Problem A. Bitwise Queries

Input file:	*standard input*
Output file:	*standard output*
Time limit:	5 seconds
Memory limit:	512 mebibytes

You are given an array a of size n and you need to perform m queries on it. There are three types of queries:

1. `&` l r x: change a_i to (a_i `AND` x) for all $i = l, l+1, \ldots, r$;
2. `|` l r x: change a_i to (a_i `OR` x) for all $i = l, l+1, \ldots, r$;
3. `?` l r: find the minimal value among $a_l, a_{l+1}, \ldots, a_r$.

Output the answers for all queries of the third type.

Input

The first line contains one integer n ($1 \le n \le 5 \cdot 10^5$) — the size of the array.

The second line contains n space-separated integers a_i ($0 \le a_i < 2^{30}$) — the elements of the array.

The third line contains one integer m ($1 \le m \le 2 \cdot 10^5$) — the number of queries.

Next m lines contain descriptions of queries in the format described above. For all queries $1 \le l \le r \le n$, for queries of the first and second types $0 \le x < 2^{30}$.

Output

For each query of the third type, print the answer on a separate line.

Example

standard input	standard output
5 1 2 3 4 5 4 & 1 2 6 | 3 5 4 ? 1 2 ? 3 5	0 4

Problem B. Sum of distinct

Input file:	*standard input*
Output file:	*standard output*
Time limit:	5 seconds
Memory limit:	256 mebibytes

You are given n positive integers — a_1, a_2, ..., a_n. You have to perform two type of queries — change one number and get sum of distinct numbers on subsegment $a_i, ..., a_j$. For example, for segment $a_4 = 5$, $a_5 = 13$, $a_6 = 5$ result of the query will be $5 + 13 = 18$, because set of numbers on the segment is $\{5, 13\}$.

Input

In the first line one integer n ($1 \le n \le 50\,000$) is given. In the second line you are given n positive integers a_i ($1 \le a_i \le 10^9$). In the third line you are given integer m — number of queries ($1 \le m \le 10^5$). Next m lines contain queries. Update queries given as "`U` u v" ($1 \le u \le n$, $1 \le v \le 10^9$), meaning that after operation a_u changes to v. Subsegment query given as "`Q` l r" ($1 \le l \le r \le n$). On subsegment query you have to output sum of distinct numbers on subsegment $a_{l..r}$.

Output

For each query output answer on a separate line.

Example

standard input	standard output
5 1 2 4 2 3 3 Q 2 4 U 4 7 Q 2 4	6 13

Problem C. Inception

Input file:	*standard input*
Output file:	*standard output*
Time limit:	5 seconds
Memory limit:	256 mebibytes

Aliens attack city χ! The city have n buildings, each building is a point x_i, y_i. Aliens can conquer any square with sides parallel to coordinate axis. They want to conquer at least k buildings. But conquering big area is tedious and time consuming, so aliens want to minimize sidelength of the square.

Please help aliens to find smallest square, containing at least k buildings.

Input

First line of input contains two integers n ($2 \le n \le 50\,000$) — number of buildings — and k ($2 \le k \le n$) — minimal number of buildings aliens want to conquer. Next n lines contains points coordinates of buildings x_i and y_i ($1 \le x_i, y_i \le 10\,000$) — integer, separated by space.

Output

In the first line output single integer A — minimal sidelength of the square, containing at least k points. In the second line output two integers — coordinates x and y of the bottom-left corner of the square (so, square will contain all points x_i, y_i that satisfy $x \le x_i \le x + a$ и $y \le y_i \le y + a$). If there are multiple solutions, output any.

Examples

standard input	standard output
3 3 1 2 2 1 3 4	3 1 1

Problem D. Mega-inversions

Input file:	*standard input*
Output file:	*standard output*
Time limit:	1 second
Memory limit:	256 mebibytes

Inversion in permutation $p_1, p_2, \ldots, p_n$ is pair (i, j) such that $i < j$ and $p_i > p_j$.

Define *mega-inversion* in permutation $p_1, p_2, \ldots, p_N$ as a triple (i, j, k) such that $i < j < k$ and $p_i > p_j > p_k$. Find number of mega-inversion in given permutation.

Input

The first line of input contains one integer n ($1 \leqslant n \leqslant 100\,000$). Next n numbers contains permutation: $p_1, p_2, \ldots, p_n$ ($1 \leqslant p_i \leqslant n$), all p_i are pairwise distinct. Numbers are separated by newlines.

Output

Output should contain single integer — number of mega-inversion in permutation $p_1, p_2, \ldots, p_N$.

Examples

standard input	standard output
4 4 3 2 1	4

Problem E. Near and More

Input file:	`nearandmore.in`
Output file:	`nearandmore.out`
Time limit:	1 second
Memory limit:	256 mebibytes

You are given array a of n integers. You have to process two types of queries:

0. `set(i, x)` – assign value x to i-th element `a[i] = x;`
1. `get(i, x)` – find $\min k$: $k \geq i$ and $a_k \geq x$.

Input

The first line of input contain two integers: length of the array n and number of queries m ($1 \leq n, m \leq 200\,000$).

The second line contains n integers – elements of array a ($0 \leq a_i \leq 200\,000$).

Next m lines contains queries, each query contain three integers t, i, x. First integer t equals 0 or 1 – type of the query. $t = 0$ means query of type `set`, $t = 1$ corresponds to query `get`, $1 \leq i \leq n$, $0 \leq x \leq 200\,000$. All indicies are 1-based.

Output

For each query of type `get` output corresponding k on a separate line. If there's no such k, output -1.

Examples

nearandmore.in	nearandmore.out
4 5 1 2 3 4 1 1 1 1 1 3 1 1 5 0 2 3 1 1 3	1 3 -1 2

Problem F. Tom and his friends

Input file:	*standard input*
Output file:	*standard output*
Time limit:	1 second
Memory limit:	256 mebibytes

Tom and his friends paint a fence using different colors. Each of them paint some number of consecutive sections of the fence in particular color, colors may be the same for different friends. Friends color fence in order from 1st to the m-th. For each color find number of sections of this color after all queries are performed.

Input

The first line contains two integers: n ($1 \leqslant n \leqslant 10^9$) and k ($1 \leqslant k \leqslant 50000$) — number of sections and number of different colors.

The second line contains single integer m ($0 \leqslant m \leqslant 50000$) — number of friends.

Next m lines contains description of i-th friend, given as three integers c_i, l_i, r_i ($1 \leqslant c_i \leqslant k$, $1 \leqslant l_i \leqslant r_i \leqslant n$) — number of color that i-th friend used, number of the first and the last painted section.

Output

Output single line with k integers: i-th number should equal to number of sections painted in the i-th color.

Examples

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

Problem G. Abdullah and Parantheses

Input file:	*standard input*
Output file:	*standard output*
Time limit:	2 second
Memory limit:	256 mebibytes

Abdullah has a string s of length n consisting of parentheses («(» and «)»).

Abdullah needs to answer the following queries: given 2 integers l_i, r_i $(1 \leqslant l_i \leqslant r_i \leqslant n)$, find the longest regular parentheses subsequence of the substring $s_{l_i}, s_{l_i+1}, \ldots, s_{r_i}$. Help Abdullah answer m such queries. See notes for the definition of a regular parentheses subsequence.

Input

The first line contains a string s, $s_1, s_2, \ldots, s_n$ $(1 \leqslant n \leqslant 10^6)$ consisting of symbols «(» and «)». The second line contains the number of queries m $(1 \leqslant m \leqslant 10^5)$. The following m lines each contain a pair of integers. The i-th line contains 2 integers l_i, r_i $(1 \leqslant l_i \leqslant r_i \leqslant n)$ – the parameters of the i-th query.

Output

For each query, output the answer on a separate line. The answers should be given in the same order as the input queries.

Examples

standard input	standard output
())(())(())(	0
7	0
1 1	2
2 3	10
1 2	4
1 12	6
8 12	6
5 11	
2 10	

Notes

A *subsequence* of length $|x|$ of string $s = s_1 s_2 \ldots s_{|s|}$ (where $|s|$ is the length of string s) is given by $x = s_{k_1} s_{k_2} \ldots s_{k_{|x|}}$ $(1 \leqslant k_1 < k_2 < \cdots < k_{|x|} \leqslant |s|)$.

A *regular parentheses sequence* is a parentheses sequence which can be converted to a correct arithmetic expression by inserting the symbols «1» and «+». For example, the sequences «()()», «(())» are regular (can be converted to «(1)+(1)» and «((1+1)+1)»), while «)(» and «(» are not.

The subsequence corresponding to the third query is «()».

The subsequence corresponding to the fourth query is «()(())(())».

Problem H. Add and max at subsegment

Input file:	*standard input*
Output file:	*standard output*
Time limit:	1 second
Memory limit:	256 mebibytes

Implement efficient data structure for two types of queries: addition at subsegment and maximum at subsegment.

Input

The first line contains one integer $n(1 \leqslant n \leqslant 100000)$ – size of the array.

The second line contains n integers between 0 and 100000 – elements of the array.

The third line contains one integer $m(1 \leqslant m \leqslant 30000)$ – number of queries.

Next m lines contains description of the queries. The first you are given one letter, m – for maximum, a – for addition.

m is followed by two integers l, r, meaning you need to find maximum among elements $a_l, \ldots, a_r$.

a is followed by three integers l, r, a, meaning you need to add a to elements $a_l, \ldots, a_r (0 \leqslant add \leqslant 100000)$. In both cases, $1 \leqslant l \leqslant r \leqslant n$.

Output

Output answers in one line separated by space.

Examples

standard input	standard output
5 2 4 3 1 5 5 m 1 3 a 2 4 100 m 1 3 a 5 5 10 m 1 5	4 104 104

Problem I. GCD on subsegment

Input file:	*standard input*
Output file:	*standard output*
Time limit:	3 seconds
Memory limit:	256 mebibytes

Implement structure for efficient calculation of GCD (Greatest Common Divisor) on subsegment.

Input

The first line contains one integer $n(1 \leqslant n \leqslant 10^5)$ – size of the array.

The second line contains n integers from 1 до 10^9 – array elements.

The third line contains one integer $k(1 \leqslant k \leqslant 10^6)$ – number of queries.

Next k lines contains two integers l_i, r_i $(1 \leqslant l_i \leqslant r_i \leqslant n)$ each – left and right bounds of the query.

Output

For each query output answer – $\texttt{gcd}(a_{l_i}, \ldots, a_{r_i})$ on a separate line.

Examples

standard input	standard output
5 2 2 2 1 5 2 2 3 2 5	2 1

Problem J. Range Variation Query

Input file:	*standard input*
Output file:	*standard output*
Time limit:	0.5 seconds
Memory limit:	256 mebibytes

You are given array a_n. At the beginning a_n are given by the following formula: $a_n = n^2 \bmod 12345 + n^3 \bmod 23456$.

You have to process the following queries:

- given i, j, find $\max(a_i, \ldots, a_j) - \min(a_i, \ldots, a_j)$.
- set element a_i value x.

Input

First line of input file contains positive number k — number of queries ($1 \leqslant k \leqslant 100\,000$). Next k lines describes queries in separate lines. i-th query described by two integers x_i, y_i.

If $x_i > 0$, you have to find the difference between minimum and maximum on segment $a_{x_i}, \ldots, a_{y_i}$. In this case $1 \leqslant x_i \leqslant y_i \leqslant 100\,000$.

If $x_i < 0$, you have to set element $a_{|x_i|}$ value y_i. In this case $-100\,000 \leqslant x_i \leqslant -1$ and $|y_i| \leqslant 100\,000$.

Output

For each query output answer on a separate line.

Examples

standard input	standard output
7 1 3 2 4 -2 -100 1 5 8 9 -3 -101 2 3	34 68 250 234 1