

The 2022 Hangzhou Normal U Summer Trials

Prepared by Hangzhou Normal U ACM/ICPC Team

May, 14, 2022

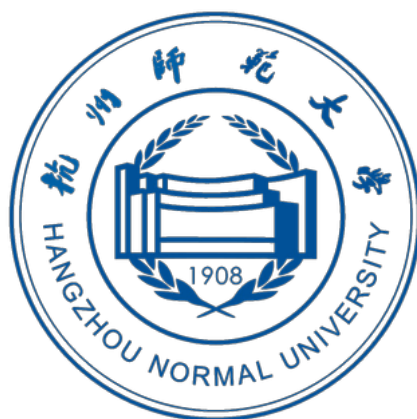


杭州师范大学
HANGZHOU NORMAL UNIVERSITY

Problems

A	Hello, ACMer!
B	New String
C	Check Problems
D	Tree Problem
E	Easy Problem
F	Subarrays
G	Ganyu Segment Tree
H	Optimal Biking Strategy
I	IHI's Homework
J	IHI's Magic String
K	Klee's Wonderful Adventure

Do not open before the contest has started.



Problem A. Hello, ACMer!

Input file: `standard input`
Output file: `standard output`
Time limit: `1 second`
Memory limit: `256 megabytes`

Today, we are happy to invite so many good friends. IHI has just written a long string to celebrate this great party. How many occurrences of "hznu" as a continuous substring are there in the long string? Please write a program to count the number.

Input

The only line contains a non-empty string, consisting of no more than 100 000 lowercase Latin letters, 'a' to 'z'.

Output

Output a line containing a single integer, indicating the number of occurrences of "hznu" in the given string.

Examples

<code>standard input</code>	<code>standard output</code>
hznu	1
hhzznnuu	0

Problem B. New String

Input file: standard input
Output file: standard output
Time limit: 1 second
Memory limit: 64 megabytes

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
zabcdefghijklmnopqrstuvwxyz 2 a b 1	a

Problem C. Check Problems

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

Now, n people are checking the problems. There are $10^{10^{10}}$ problems in total and numbered from 1 to $10^{10^{10}}$. The i -th person starts the check from the a_i problem. It takes one minute to check a problem. More formally, the problem with the number $a_i + j - 1$ will be checked by the i -th person at the j -th minute.

Now, your task is to answer q queries. Each query will give you an integer t , please calculate the number of problems that have been tested at least once after t minutes.

Input

The first line contains a single integer n ($1 \leq n \leq 5 \cdot 10^5$) indicating the number of people.

The second line contains n integers a_1, a_2, \dots, a_n ($1 \leq a_1 \leq a_2 \leq \dots \leq a_n \leq 10^{18}$) indicating the problem number of the first check for the i -th person.

The next line contains a single integer q ($1 \leq q \leq 5 \cdot 10^5$) indicating the number of queries.

Each of the next q lines contains an integer t ($0 \leq t \leq 10^{18}$).

it is guaranteed that $a_{i+1} - a_i \leq a_{i+2} - a_{i+1}$ for every $1 \leq i \leq n - 2$.

Output

For each query, output the total number of problems that have been tested at least once after t minutes in one line.

Example

standard input	standard output
3	6
1 3 8	10
3	17
2	
4	
10	

Note

After two minutes, the first person checked $[1, 2]$ problems, the second person checked $[3, 4]$ problems and the third person checked $[8, 9]$ problems, the answer is 6.

After four minutes, the first person checked $[1, 2, 3, 4]$, the second person checked $[3, 4, 5, 6]$, and the third person checked $[8, 9, 10, 11]$, the answer is 10.

Problem D. Tree Problem

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

You are given a tree consisting of n vertices. Recall that a tree is a connected graph consisting of n vertices and $(n - 1)$ undirected edges.

Now, your task is to answer q queries. Each query will give you a vertex x , please calculate the number of simple paths of length **at least one** through the vertex x . Note that paths that differ only by their direction are considered the same (i. e. you have to calculate the number of **undirected** paths). For example, paths $[1, 2, 3]$ and $[3, 2, 1]$ are considered the same.

Recall that a simple path is a path that visits each vertex at most once.

Input

The first line contains an integer n ($2 \leq n \leq 10^5$) indicating the number of vertices.

For the next $(n - 1)$ lines, the i -th line contains two integers u_i and v_i ($1 \leq u_i, v_i \leq n$) indicating an edge connecting vertices u_i and v_i in the tree.

The first line contains an integer q ($1 \leq q \leq 10^5$) indicating the number of queries.

For the next q lines, the i -th line contains an integer x_i ($1 \leq x_i \leq n$) indicating the vertex as described above.

Output

For each query output one line containing one integer indicating the answer.

Example

standard input	standard output
5	7
1 2	4
1 3	9
3 4	4
3 5	4
5	
1	
2	
3	
4	
5	

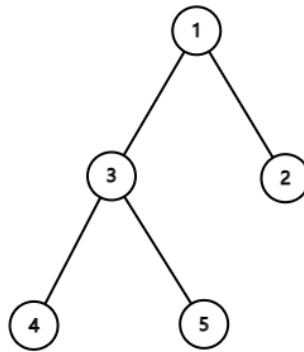
Note

The example looks like that:

There are 7 different simple paths through vertex 1:

- $[1, 2]$;
- $[1, 3]$;

- $[2, 1, 3]$;
- $[1, 3, 4]$;
- $[1, 3, 5]$;
- $[2, 1, 3, 4]$;
- $[2, 1, 3, 5]$.



Problem E. Easy Problem

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

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 ($2 \leq n \leq 50$).

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
7 ..a.b..	4

Problem F. Subarrays

Input file: `standard input`
Output file: `standard output`
Time limit: 1 second
Memory limit: 256 megabytes

Give you an array of n positive integers, your task is to count the number of subarrays whose sum is a multiple of k .

A subarray of array is a sequence a_l, a_{l+1}, \dots, a_r for some integers (l, r) such that $1 \leq l \leq r \leq n$.

Input

The first line contains two positive integers n ($1 \leq n \leq 10^5$), k ($1 \leq k \leq 10^9$).

The second line contains n positive integers a_1, a_2, \dots, a_n ($0 \leq a_i \leq 10^9$).

Output

Output one line containing one integer indicating the number of subarrays.

Example

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

Note

For the test

$sum[1, 1] = 0$, 0 is a multiple of 3

$sum[1, 3] = 0 + 1 + 2 = 3$, 3 is a multiple of 3

$sum[1, 6] = 0 + 1 + 2 + 4 + 7 + 7 = 21$, 21 is a multiple of 3

$sum[2, 3] = 1 + 2 = 3$, 3 is a multiple of 3

$sum[2, 6] = 1 + 2 + 4 + 7 + 7 = 21$, 21 is a multiple of 3

$sum[3, 4] = 2 + 4 = 6$, 6 is a multiple of 3

$sum[3, 6] = 4 + 7 + 7 = 18$, 18 is a multiple of 3

So the ans is 7.

Problem G. Ganyu Segment Tree

Input file: **standard input**
Output file: **standard output**
Time limit: **2 seconds**
Memory limit: **512 megabytes**

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, \dots, 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 $(a_l, a_{l+1}, \dots, a_r)$, replace unlocked a_i with $a_i \times x$.
- div $l\ r\ x$: for every element a_i in the subarray $(a_l, a_{l+1}, \dots, a_r)$, 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 ($1 \leq n, m \leq 10^5$) 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 $1 \leq p \leq n$, $1 \leq l \leq r \leq n$ and $1 \leq x \leq 30$.

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).

Example

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	

Note

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 divisible by 9
- [18, 3, 3, 18, 18], a_2 is locked
- [18, 3, 3, 18, 18], a_2 is not divisible by 9
- [18, 3, 3, 18, 18], a_2 is locked so you can't change its value
- [18, 3, 3, 18, 18], a_2 is not divisible by 9
- [18, 3, 1, 18, 18], a_2 and a_3 are divisible 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 H. Optimal Biking Strategy

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

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 ($1 \leq n \leq 10^6, 1 \leq p \leq 10^9, 1 \leq s \leq 10^9$) 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, \dots, a_n ($0 \leq a_i \leq p$) indicating the position of the i -th stop ($\forall i < j, a_i < a_j$).

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

Output

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

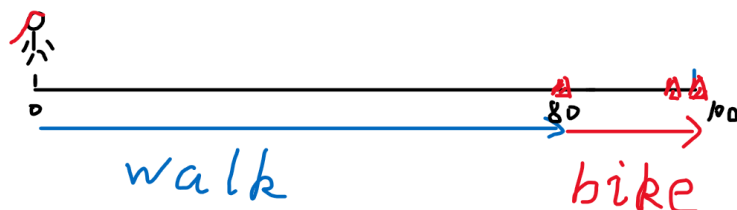
Examples

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

Note

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 I. IHI's Homework

Input file: `standard input`
Output file: `standard output`
Time limit: 1 second
Memory limit: 256 megabytes

IHI's assignment today is to solve inequality. His task is to calculate the number of solutions of the inequality $x_1 + x_2 + \dots + x_n \leq s$.

To exercise IHI's ability to draw inferences from one another, the teacher gave each x_i a limit. Each x_i should be greater than or equal to a_i . In other words, $\forall x_i \geq a_i (1 \leq i \leq n)$.

The teacher assigned a total of q problems. For each of them, you are given two integers x and val . You need to change the value of a_x to val . In other words, each times, $a_x = val$. Then calculate the answer. Since this number may be very large, the teacher asks IHI to find the number modulo $(10^9 + 7)$.

Can you help IHI?

Input

The first line of input contains three space-separated integers n , s and q ($1 \leq n \leq 2 \times 10^5$, $0 \leq s \leq 2 \times 10^5$, $1 \leq q \leq 2 \times 10^5$) — the length of the array a , the right value of the inequality, and the number of operations.

The second line contains n space-separated integers a_1, a_2, \dots, a_n ($0 \leq a_i \leq 2 \times 10^5$)

The next q lines contains two integers x and val ($1 \leq x \leq n$, $0 \leq val \leq 2 \times 10^5$) — You need to change a_x to val .

Note that each operation permanently changes array a . In other words, a_x will be forever changed.

Output

Print q integers. The i -th line is the answer to this problem after the i -th operation.

Example

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

Note

After the first operation, $a=[1, 1, 1]$. The value of $\{x_1, x_2, x_3\}$ can be $\{1, 1, 1\}$, $\{1, 1, 2\}$, $\{1, 2, 1\}$, $\{2, 1, 1\}$, $\{2, 2, 1\}$, $\{2, 1, 2\}$, $\{1, 2, 2\}$, $\{3, 1, 1\}$, $\{1, 3, 1\}$, $\{1, 1, 3\}$. So there are 10 solutions, output 10.

After the third operation, $a = [2, 2, 1]$. There's only one set of solution satisfies the inequality. — The value of $\{x_1, x_2, x_3\} = \{2, 2, 1\}$.

After the fourth operation, $a = [2, 2, 2]$. No set of solutions satisfies the inequality, so the output is 0.

Problem J. IHI's Magic String

Input file: `standard input`
Output file: `standard output`
Time limit: 1 second
Memory limit: 256 megabytes

IHI has recently acquired a magic power that can perform the following operations on strings:

1. puts a character at the end of a string.
2. deletes a character at the end of a string (does not operate if the string is empty).
3. replaces all current character x in the string to characters y .

All characters are lowercase.

Now IHI has a string that starts empty, he wants to know what the final string is after q operations.

Input

The first line contains a number q ($1 \leq q \leq 10^5$) indicating the number of operations.

The next q lines represent q operations.

There are three types of operations:

- 1 x : In the first operation, you should put character x at the end of the string.
- 2: In the second operation, you should delete a character at the end of the string if and only if this string is not empty.
- 3 x y : In the third operation, replace x in the string with y .

Output

If the final string is empty, you should output "The final string is empty".

Otherwise, output the final string in one line.

Examples

standard input	standard output
5 1 a 1 b 1 c 3 a c 1 b	cbcb
6 1 a 1 b 1 c 1 c 3 a c 3 c a	abaa
5 1 a 1 c 3 a c 2 2	The final string is empty

Problem K. Klee's Wonderful Adventure

Input file: standard input
Output file: standard output
Time limit: 1 second
Memory limit: 512 megabytes

Today little Klee wanted to go on an adventure outside. Klee wanted to go to so many places to play, but finally, she could only go to one place to play and she can pass away any number of times in other places. First of all, let's consider the plane where Klee is located as a two-dimensional plane, which is divided into four quadrants, and there are n points in the whole plane. Klee can only move directly from one point to another for each movement.

Since each quadrant represents a different country, the maximum speed that can be achieved differs for each quadrant. The ranges and maximum speeds of these four quadrants are defined as follows:

quadrant	x	y	maximum speed
1	$x \geq 1$	$y \geq 1$	v_1
2	$x \leq -1$	$y \geq 1$	v_2
3	$x \leq -1$	$y \leq -1$	v_3
4	$x \geq 1$	$y \leq -1$	v_4

If two points are in the same quadrant, Klee can reach the maximum speed given in that quadrant. For example, in one movement, if Klee's start point and Klee's endpoint are both located in the first quadrant, the maximum speed is v_1 .

If the two points are not in the same quadrant, then Klee's speed must not be greater than v_0 to transfer directly in both quadrants. For example, in one movement, if Klee's start point is located in the first quadrant and Klee's endpoint is located in the second quadrant, the maximum speed is v_0 in this movement.

Now, given the initial index of the point that Klee is in and the final index of the point that Klee finally wants to play, please find the minimum time needed for Klee to get from the initial point to the final point.

Input

The First line is an integer n ($1 \leq n \leq 3 \cdot 10^3$) indicating the number of points.

The second line contains 5 integers v_1, v_2, v_3, v_4, v_0 ($1 \leq v_1, v_2, v_3, v_4, v_0 \leq 10^5$) as described above.

The third line contains two integers s, t ($1 \leq s, t \leq n$), indicating the initial index of the point and the final index of the point.

For the following n lines, the i -th line contains two integers x_i, y_i ($-10^5 \leq x_i, y_i \leq 10^5, x_i \neq 0, y_i \neq 0$), indicating the coordinate of the i -th point.

We ensure that there is no overlap between the two points(which means for any point i and j , we can ensure that $x_i = x_j$ and $y_i = y_j$ do not occur simultaneously).

Output

Output one line containing one integer indicating the minimum time needed for Klee to get from the initial point to the final point.

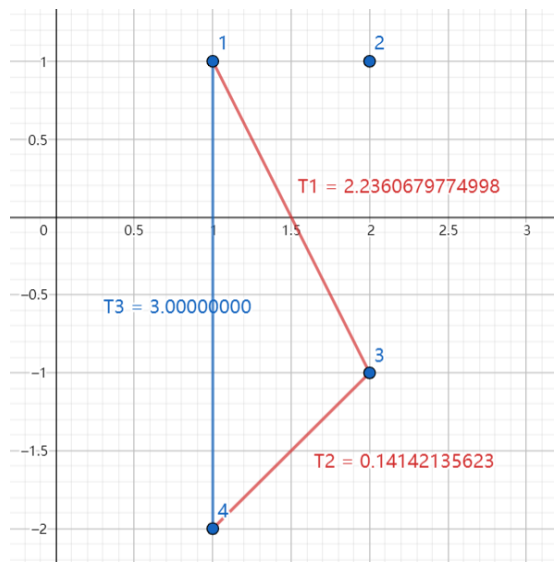
Your answer will be accepted if the absolute or relative error does not exceed 10^{-6} . Formally, let your answer be a , and the jury's answer is b . Your answer is considered correct if $\frac{|a-b|}{\max(1, |b|)} \leq 10^{-6}$.

Examples

standard input	standard output
3 1 2 3 4 5 1 3 1 -5 1 -1 1 1	1.2000000000
4 5 1 1 10 1 1 4 1 1 2 1 2 -1 1 -2	2.3414213562
3 1 1 1 1 100 1 3 1 1 2 -1 1 -100	1.0100000000

Note

In the second test case:



Klee needs to move from point 1 to point 4.

She will first move from point 1 to point 3 with a speed of v_0 , and then from point 3 to point 4 at a speed of v_4 .