The 4th Universal Cup

Stage 1: Grand Prix of Korolyov

October 3-4, 2025

This problem set should contain 11 problems on 14 numbered pages.

Based on

Northern Eurasia Camp 2025

Problem A. Grid Problem

Time limit: 3 seconds
Memory limit: 256 megabytes

Yotsugi was playing with a grid A of height h and width w, which was initially filled with zeroes by applying the following 2 operations in some order:

- Pick values i, j $(0 \le i \le h-1, 0 \le j \le w-1)$ and sign $+$ or $-$, and change values $A_{i,j}$, $A_{i,j+1}$, $A_{i+1,j}$, $A_{i+1,j+1}$ to $A_{i,j} \pm 2$, $A_{i,j+1} \pm 1$, $A_{i+1,j} \pm 1$, $A_{i+1,j+1} \pm 2$ correspondingly, where $\pm$ is replaced with the chosen sign. Here Yotsugi considers $A_{h,j}$ to be equivalent to $A_{0,j}$ and $A_{i,w}$ to be equivalent to $A_{i,0}$. In other words, she treats her grid as a torus.
- Pick values i, j $(0 \le i \le h-1, 0 \le j \le w-1)$ and sign $+$ or $-$, and change values $A_{i,j}$, $A_{i,j+1}$, $A_{i,j+2}$, $A_{i+1,j}$, $A_{i+1,j+1}$, $A_{i+1,j+2}$, $A_{i+2,j}$, $A_{i+2,j+1}$, $A_{i+2,j+2}$ to $A_{i,j} \pm 2$, $A_{i,j+1} \pm 5$, $A_{i,j+2} \pm 2$, $A_{i+1,j} \pm 5$, $A_{i+1,j+1} \pm 5$, $A_{i+1,j+2} \pm 5$, $A_{i+2,j} \pm 2$, $A_{i+2,j+1} \pm 5$, $A_{i+2,j+2} \pm 2$ correspondingly, where $\pm$ is replaced with the chosen sign. As in the previous operation, Yotsugi treats the grid as a torus.

For easier understanding of the operations, refer to the notes section.

On the next day, the grid was eaten by the fire-breathing slug, and now Yotsugi wonders how many possible grids she could have if, after she finished applying operations, all values $A_{i,j}$ lay in $[0, k]$. Notice that when Yotsugi was applying operations, values $A_{i,j}$ could be negative or exceed k. Since the answer can be large, help her find it modulo $10^9 + 9$.

Input

First line contains 3 integers h, w, k $(3 \le h, w \le 1\,000,\ 1 \le k \le 10^9)$ — size of the grid and maximum possible value in the grid.

Output

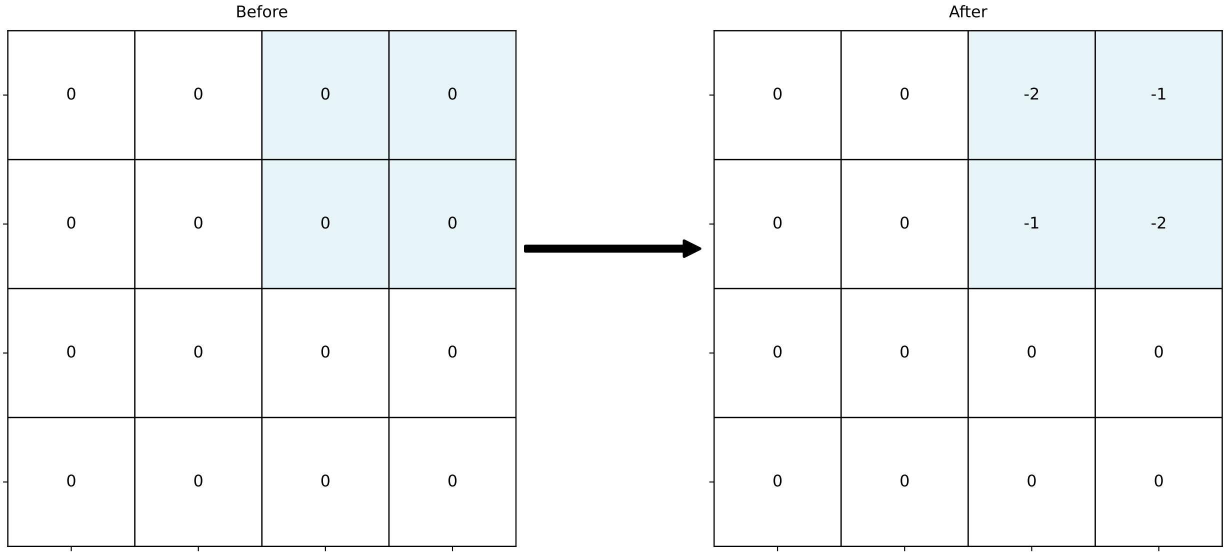
Output the answer to the problem modulo $10^9 + 9$.

Examples

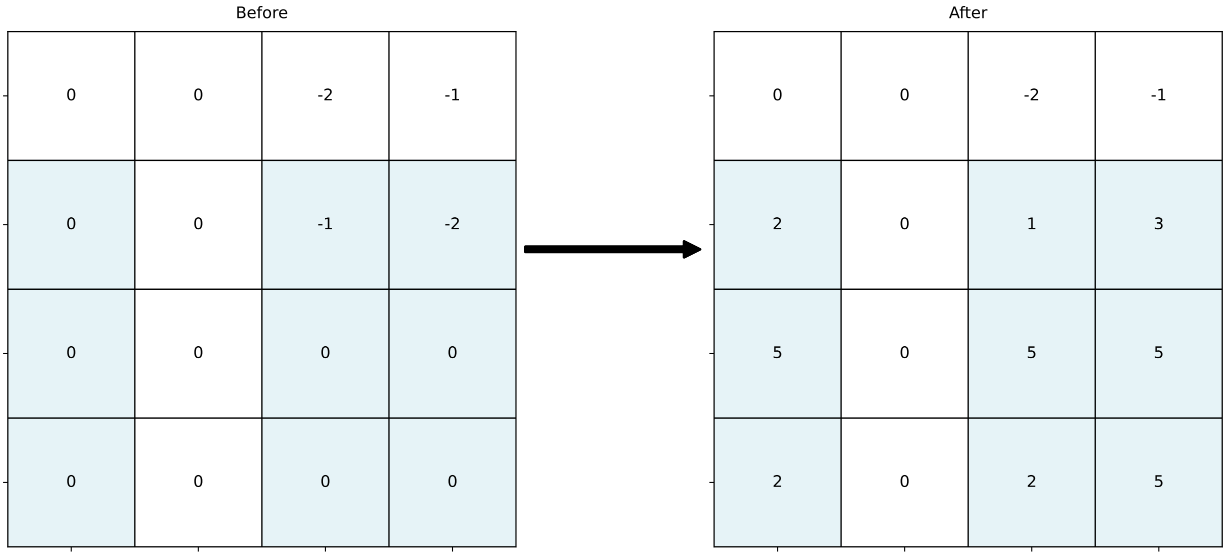
standard input	standard output
3 3 1	2
4 4 52	972950693
7 10 123	93519598

Note

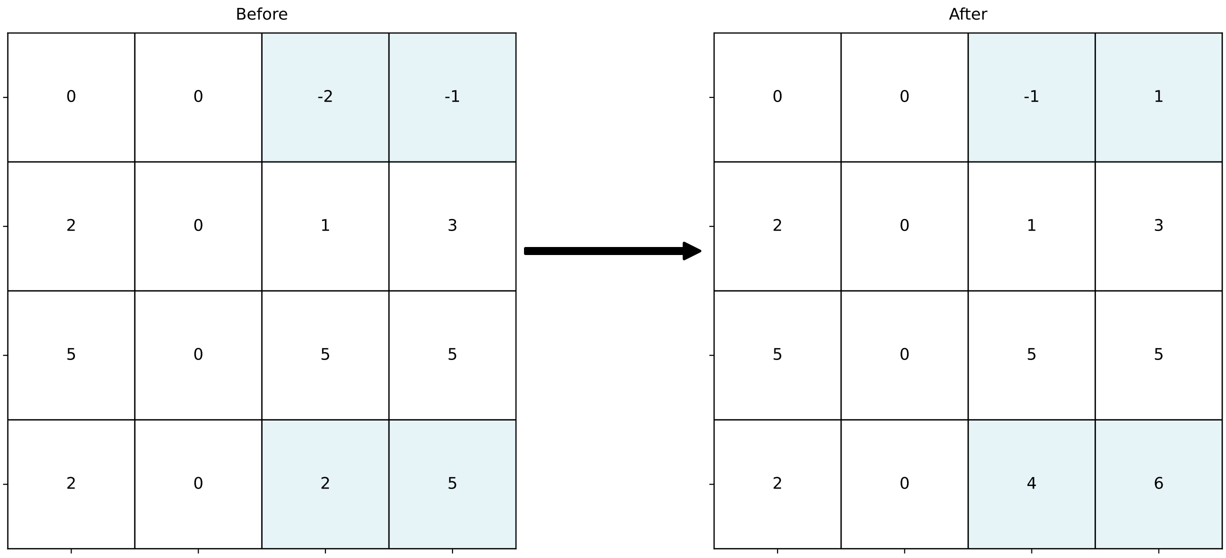
For example, if we had a 4×4 matrix, we could apply the operations in the following manner:



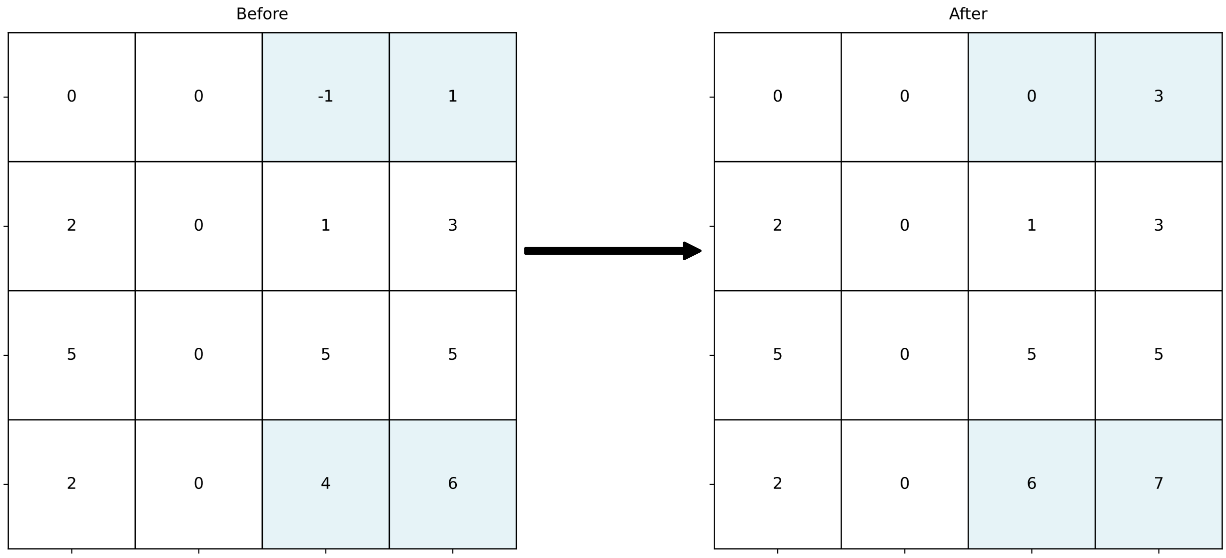
Applying the first operation with sign $-$.



Applying the second operation with sign $+$.



Applying the first operation with sign $+$.



Applying the first operation with sign $+$.

And since all values lie in $[0, 42]$, this is one of the matrices you are asked to count in the second sample.

Problem B. Domain Compression

Time limit: 2 seconds
Memory limit: 512 megabytes

Gojo has a tree with n vertices. He learned the technique of domain compression, which works as follows:

- First, he chooses some vertex v which was not previously deleted from the graph. Next, for each pair of vertices $u < w$ which are connected to v, he adds an undirected edge (u, w) if it was not previously contained in the graph. After that, he erases v from the graph with all the edges incident to it.

For each k ($1 \le k \le n$), he is interested in the total number of edges in the remaining graphs for all $\frac{n!}{(n-k)!}$ ways to perform the operations. Since the answers can be very large, output them modulo $998\,244\,353$.

Input

The first line contains an integer n ($2 \le n \le 10^5$) — the number of vertices in the tree.

Next, $n-1$ lines contain pairs of integers u_i, v_i ($1 \le u_i, v_i \le n$) — the description of the edges of the tree.

Output

Output n integers — the total number of edges for each k modulo $998\,244\,353$.

Examples

standard input	standard output
5 1 2 2 3 3 5 3 4	16 44 60 0 0
8 1 2 2 3 2 4 4 5 4 7 7 8 6 7	51 316 1596 6120 15720 20160 0 0

Problem C. Staple Stable

Time limit:	3 seconds
Memory limit:	256 megabytes

Hitagi Senjougahara has a piece of paper of height h and width w. The list is divided into 1×1 cells. Let's assume that the bottom left corner of the paper has coordinates $(0, 0)$ and the top right corner has coordinates (h, w).

Hitagi can make multiple cuts, each one being either vertical or horizontal. For horizontal cuts, Hitagi chooses integer height $h' \in [1, h-1]$ and cuts all pieces of paper that cross the line. For vertical cuts, Hitagi chooses integer width $w' \in [1, w-1]$ and cuts all pieces of paper that cross the line. So if she performs c_h horizontal cuts at different heights and c_w vertical cuts at different widths, she will end up with $(c_h + 1)(c_w + 1)$ pieces of paper.

Now Hitagi wonders what is the minimal number of cuts needed to make so all rectangular pieces of paper she ends up with have an area not exceeding s.

You are tasked to find the answer for t tuples h, w, s.

Input

First line contains integer t $(1 \le t \le 1\,000)$ — number of tests you need to solve.

Next t lines contain tuples of integers h, w, s $(1 \le h, w, s \le 10^9)$ — dimensions of the paper and wanted area.

Output

Output t lines, each containing 1 integer — answer for the tuple.

Example

standard input	standard output
5 2 2 1 1 7 2 4 4 17 4 4 15 120 216 34	2 3 0 1 55

Problem D. Sequence Is Not Subsequence

Time limit: 1 second
Memory limit: 256 megabytes

Along with a twisted fate, Nadeko has a string s which consists of lowercase letters. Her ultimate wish is to construct a string t for which the following conditions hold:

- t does not contain s as a subsequence.
- Every subsequence of s which does not coincide with s is contained in t as a subsequence.

Help her find string t of the minimal possible length. If there are multiple such strings, you are allowed to output any.

Input

A single line of the input contains string s ($2 \le |s| \le 10^6$) — Nadeko's string. It is guaranteed that s only contains lowercase letters.

Output

In the single line, output any string t of minimal length which satisfies the conditions of the problem.

Examples

standard input	standard output
`aaaa`	`aaa`
`bba`	`bab`

Note

In the second example, string ``bab'' contains ``bb'', ``ba'', ``a'', ``b'' as subsequences but does not contain ``bba''. It is also obvious that it is impossible to construct a correct sequence of length 2, so ``bab'' is a possible answer.

Problem E. Coffee Shops

Time limit: 2 seconds
Memory limit: 512 megabytes

Anton found a street full of coffee shops. The street can be thought of as a circle. On the inner side, there are n Varka coffee shops, while on the outer side are n Surf coffee shops. You know that there exist arrays a and b which represent the quality of coffee produced in the shops: a_i denotes the quality for the i-th Varka coffee shop and b_i denotes the quality for the i-th Surf coffee shop. Visitors sitting in some coffee shop do not like it when they see people having a better time on the opposing side of the street, so visitors of a coffee shop are sad if one of the following conditions holds:

- Visitors are sitting in the i-th Varka coffee shop and at least two of b_{i-1}, b_i, b_{i+1} are larger than a_i. Here b_0 stands for b_n and b_{n+1} stands for b_1.
- Visitors are sitting in the i-th Surf coffee shop and at least two of a_{i-1}, a_i, a_{i+1} are larger than b_i. Here a_0 stands for a_n and a_{n+1} stands for a_1.

We say that a coffee shop is sad if its visitors are sad. Now Anton wonders what is the maximum possible number of sad coffee shops if all values in a and b form a permutation of $1, \ldots, 2 \cdot n$ when combined. Help Anton and construct one possible pair a, b which achieves the maximum.

Input

The only line contains an integer n ($3 \le n \le 10^5$) — number of Varka and Surf coffee shops.

Output

In the first line, output n integers a_i representing the quality of the Varka coffee shops.

In the second line, output n integers b_i representing the quality of Surf coffee shops.

The union of a and b must form a permutation of $1, \ldots, 2 \cdot n$.

Examples

standard input	standard output
3	1 3 4 2 5 6
4	1 3 7 8 4 2 5 6

Note

For the first sample test, a coffee shop is sad if at least 2 values from the other array are larger than the value of that coffee shop (since there are only 3 shops on each side of the street), so Varka coffee shops with values $1, 3, 4$ are sad and the Surf coffee shop with value 2 is sad. It can be shown that it is impossible to achieve 5 sad coffee shops.

In the second test, we analyze two coffee shops as an example:

- The Varka shop with index 2 is sad because it has a value of 3 and at least 2 values of $\{4, 2, 5\}$ are larger than its value.
- The Surf shop with index 1 is not sad because it has a value of 4 and it does not hold that at least 2 values of $\{8, 1, 3\}$ are larger than its value.

If you consider all shops, you will find that there are 5 sad coffee shops, and it can be shown that larger values are not reachable.

Problem F. Yet Another MST Problem

Time limit: 3 seconds
Memory limit: 512 megabytes

Let us define $\mathbf{mex}(S)$ as the minimal non-negative integer which is not contained in S.

You are given a permutation p consisting of n elements from 0 to $n-1$. You also have m segments $[l_i, r_i]$ which you use to build a graph in the following way:

- For each pair of vertices $i \neq j$, you add an edge with cost equal to the **mex** of the union of values of the permutation on segments $[l_i, r_i]$ and $[l_j, r_j]$. More formally, the cost is equal to $\mathbf{mex}(\{p_k | k \in [l_i, r_i] \cup [l_j, r_j]\})$.

Find the minimum spanning tree of the described graph.

Input

The first line contains 2 integers n, m ($2 \le n, m \le 3 \cdot 10^5$) — length of the permutation and number of segments.

The second line contains n integers p_i ($0 \le p_i \le n-1$) — description of permutation p.

Next m lines contain pairs of integers l_i, r_i ($1 \le l_i \le r_i \le n$) — description of the segments.

Output

In a single line, output the answer to the problem.

Examples

standard input	standard output
6 4 3 1 5 2 0 4 3 6 1 2 2 5 4 6	8
2 2 0 1 1 2 1 2	2

Problem G. Cyclic Topsort

Time limit: 1.5 seconds
Memory limit: 256 megabytes

You are given a directed graph with n vertices and m edges. Your task is to find the length of the longest sequence a of distinct integers from 1 to n such that the following condition holds:

- Let l be the length of a. Then for all integers $i \in [\mathbf{2}, l]$ and all vertices v such that there exists an edge (v, a_i), there exists an index $j < i$ such that $a_j = v$.

Input

First line contains 2 integers n, m $(2 \le n \le 3 \cdot 10^5, 0 \le m \le 3 \cdot 10^5)$ — number of vertices and edges of the graph.

Next m lines contain pairs of integers u_i, v_i $(1 \le u_i, v_i \le n)$ — edges of the graph. The graph is allowed to have loops and multiple edges.

Output

In the first line, output the maximal length of the correct sequence l.

Examples

standard input	standard output
5 0	5
5 5 1 2 3 2 2 4 3 5 5 3	5
3 3 1 1 2 2 3 3	1

Note

In the first sample case, one of the correct sequences could be {1, 2, 3, 4, 5}.

In the second sample case, one of the correct sequences could be {3, 5, 1, 2, 4}.

In the third sample case, one of the correct sequences could be {1}. Sequences {2} and {3} also satisfy the conditions of the problem.

Problem H. Misread Problem

Time limit: 3 seconds
Memory limit: 256 megabytes

As you might know, the best way to come up with problems is misreading statements. So here you have one problem created in such a way.

There are m stones and k of their distributions into n boxes. Each distribution of stones can be represented as an array of non-negative integers a such that $a_1 + \ldots + a_n = m$. Alice will construct a distribution of stones b for which she will define $\text{cost}(b, a)$ as the minimal number of operations described below needed to make a equal to b:

- Take 1 stone from box i ($1 \le i \le n$) with a positive number of stones and put it in box j ($1 \le j \le n$).

She wants to find the minimal sum of costs for the optimal distribution b to all given distributions.

Input

First line contains 3 integers n, m, k ($1 \le n, k \le 400, 1 \le m \le 10^9$) — number of boxes, stones, and distributions.

Each of the next k lines contains n non-negative integers $a_{i,j}$ ($a_{i,1} + \ldots + a_{i,n} = m$) — description of the distributions.

Output

Output a single integer — answer to the problem.

Examples

standard input	standard output
5 12 3 3 0 4 1 4 5 2 3 1 1 1 2 3 5 1	8
1 1 2 1 1	0

Note

For the first sample case, one possible optimal distribution is $b = \{3, 2, 3, 2, 2\}$. For this distribution b we have:

- For $a = \{3,\ 0,\ 4,\ 1,\ 4\}$ we have $\text{cost}(b, a) = 3$ since we can apply operations in the following order: $\{3, 0, 4, 1, 4\} \to \{3, 1, 4, 1, 3\} \to \{3, 2, 4, 1, 2\} \to \{3, 2, 3, 2, 2\}$.
- For $a = \{5,\ 2,\ 3,\ 1,\ 1\}$ we have $\text{cost}(b, a) = 2$ since we can apply operations in the following order: $\{5, 2, 3, 1, 1\} \to \{4, 2, 3, 1, 2\} \to \{3, 2, 3, 2, 2\}$.
- For $a = \{1,\ 2,\ 3,\ 5,\ 1\}$ we have $\text{cost}(b, a) = 3$ since we can apply operations in the following order: $\{1, 2, 3, 5, 1\} \to \{1, 2, 3, 4, 2\} \to \{2, 2, 3, 3, 2\} \to \{3, 2, 3, 2, 2\}$.

So the sum of costs is 8. It can be shown that it is not possible to obtain a smaller sum of costs.

Problem I. troS XEM

Time limit: 1.5 seconds
Memory limit: 512 megabytes

Let us define $\mathbf{mex}(S)$ as the minimal non-negative integer which is not contained in S.

Mayoi Hachikuji has an array a consisting of n elements which she wants to be sorted in non-decreasing order. To do so, she can perform the following operation any number of times (possibly zero):

- Choose integer $i \in [1, \text{length}(a) - 2]$ and replace a_i, a_{i+1}, a_{i+2} with their mex. Notice that after applying this operation, the length of the array decreases by 2.

Find the minimum number of operations Mayoi needs to perform to make the array sorted in non-decreasing order. In case it is impossible to do so, output -1.

Input

First line of the input contains integer n ($2 \le n \le 5 \cdot 10^5$) — length of the array a.

Next line contains n integers a_i ($0 \le a_i \le 10^9$) — elements of the array a.

Output

Output one number — the answer to the problem.

Examples

standard input	standard output
3 4 1 3	1
9 3 0 2 2 3 0 4 7 8	2
2 4 3	-1

Problem J. Yet Another Constructive Problem

Time limit: 1 second
Memory limit: 256 megabytes

You are given a permutation a of length n. You are also given positive integers k and m ($1 \le k \le m \le n$). You are asked to tell whether there exists a subsequence b of a of length m such that the length of the longest increasing subsequence of b equals k. In case such a subsequence exists, you are asked to print any correct one.

Input

First line contains three integers n, m, k ($1 \le k \le m \le n \le 5\,000$) — length of the permutation, length of the subsequence b, and length of the longest increasing subsequence of b.

Output

In the first line, output ‘‘Yes’’ or ‘‘No’’. Checker is case insensitive.

In the second line, output one of the correct subsequences if such exists.

Examples

standard input	standard output
5 2 2 4 5 3 2 1	Yes 4 5
5 2 2 5 4 3 2 1	No
6 4 2 1 3 4 2 6 5	Yes 4 2 6 5

Problem K. Robot Construction

Time limit: 1 second
Memory limit: 256 megabytes

You have an ability to create robots of height lying in $[0, d]$. To test them, you constructed a line of length n containing some obstacles. The description of obstacles is given to you as an array a. If $a_i = 0$, then there is no obstacle at the i-th position, and otherwise, there is an obstacle of height $a_i > 0$.

To test your robot, you select a segment $[l, r]$ and run it through all obstacles contained in the segment. When the robot encounters an obstacle with value $a_i > 0$, one of the two possibilities happens:

- If the current height of the robot h is smaller than a_i, then nothing happens since the robot is too short to reach the obstacle.
- If the current height of the robot h is at least a_i, then the new height changes to $h' = h - a_i$.

Now you need to answer q queries. For each of the q segments $[l_i, r_i]$, you need to tell the maximum possible height with which the robot can end up after running through the obstacles in the segment if the initial height was picked from the interval $[0, d]$.

Input

First line contains a pair of integers n, q, d $(1 \le n, q \le 3 \cdot 10^5, 1 \le d \le 10^9)$ — length of the array and number of queries.

Second line contains n integers a_i $(0 \le a_i \le 10^9)$ — description of the array.

Next q lines contain a pair of integers $[l_i, r_i]$ $(1 \le l_i \le r_i \le n)$ — description of the queries.

Output

Output q integers — answers to the queries.

Examples

standard input	standard output
5 3 5 0 2 6 1 3 5 5 1 5 1 3	2 2 3
7 5 10 7 6 2 5 0 1 4 1 3 1 7 4 7 2 5 4 6	3 2 3 3 4