ShanDong Multi-University Training 3

Contest Session

June 25, 2022

Problem List

A	Problem A
B	Problem B
C	Problem C
D	Problem D
E	Problem E
F	Problem F
G	Problem G
H	Problem H
I	Problem I
J	Problem J
K	Problem K
L	Problem L

Prepared by QLU.

Problem A. A

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 seconds
Memory limit:	256 mebibytes

Mole is hungry again. He found one ant colony, consisting of n ants, ordered in a row. Each ant i ($1 \leq i \leq n$) has a strength s_i.

In order to make his dinner more interesting, Mole organizes a version of «Hunger Games» for the ants. He chooses two numbers l and r ($1 \leq l \leq r \leq n$) and each pair of ants with indices between l and r (inclusively) will fight. When two ants i and j fight, ant i gets one battle point only if s_i divides s_j (also, ant j gets one battle point only if s_j divides s_i).

After all fights have been finished, Mole makes the ranking. An ant i, with v_i battle points obtained, is going to be freed only if $v_i = r$ - l, or in other words only if it took a point in every fight it participated. After that, Mole eats the rest of the ants. Note that there can be many ants freed or even none.

In order to choose the best sequence, Mole gives you t segments $[l_i,r_i]$ and asks for each of them how many ants is he going to eat if those ants fight.

Input

The first line contains one integer n ($1 \leq n \leq 10^5$), the size of the ant colony.

The second line contains n integers $s_1,s_2,\ldots,s_n$ ($1 \leq s_i \leq 10^9$), the strengths of the ants.

The third line contains one integer t ($1 \leq t \leq 10^5$), the number of test cases.

Each of the next t lines contains two integers l_i and r_i ($1 \leq l_i \leq r_i \leq n$), describing one query.

Output

Print to the standard output t lines. The i-th line contains number of ants that Mole eats from the segment $[l_i,r_i]$.

Examples

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

Explanations

In the first test battle points for each ant are v=[4,0,2,0,2], so ant number 1 is freed. Mole eats the ants 2, 3, 4, 5.

In the second test case battle points are v=[0,2,0,2], so no ant is freed and all of them are eaten by Mole.

In the third test case battle points are v=[2,0,2], so ants number 3 and 5 are freed. Mole eats **only the ant** 4.

In the fourth test case battle points are v=[0,1], so ant number 5 is freed. Mole eats the ant 4.

Problem B. B

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 seconds
Memory limit:	64 mebibytes

You are given n strings and a new rule of overload the less-than sign, please print the K-th lexicographically smallest string.

Every string contains only lowercase letters.

A string a is lexicographically smaller than a string b if and only if one of the following holds:

- there exists a position i, $a_i < b_i$ and for all $j < i$, $a_j = b_j$,
- for every i, $a_i = b_i$ and the length of a is less than the length of b.

Input

The first line gives 26 characters from a to z, each character will appear only once. The earlier a letter appears, the smaller it is. For example, $acb \cdots xyz$ in the first line means that character 'c' is less than character 'b' and character 'c' is greater than character 'a'.

The second line contains an integer n $(1 < n < 10^3)$, indicating the number of strings.

Next n lines, each line contains a string, whose length of the string is less than 10^3.

The next line contains an integer K $(1 \leq K \leq n)$, indicating the K-th smallest string you need to find.

Output

The K-th lexicographically smallest string.

Examples

standard input	standard output
acbdefghijklmnopqrstuvwxyz 2 abc acb 1	acb
zabcdefghijklmnopqrstuvwxy 2 a b 1	a

Problem C. C

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	512 mebibytes

Now you have an $n \times n$ map. It contains obstacles '$*$', space '.', playerA 'a', playerB 'b'.

You can control two players to move up, down, left or right **at the same time**. For example, now A and B are in (x_a, y_a) and (x_b, y_b), next time the two players can go to:

- $(x_a, y_a + 1)$ and $(x_b, y_b + 1)$
- $(x_a, y_a - 1)$ and $(x_b, y_b - 1)$
- $(x_a + 1, y_a)$ and $(x_b + 1, y_b)$
- $(x_a - 1, y_a)$ and $(x_b - 1, y_b)$

Note that if the next location is a boundary or obstacle the player will not move.

You need to find the minimum number of steps when two players can meet.

Input

The first line contains one integer n ($2 \le n \le 50$).

For the next n lines, each line contains a string of length n.

Output

If in any case, the two players do not meet, print "no solution"in one line.

Otherwise, output the minimum number of steps.

Examples

standard input	standard output
6 .a.... *..... ....b*.. ..**..	4
4 a*.. *... ....b	no solution
4 .ab.	2

standard input	standard output
7 ..a.b..	4

Problem D. D

Input file:	`standard input`
Output file:	`standard output`
Time limit:	1 seconds
Memory limit:	256 mebibytes

You are given a sequence a consisting of n integers. Find the maximum possible value of $a_i \mod a_j$ (integer remainder of a_i divided by a_j), where $1 \le i, j \le n$ and $a_i \ge a_j$.

Input

The first line contains integer n — the length of the sequence ($1 \le n \le 2 \times 10^5$).

The second line contains n space-separated integers a_i ($1 \le a_i \le 10^6$).

Output

Print the answer to the problem.

Examples

standard input	standard output
3 3 4 5	2

Problem E. E

Input file:	`standard input`
Output file:	`standard output`
Time limit:	4 seconds
Memory limit:	1024 mebibytes

You are given integer sequences $A = (a_1, a_2, \ldots, a_N)$ and $B = (b_1, b_2, \ldots, b_N)$,each of length N.

For $i = 1, \ldots, Q$,answer the query in the following format.

- If the set of values contained in the first x_i terms of $A, (a_1, a_2, \ldots, a_N)$, and the set of values contained in the first y_i terms of $B, (b_1, b_2, \ldots, b_N)$, are equal, then print "Yes"; otherwise, print "No".

Constraints

- $1 \leq N, Q \leq 2 \times 10^5$
- $1 \leq a_i, b_i \leq 10^9$
- $1 \leq x_i, y_i \leq N$
- All values in input are integers.

Input

Input is given from Standard Input in the following format:

N
$a_1 \ldots a_N$
$b_1 \ldots b_N$
Q
$x_1\ y_1$
$\vdots$
$x_Q\ y_Q$

Output

Print Q lines. The i-th line should contain the response to the i-th query.

Examples

standard input	standard output
5 1 2 3 4 5 1 2 2 4 3 7 1 1 2 2 2 3 3 3 4 4 4 5 5 5	Yes Yes Yes No No Yes No

Explanations

Note that sets are a concept where it matters only whether each value is contained or not.

For the 3-rd query, the first 2 terms of A contain one 1 and one 2, while the first 3 terms of B contain one 1 and two 2's. However, the sets of values contained in the segments are both $\{1, 2\}$, which are equal.

Also, for the 6-th query, the values appear in different orders, but they are still equal as sets.

Problem F. F

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	512 mebibytes

Ganyu just learns the persistent segment tree and decides to practice immediately. Since Paimon has become a master of the persistent segment tree, Ganyu asks her for help. Therefore Paimon gives her an easy problem to start:

The elements in the sequence have two states - locked and unlocked. If one element a_i is locked, it cannot be modified, but the unlocked state has no effect. More formally, you can only modify elements that are in the unlocked state.

Given a sequence $a_1, a_2, \ldots, a_n$ of length n with initial values of 1 and states are unlocked.

Paimon will apply m modifications to the sequence. There are 3 types of modifications:

- flip p: change the state of the element a_p.
- mul l r x: for every element a_i in the subarray $(a_l, a_{l+1}, \ldots, a_r)$, replace unlocked a_i with $a_i \times x$.
- div l r x: for every element a_i in the subarray $(a_l, a_{l+1}, \ldots, a_r)$, you should check if all of them(including elements in unlocked and locked states) are divisible by x. If it holds, replace unlocked a_i with $a_i \div x$, otherwise do not perform the operation.

Please help Ganyu to check whether each div modification can be performed.

Input

There is only one test case in each test file.

The first line of the input contains two integers n and m ($1 \le n, m \le 10^5$) indicating the length of the sequence, and the number of modifications.

Then the following m lines indicate modifications. Each of them must be in one of the following forms.

- flip p
- mul l r x
- div l r x

All p,l,r and x mentioned in these m lines satisfy $1 \le p \le n$, $1 \le l \le r \le n$ and $1 \le x \le 30$.

Output

For each div modifications, print "YES"on the single line if all the elements in the subarray are divisible by x. Otherwise, print "NO".

You can print each letter in any case (upper or lower).

Examples

standard input	standard output
5 11	YES
mul 1 5 18	NO
div 2 3 6	NO
div 1 2 9	NO
flip 2	YES
div 1 2 9	YES
mul 2 2 3	
div 1 2 9	
div 2 3 3	
flip 2	
mul 2 2 3	
div 1 2 9	

Explanations

Here is the explanation for the example:

- $[18, 18, 18, 18, 18]$
- $[18, 3, 3, 18, 18]$
- $[18, 3, 3, 18, 18]$, a_2 and a_3 are not divsiable by 9
- $[18, \underline{\mathbf{3}}, 3, 18, 18]$, a_2 is locked
- $[18, \underline{\mathbf{3}}, 3, 18, 18]$, a_2 is not divsiable by 9
- $[18, \underline{\mathbf{3}}, 3, 18, 18]$, a_2 is locked so you can't change its value
- $[18, \underline{\mathbf{3}}, 3, 18, 18]$, a_2 is not divsiable by 9
- $[18, \underline{\mathbf{3}}, 1, 18, 18]$, a_2 and a_3 are divsiable by 3 but a_2 is locked, so you only change a_3
- $[18, 3, 1, 18, 18]$, a_2 is unlocked
- $[18, 9, 1, 18, 18]$
- $[2, 1, 1, 18, 18]$

Problem G. G

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	256 mebibytes

Alice is going to go to the supermarket, which is p meters away from her. She can walk or ride a bike. There are n bike stops on the road, and **she can only get on and off at these stops if she rides a bike**.

She needs to pay to ride a bike. One yuan allows riding at most s meters(less than s meters costs one yuan, too). Formally, if two stops are x meters away from each other and Alice chooses to bike, it costs her $\lceil \frac{x}{s} \rceil$ yuan.

Now, she has k yuan, and she wants to know, what is the minimum number of meters to walk.

Input

The first line contains three integers n, p, s $(1 \le n \le 10^6, 1 \le p \le 10^9, 1 \le s \le 10^9)$ indicating the number of the bike shops, the location of the store, and maximum meters that one yuan allows to ride.

The second line contains n integers, $a_1, a_2, \ldots, a_n (0 \le a_i \le p)$ indicating the position of the i-th stop $(\forall i \le j, a_i \le a_j)$.

The third line contains an integer $k (1 \le k \le 5)$, indicating Alice has k yuan.

Output

Output one line containing one integer indicating the minimum number of meters that Alice needs to walk.

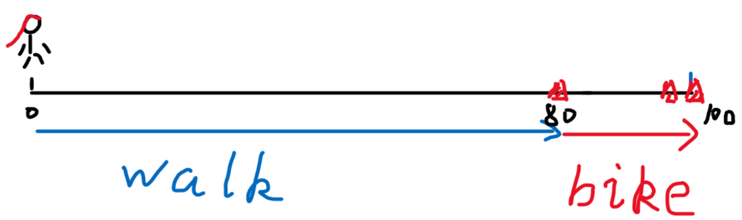
Examples

standard input	standard output
1 10 10 2 5	10
3 100 10 80 99 100 2	80
4 10 3 1 2 6 7 1	9

Explanations

In the first test case, Alice cannot find any other bike stops to get off, so she has to walk 10 meters.

In the second test case:



Alice can walk 80 meters and then pay two yuan to ride 20 meters.

In the third test case, Alice can only ride from 1 to 2 or 6 to 7, so she has to walk the remaining 9 meters.

Problem H. H

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

Diana is now playing a game with Ava called "Nim Game". They take turns removing stones from several piles. On each turn, a player must remove one or more stones from exactly one pile. The one who can't take any stone loses the game.

They play several rounds of the games on N piles. The i-th pile contains A_i stones. Each round, they play Nim game by selecting several piles (non-zero) from an interval $A_l, A_{l+1}, \ldots, A_r$. Sometimes, they add x stones to each pile in interval $A_l, A_{l+1}, \ldots, A_r$. After that, the piles in the interval turns to $A_l + x, A_{l+1} + x, \ldots, A_r + x$.

Ava plays first-hand in the game, and Diana chooses the piles to play the game. Please you help Diana to determine if there is a way to choose of piles to guarantee her sure win.

Input

The first line contains two positive integers $N(1 \le N \le 10^5)$ and $M(1 \le M \le 10^5)$, indicating the number of piles and the number of operations. The next line contains N integers $A_1, A_2, \ldots, A_n(1 \le A_i \le 10^9)$, indicating the number of stones in each pile.

The following M lines describe all the operations. Each line describes one operation, and it is formatted in one of the two following formats, depends on the kind it is:

- "1 l r x"— Add x stones to each pile in the interval $A_l, A_{l+1}, \ldots, A_r$.
- "2 l r"— Query if there is a way to choose of piles to guarantee Diana's win if they play game on interval $A_l, A_{l+1}, \ldots, A_r$.

It's guaranteed that $1 \le l \le r \le N$,$1 \le x \le 10^4$.

Output

For each query, if there is a way to choose of piles to guarantee Diana's win, output "Yes"; otherwise, output "No"(no quotes).

Examples

standard input	standard output
5 2 1 2 3 4 5 2 2 5 2 2 4	Yes No
5 4 1 2 4 8 16 2 1 5 1 1 1 1 1 2 3 2 2 1 5	No Yes

Problem I. I

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	1024 mebibytes

Let us regard an integer k as "similar to 250"if the following condition is satisfied:

- k is represented as $k = p \times q^3$ with primes $p < q$.

How many integers less than or equal to N are "similar to 250"?

Constraints

N is an integer between 1 and 10^{18}(inclusive)

Input

Input is given from Standard Input in the following format:

N

Output

Print the answer as an integer.

Examples

standard input	standard output
250	2
1	0
123456789012345	226863

Explanations

In the first test case:

- $54 = 2 \times 3^3$ is "similar to 250".
- $250 = 2 \times 5^3$ is "similar to 250".

The two integers above are all the integers "similar to 250".

Problem J. J

Input file:	`standard input`
Output file:	`standard output`
Time limit:	4 seconds
Memory limit:	1024 mebibytes

Takahashi has N pieces of chocolate. The i-th piece has a rectangular shape with a width of A_i centimeters and a length of B_i centimeters.

He also has M boxes. The i-th box has a rectangular shape with a width of C_i centimeters and a length of D_i centimeters.

Determine whether it is possible to put the N pieces of chocolate in the boxes under the conditions below.

- A box can contain at most one piece of chocolate.
- $A_i \le C_j$ and $B_i \le D_j$ must hold when putting the i-th piece of chocolate in the j-th box (they cannot be rotated).

Constraints

- $1 \le N \le M \le 2 \times 10^5$
- $1 \le A_i, B_i, C_i, D_i \le 10^9$
- All values in input are integers.

Input

Input is given from Standard Input in the following format:

N M
A_1 ... A_N
B_1 ... B_N
C_1 ... C_M
D_1 ... D_M

Output

If it is possible to put the N pieces of chocolate in the boxes, print "Yes"; otherwise, print "No".

Examples

standard input	standard output
2 3 2 4 3 2 8 1 5 2 10 5	Yes
2 2 1 1 2 2 100 1 100 1	No
1 1 10 100 100 10	No

standard input	standard output
1 1 10 100 10 100	Yes

Explanations

In the first test case,we can put the first piece of chocolate in the third box and the second piece in the first box.

In the second test case,a box can contain at most one piece of chocolate..

Problem K. K

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	1024 mebibytes

We have a simple directed graph G with N vertices and M edges. The vertices are labeled as Vertex 1, Vertex 2, ..., Vertex N. The i-th edge ($1 \le i \le M$) goes from Vertex U_i to Vertex V_i.

Takahashi will start at a vertex and repeatedly travel on G from one vertex to another along a directed edge. How many vertices of G have the following condition: Takahashi can start at that vertex and continue traveling indefinitely by carefully choosing the path?

Constraints

- $1 \le N \le 2 \times 10^5$
- $0 \le M \le \min(N(N-1), 2 \times 10^5)$
- $1 \le U_i, V_i \le N$
- $U_i \ne V_i$
- $(U_i, V_i) \ne (U_j, V_j)$ if $i \ne j$.
- All values in input are integers.

Input

Input is given from Standard Input in the following format:

```
N M
U_1 V_1
U_2 V_2
.
.
.
U_M V_M
```

Output

Print the answer.

Examples

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

Explanations

In the first test case.When starting at Vertex 2, Takahashi can continue traveling indefinitely: $2 \to 3 \to 4 \to 2 \to 3 \to \ldots$ The same goes when starting at Vertex 3 or Vertex 4. From Vertex 1, he can first go to Vertex 2 and then continue traveling indefinitely again. On the other hand, from Vertex 5, he cannot move at all. Thus, four vertices —Vertex 1, 2, 3, and 4— satisfy the conditions, so 4 should be printed.

In the second test case.Note that, in a simple directed graph, there may be two edges in opposite directions between the same pair of vertices. Additionally, G may not be connected.

Problem L. L

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

An array of positive integers a_1,a_2,. . . ,a_n is given. Let us consider its arbitrary subarray $a_l, a_{l+1}, \ldots, a_r$, where $1 \le l \le r \le n$. For every positive integer s denote by K_s the number of occurrences of s into the subarray. We call the power of the subarray the sum of products $K_s \cdot K_s \cdot s$ for every positive integer s. The sum contains only finite number of nonzero summands as the number of different values in the array is indeed finite.

You should calculate the power of t given subarrays.

Input

First line contains two integers n and t ($1 \le n, t \le 200000$) — the array length and the number of queries correspondingly.

Second line contains n positive integers a_i ($1 \le a_i \le 10^6$) — the elements of the array.

Next t lines contain two positive integers l, r ($1 \le l \le r \le n$) each — the indices of the left and the right ends of the corresponding subarray.

Output

Output t lines, the i-th line of the output should contain single positive integer — the power of the i-th query subarray.

Please, do not use $\%lld$ specificator to read or write 64-bit integers in C++. It is preferred to use *cout* stream (also you may use $\%I64d$).

Examples

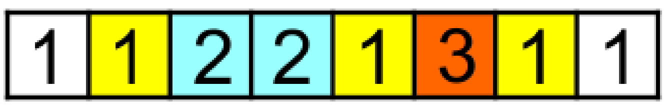
standard input	standard output

`3 2`
`1 2 1`
`1 2`
`1 3`
	`3`
	`6`

`8 3`
`1 1 2 2 1 3 1 1`
`2 7`
`1 6`
`2 7`
	`20`
	`20`
	`20`

Explanations

Consider the following array (see the second sample) and its $[2, 7]$ subarray (elements of the subarray are colored):



Then $K_1 = 3$, $K_2 = 2$, $K_3 = 1$, so the power is equal to $3^2 \cdot 1 + 2^2 \cdot 2 + 1^2 \cdot 3$ =20.