3 + 3 + 3 + 2 = 13$ which is odd, so the answer is "`YES`".

Problem F. Ilya and Queries

Time limit 2000 ms
Mem limit 262144 kB

Ilya the Lion wants to help all his friends with passing exams. They need to solve the following problem to pass the IT exam.

You've got string $s = s_1 s_2 \dots s_n$ (n is the length of the string), consisting only of characters "." and "#" and m queries. Each query is described by a pair of integers l_i, r_i ($1 \le l_i < r_i \le n$). The answer to the query l_i, r_i is the number of such integers i ($l_i \le i < r_i$), that $s_i = s_{i+1}$.

Ilya the Lion wants to help his friends but is there anyone to help him? Help Ilya, solve the problem.

Input

The first line contains string s of length n ($2 \le n \le 10^5$). It is guaranteed that the given string only consists of characters "." and "#".

The next line contains integer m ($1 \le m \le 10^5$) — the number of queries. Each of the next m lines contains the description of the corresponding query. The i-th line contains integers l_i, r_i ($1 \le l_i < r_i \le n$).

Output

Print m integers — the answers to the queries in the order in which they are given in the input.

Sample 1

Input	Output
...... 4 3 4 2 3 1 6 2 6	1 1 5 4

Sample 2

Input	Output
#..### 5 1 3 5 6 1 5 3 6 3 4	1 1 2 2 0

Problem G. Nineteen

Time limit 1000 ms
Mem limit 262144 kB

Alice likes word "`nineteen`" very much. She has a string s and wants the string to contain as many such words as possible. For that reason she can rearrange the letters of the string.

For example, if she has string "`xiineteenppnnnewtnee`", she can get string "`x`**`nineteen`**`pp`**`nineteen`**`w`", containing (the occurrences marked) two such words. More formally, word "`nineteen`" occurs in the string the number of times you can read it starting from some letter of the string. Of course, you shouldn't skip letters.

Help her to find the maximum number of "`nineteen`"s that she can get in her string.

Input

The first line contains a non-empty string s, consisting only of lowercase English letters. The length of string s doesn't exceed 100.

Output

Print a single integer — the maximum number of "`nineteen`"s that she can get in her string.

Sample 1

Input	Output
nniinneetteeeenn	2

Sample 2

Input	Output
nneteenabcnneteenabcnneteenabcnneteenabcnnet	2

Sample 3

Input	Output
nineteenineteen	2

Problem H. Stripe

Time limit 2000 ms
Mem limit 65536 kB

Once Bob took a paper stripe of n squares (the height of the stripe is 1 square). In each square he wrote an integer number, possibly negative. He became interested in how many ways exist to cut this stripe into two pieces so that the sum of numbers from one piece is equal to the sum of numbers from the other piece, and each piece contains positive integer amount of squares. Would you help Bob solve this problem?

Input

The first input line contains integer n ($1 \leq n \leq 10^5$) — amount of squares in the stripe. The second line contains n space-separated numbers — they are the numbers written in the squares of the stripe. These numbers are integer and do not exceed 10000 in absolute value.

Output

Output the amount of ways to cut the stripe into two non-empty pieces so that the sum of numbers from one piece is equal to the sum of numbers from the other piece. Don't forget that it's allowed to cut the stripe along the squares' borders only.

Sample 1

Input	Output
9 1 5 -6 7 9 -16 0 -2 2	3

Sample 2

Input	Output
3 1 1 1	0

Sample 3

Input	Output
2 0 0	1

Problem I. Ohana Cleans Up

Time limit 2000 ms
Mem limit 262144 kB

Ohana Matsumae is trying to clean a room, which is divided up into an n by n grid of squares. Each square is initially either clean or dirty. Ohana can sweep her broom over columns of the grid. Her broom is very strange: if she sweeps over a clean square, it will become dirty, and if she sweeps over a dirty square, it will become clean. She wants to sweep some columns of the room to maximize the number of rows that are completely clean. It is not allowed to sweep over the part of the column, Ohana can only sweep the whole column.

Return the maximum number of rows that she can make completely clean.

Input

The first line of input will be a single integer n ($1 \le n \le 100$).

The next n lines will describe the state of the room. The i-th line will contain a binary string with n characters denoting the state of the i-th row of the room. The j-th character on this line is '1' if the j-th square in the i-th row is clean, and '0' if it is dirty.

Output

The output should be a single line containing an integer equal to a maximum possible number of rows that are completely clean.

Sample 1

Input	Output
4 0101 1000 1111 0101	2

Sample 2

Input	Output
3 111 111 111	3

Note

In the first sample, Ohana can sweep the 1st and 3rd columns. This will make the 1st and 4th row be completely clean.

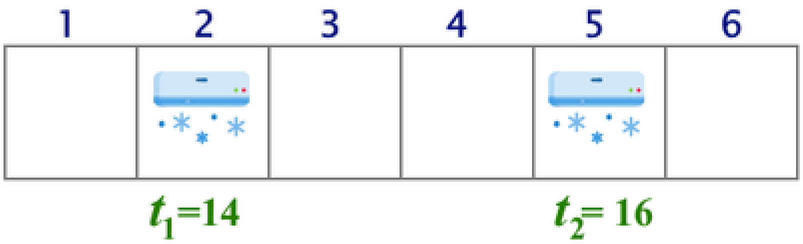
In the second sample, everything is already clean, so Ohana doesn't need to do anything.

Problem J. Air Conditioners

Time limit 2000 ms
Mem limit 524288 kB

On a strip of land of length n there are k air conditioners: the i-th air conditioner is placed in cell a_i ($1 \le a_i \le n$). Two or more air conditioners cannot be placed in the same cell (i.e. all a_i are distinct).

Each air conditioner is characterized by one parameter: temperature. The i-th air conditioner is set to the temperature t_i.



Example of strip of length $n = 6$, where $k = 2$, $a = [2, 5]$ and $t = [14, 16]$.

For each cell i ($1 \le i \le n$) find it's temperature, that can be calculated by the formula

$$\min_{1 \le j \le k} (t_j + |a_j - i|),$$

where $|a_j - i|$ denotes absolute value of the difference $a_j - i$.

In other words, the temperature in cell i is equal to the minimum among the temperatures of air conditioners, increased by the distance from it to the cell i.

Let's look at an example. Consider that $n = 6, k = 2$, the first air conditioner is placed in cell $a_1 = 2$ and is set to the temperature $t_1 = 14$ and the second air conditioner is placed in cell $a_2 = 5$ and is set to the temperature $t_2 = 16$. In that case temperatures in cells are:

1. temperature in cell 1 is:
   $\min(14 + |2 - 1|, 16 + |5 - 1|) = \min(14 + 1, 16 + 4) = \min(15, 20) = 15;$
2. temperature in cell 2 is:
   $\min(14 + |2 - 2|, 16 + |5 - 2|) = \min(14 + 0, 16 + 3) = \min(14, 19) = 14;$
3. temperature in cell 3 is:
   $\min(14 + |2 - 3|, 16 + |5 - 3|) = \min(14 + 1, 16 + 2) = \min(15, 18) = 15;$
4. temperature in cell 4 is:
   $\min(14 + |2 - 4|, 16 + |5 - 4|) = \min(14 + 2, 16 + 1) = \min(16, 17) = 16;$
5. temperature in cell 5 is:
   $\min(14 + |2 - 5|, 16 + |5 - 5|) = \min(14 + 3, 16 + 0) = \min(17, 16) = 16;$

6. temperature in cell 6 is: $\min(14 + |2 - 6|, 16 + |5 - 6|) = \min(14 + 4, 16 + 1) = \min(18, 17) = 17$.

For each cell from 1 to n find the temperature in it.

Input

The first line contains one integer q ($1 \le q \le 10^4$) — the number of test cases in the input. Then test cases follow. Before each test case, there is an empty line.

Each test case contains three lines. The first line contains two integers n ($1 \le n \le 3 \cdot 10^5$) and k ($1 \le k \le n$) — the length of the strip of land and the number of air conditioners respectively.

The second line contains k integers $a_1, a_2, \ldots, a_k$ ($1 \le a_i \le n$) — positions of air conditioners on the strip of land.

The third line contains k integers $t_1, t_2, \ldots, t_k$ ($1 \le t_i \le 10^9$) — temperatures of air conditioners.

It is guaranteed that the sum of n over all test cases does not exceed $3 \cdot 10^5$.

Output

For each test case output n integers separated by space: temperatures of air in cells.

Sample 1

Input	Output

<pre>5
 	
6 2
2 5
14 16
 	
10 1
7
30
 	
5 5
3 1 4 2 5
3 1 4 2 5
 	
7 1
1
1000000000
 	
6 3
6 1 3
5 5 5</pre>
	<pre>15 14 15 16 16 17
	36 35 34 33 32 31 30 31 32 33
	1 2 3 4 5
	1000000000 1000000001 1000000002 1000000003
	5 6 5 6 6 5</pre>

Problem K. Chef and Recipe

Time limit	1000 ms
Code length Limit	50000 B
OS	Linux

Read problems statements in Hindi, Mandarin Chinese, Russian, Vietnamese, and Bengali as well.

In Chefland, types of ingredients are represented by integers and recipes are represented by sequences of ingredients that are used when cooking. One day, Chef found a recipe represented by a sequence $A_1, A_2, \ldots, A_N$ at his front door and he is wondering if this recipe was prepared by him.

Chef is a very picky person. He uses one ingredient jar for each type of ingredient and when he stops using a jar, he does not want to use it again later while preparing the same recipe, so ingredients of each type (which is used in his recipe) always appear as a contiguous subsequence. Chef is innovative, too, so he makes sure that in each of his recipes, the quantity of each ingredient (i.e. the number of occurrences of this type of ingredient) is unique — distinct from the quantities of all other ingredients.

Determine whether Chef could have prepared the given recipe.

Input

- The first line of the input contains a single integer T denoting the number of test cases. The description of T test cases follows.
- The first line of each test case contains a single integer N.
- The second line contains N space-separated integers $A_1, A_2, \ldots, A_N$.

Output

For each test case, print a single line containing the string `"YES"` if the recipe could have been prepared by Chef or `"NO"` otherwise (without quotes).

Constraints

- $1 \le T \le 100$
- $1 \le N \le 10^3$
- $1 \le A_i \le 10^3$ for each valid i

Sample 1

Input	Output
3 6 1 1 4 2 2 2 8 1 1 4 3 4 7 7 7 8 1 7 7 3 3 4 4 4	YES NO NO

Example case 1: For each ingredient type, its ingredient jar is used only once and the quantities of all ingredients are pairwise distinct. Hence, this recipe could have been prepared by Chef.

Example case 2: The jar of ingredient 4 is used twice in the recipe, so it was not prepared by Chef.

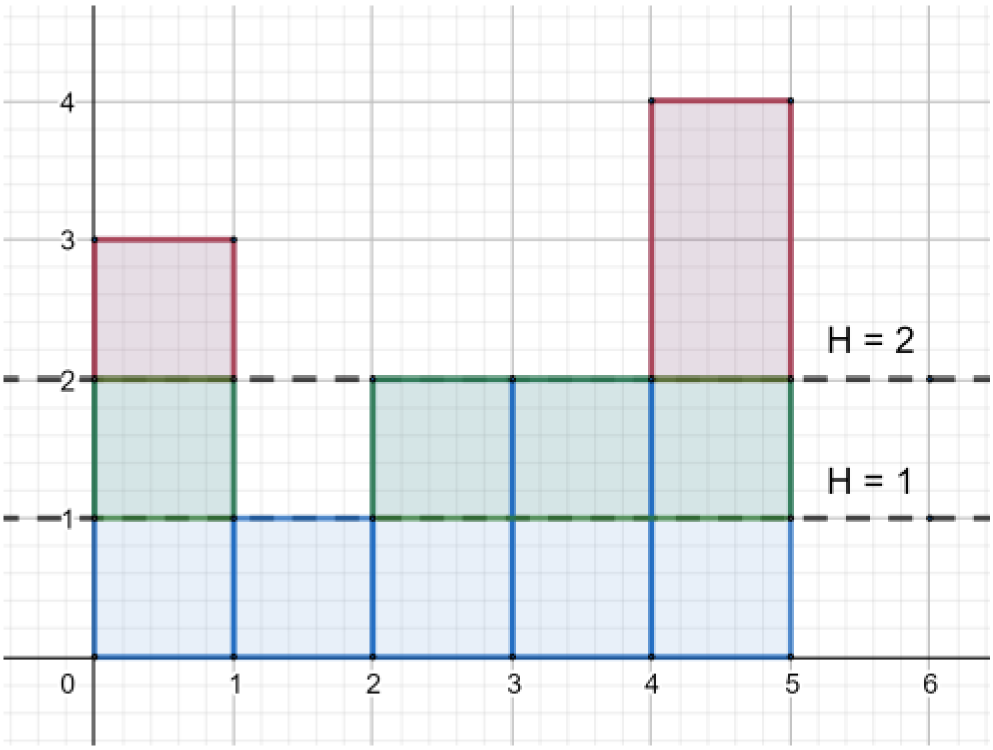
Problem L. Make It Equal

Time limit 2000 ms
Mem limit 262144 kB

There is a toy building consisting of n towers. Each tower consists of several cubes standing on each other. The i-th tower consists of h_i cubes, so it has height h_i.

Let's define operation *slice* on some height H as following: for each tower i, if its height is greater than H, then remove some top cubes to make tower's height equal to H. Cost of one "slice" equals to the total number of removed cubes from all towers.

Let's name slice as *good* one if its cost is lower or equal to k ($k \geq n$).



Calculate the minimum number of good slices you have to do to make all towers have the same height. Of course, it is always possible to make it so.

Input

The first line contains two integers n and k ($1 \leq n \leq 2 \cdot 10^5$, $n \leq k \leq 10^9$) — the number of towers and the restriction on slices, respectively.

The second line contains n space separated integers $h_1, h_2, \ldots, h_n$ ($1 \leq h_i \leq 2 \cdot 10^5$) — the initial heights of towers.

Output

Print one integer — the minimum number of good slices you have to do to make all towers have the same heigth.

Sample 1

Input	Output
5 5 3 1 2 2 4	2

Sample 2

Input	Output
4 5 2 3 4 5	2

Note

In the first example it's optimal to make 2 slices. The first slice is on height 2 (its cost is 3), and the second one is on height 1 (its cost is 4).

A problem that is simple to solve in one dimension is often much more difficult to solve in more than one dimension. Consider satisfying a boolean expression in conjunctive normal form in which each conjunct consists of exactly 3 disjuncts. This problem (3-SAT) is NP-complete. The problem 2-SAT is solved quite efficiently, however. In contrast, some problems belong to the same complexity class regardless of the dimensionality of the problem.

Given a 2-dimensional array of positive and negative integers, find the sub-rectangle with the largest sum. The sum of a rectangle is the sum of all the elements in that rectangle. In this problem the sub-rectangle with the largest sum is referred to as the *maximal sub-rectangle*.

A sub-rectangle is any contiguous sub-array of size 1×1 or greater located within the whole array. As an example, the maximal sub-rectangle of the array:

$$\begin{array}{rrrr} 0 & -2 & -7 & 0 \\ 9 & 2 & -6 & 2 \\ -4 & 1 & -4 & 1 \\ -1 & 8 & 0 & -2 \end{array}$$

is in the lower-left-hand corner:

$$\begin{array}{rr} 9 & 2 \\ -4 & 1 \\ -1 & 8 \end{array}$$

and has the sum of 15.

Input

The input consists of an $N \times N$ array of integers.

The input begins with a single positive integer N on a line by itself indicating the size of the square two dimensional array. This is followed by N^2 integers separated by white-space (newlines and spaces). These N^2 integers make up the array in row-major order (i.e., all numbers on the first row, left-to-right, then all numbers on the second row, left-to-right, etc.). N may be as large as 100. The numbers in the array will be in the range $[-127, 127]$.

Output

The output is the sum of the maximal sub-rectangle.

Sample Input

```
4
0 -2 -7  0 9  2 -6  2
-4  1 -4  1 -1
8  0 -2
```

Sample Output

```
15
```

Problem N. Bargaining Table

Time limit 2000 ms
Mem limit 262144 kB

Bob wants to put a new bargaining table in his office. To do so he measured the office room thoroughly and drew its plan: Bob's office room is a rectangular room $n \times m$ meters. Each square meter of the room is either occupied by some furniture, or free. A bargaining table is rectangular, and should be placed so, that its sides are parallel to the office walls. Bob doesn't want to change or rearrange anything, that's why all the squares that will be occupied by the table should be initially free. Bob wants the new table to sit as many people as possible, thus its perimeter should be maximal. Help Bob find out the maximum possible perimeter of a bargaining table for his office.

Input

The first line contains 2 space-separated numbers n and m $(1 \le n, m \le 25)$ — the office room dimensions. Then there follow n lines with m characters 0 or 1 each. 0 stands for a free square meter of the office room. 1 stands for an occupied square meter. It's guaranteed that at least one square meter in the room is free.

Output

Output one number — the maximum possible perimeter of a bargaining table for Bob's office room.

Sample 1

Input	Output
3 3 000 010 000	8

Sample 2

Input	Output
5 4 1100 0000 0000 0000 0000	16

Problem O. Karen and Coffee

Time limit 2500 ms
Mem limit 524288 kB

To stay woke and attentive during classes, Karen needs some coffee!

Karen, a coffee aficionado, wants to know the optimal temperature for brewing the perfect cup of coffee. Indeed, she has spent some time reading several recipe books, including the universally acclaimed "The Art of the Covfefe".

She knows n coffee recipes. The i-th recipe suggests that coffee should be brewed between l_i and r_i degrees, inclusive, to achieve the optimal taste.

Karen thinks that a temperature is *admissible* if at least k recipes recommend it.

Karen has a rather fickle mind, and so she asks q questions. In each question, given that she only wants to prepare coffee with a temperature between a and b, inclusive, can you tell her how many admissible integer temperatures fall within the range?

Input

The first line of input contains three integers, n, k $(1 \le k \le n \le 200000)$, and q $(1 \le q \le 200000)$, the number of recipes, the minimum number of recipes a certain temperature must be recommended by to be admissible, and the number of questions Karen has, respectively.

The next n lines describe the recipes. Specifically, the i-th line among these contains two integers l_i and r_i $(1 \le l_i \le r_i \le 200000)$, describing that the i-th recipe suggests that the coffee be brewed between l_i and r_i degrees, inclusive.

The next q lines describe the questions. Each of these lines contains a and b, $(1 \le a \le b \le 200000)$, describing that she wants to know the number of admissible integer temperatures between a and b degrees, inclusive.

Output

For each question, output a single integer on a line by itself, the number of admissible integer temperatures between a and b degrees, inclusive.

Sample 1

Input	Output
3 2 4 91 94 92 97 97 99 92 94 93 97 95 96 90 100	3 3 0 4

Sample 2

Input	Output
2 1 1 1 1 200000 200000 90 100	0

Note

In the first test case, Karen knows 3 recipes.

1. The first one recommends brewing the coffee between 91 and 94 degrees, inclusive.
2. The second one recommends brewing the coffee between 92 and 97 degrees, inclusive.
3. The third one recommends brewing the coffee between 97 and 99 degrees, inclusive.

A temperature is *admissible* if at least 2 recipes recommend it.

She asks 4 questions.

In her first question, she wants to know the number of admissible integer temperatures between 92 and 94 degrees, inclusive. There are 3: 92, 93 and 94 degrees are all admissible.

In her second question, she wants to know the number of admissible integer temperatures between 93 and 97 degrees, inclusive. There are 3: 93, 94 and 97 degrees are all admissible.

In her third question, she wants to know the number of admissible integer temperatures between 95 and 96 degrees, inclusive. There are none.

In her final question, she wants to know the number of admissible integer temperatures between 90 and 100 degrees, inclusive. There are 4: 92, 93, 94 and 97 degrees are all admissible.

In the second test case, Karen knows 2 recipes.

1. The first one, "wikiHow to make Cold Brew Coffee", recommends brewing the coffee at exactly 1 degree.
2. The second one, "What good is coffee that isn't brewed at at least 36.3306 times the temperature of the surface of the sun?", recommends brewing the coffee at exactly 200000 degrees.

A temperature is *admissible* if at least 1 recipe recommends it.

In her first and only question, she wants to know the number of admissible integer temperatures that are actually reasonable. There are none.

Problem P. Greg and Array

Time limit 1500 ms
Mem limit 262144 kB

Greg has an array $a = a_1, a_2, ..., a_n$ and m operations. Each operation looks as: l_i, r_i, d_i, $(1 \le l_i \le r_i \le n)$. To apply operation i to the array means to increase all array elements with numbers $l_i, l_i + 1, ..., r_i$ by value d_i.

Greg wrote down k queries on a piece of paper. Each query has the following form: x_i, y_i, $(1 \le x_i \le y_i \le m)$. That means that one should apply operations with numbers $x_i, x_i + 1, ..., y_i$ to the array.

Now Greg is wondering, what the array a will be after all the queries are executed. Help Greg.

Input

The first line contains integers n, m, k $(1 \le n, m, k \le 10^5)$. The second line contains n integers: $a_1, a_2, ..., a_n$ $(0 \le a_i \le 10^5)$ — the initial array.

Next m lines contain operations, the operation number i is written as three integers: l_i, r_i, d_i, $(1 \le l_i \le r_i \le n)$, $(0 \le d_i \le 10^5)$.

Next k lines contain the queries, the query number i is written as two integers: x_i, y_i, $(1 \le x_i \le y_i \le m)$.

The numbers in the lines are separated by single spaces.

Output

On a single line print n integers $a_1, a_2, ..., a_n$ — the array after executing all the queries. Separate the printed numbers by spaces.

Please, do not use the `%lld` specifier to read or write 64-bit integers in C++. It is preferred to use the `cin`, `cout` streams of the `%I64d` specifier.

Sample 1

<table>
<tr><th>Input</th><th>Output</th></tr>
<tr><td><pre>3 3 3
1 2 3
1 2 1
1 3 2
2 3 4
1 2
1 3
2 3</pre></td><td><pre>9 18 17</pre></td></tr>
</table>

Sample 2

<table>
<tr><th>Input</th><th>Output</th></tr>
<tr><td><pre>1 1 1
1
1 1 1
1 1</pre></td><td><pre>2</pre></td></tr>
</table>

Sample 3

<table>
<tr><th>Input</th><th>Output</th></tr>
<tr><td><pre>4 3 6
1 2 3 4
1 2 1
2 3 2
3 4 4
1 2
1 3
2 3
1 2
1 3
2 3</pre></td><td><pre>5 18 31 20</pre></td></tr>
</table>