

TASK	NIZOVI	PROZOR	OZLJEDA	OTPOR	PROSTI	PROKLETNIK
input	standard input ( <i>stdin</i> )					
output	standard output ( <i>stdout</i> )					
time limit	1 second	1 second	1 second	1 second	0.5 seconds	4 seconds
memory limit	64 MB	64 MB	64 MB	64 MB	64 MB	128 MB
score	<b>50</b>	<b>80</b>	<b>100</b>	<b>120</b>	<b>140</b>	<b>160</b>
	<b>total 650</b>					

An **array** in a certain programming language is defined in the following way:

- The beginning and the end of an array is denoted by an open and closed curly bracket, respectively.
- Inside the curly braces, there are (possibly even 0) **values** separated by a comma (there is no comma after the last value in the array).
- Each value can be a **word** (an array of lowercase letters of the English alphabet) or another array.
- Examples of correctly defined arrays are: {}, {a,b,c}, {abc,znj},{novi,niz},pozz}.

Recently, you've noticed that the programming language does not throw an error if you do not place the appropriate number of spaces and new lines before or after the curly braces or the commas. Given the fact that you too mind the values being so „squished“ together, you've decided to get to work and modify the shape of the array in the following way:

- Each value that is not an array or denotes the beginning or the end of the array (curly braces) will be in its own **line**.
- The commas are „connected“ with the value located directly before them and there is a new line after each comma.
- After the curly bracket denoting the beginning of the array, the indentation of the content **increases** (shifting the output to the right) for **2 spaces**.
- Before the curly bracket denoting the end of the array '}', the indentation of the content **decreases** (shifting the output to the left) for **2 spaces**.

#### INPUT

The first line of input contains an array of characters  $S$  ( $1 \leq |S| \leq 1500$ ), representing the array from the task.

#### OUTPUT

The output must consist of the modified version of the array from the task.

#### SCORING

In test cases worth 50% of total points, all values in the array are going to be words (in other words, it won't be the case that a value is another array).

SAMPLE TESTS

<pre>input {abc,ono,sto}  output {   abc,   ono,   sto }</pre>	<pre>input {}  output { }</pre>	<pre>input {znakovi}  output {   znakovi }</pre>	<pre>input {a,b,{c,d},e,{} }  output {   a,   b,   {     c,     d   },   e,   {   } }</pre>
--	---------------------------------	--	---

**Clarification of the fourth example:** In the beginning, there is no indentation, it is 0. After the first curly bracket, there is a new line and the indentation increases by 2 spaces.

After that, the word „a“ is printed, as well as the comma right after it, then the new line with the same indentation of 2 spaces. The same procedure is repeated for the word „b“.

The third value in the array is a new array (let's call it array X). It begins with a curly bracket, so we need to print a new line and increase the indentation for 2 spaces. The indentation is now a total of 4 spaces. Using that indentation, we print the words „c“ and „d“ each in its own line. After the word „d“, there is no comma because it is the last value in the array X.

Before we print the curly bracket that denotes the end of array X, we need to decrease the indentation for 2. After the curly bracket, we print the comma and a new line and the continue printing the values from the main array.

Young Marin, now already a respectable star of Croatian cinematography, spends his free time looking out the window. Unfortunately for him, the same activity is also taken up by a few flies that have landed on his window and are looking into the distance. Marin immediately hurried to get his **square** racket for killing flies. At the same time, he wondered how many flies it is possible to kill in one shot. After trying to scare away the flies unsuccessfully for a few hours, he decided to ask for your help.

Marin has sent you a picture of his window, the dimensions being  $R \times S$  pixels and has also told you the length of the side of the racket he uses to kill the flies, in pixels. Your task is to determine the maximal amount of flies that Marin could have killed in a single shot and mark one such shot in the picture.

### INPUT

The first line of input contains three integers  $R$ ,  $S$  and  $K$  ( $3 \leq K \leq R, S \leq 100$ ) that denote the dimensions of Marin's picture and the length of the side of the racket.

Each of the following  $R$  lines contains  $S$  characters that describe Marin's picture. The picture's pixels marked with '\*' denote the position of a fly, whereas all the other pixels are marked with '.' and denote empty space. On his window, there is at least one fly that Marin can kill with his racket.

### OUTPUT

The first line of output must contain the maximal number of flies Marin can kill in a single shot.

The following  $R$  lines must contain Marin's picture, on it clearly marked a position of the racket that will ensure Marin kills as many flies as possible. The horizontal sides of the racket are denoted with a series of characters '-' and the vertical ones with '|', whereas angles are denoted with '+'. For a more detailed explanation, consult the sample tests.

**Please note:** Marin's racket will affect only the flies located strictly inside the racket, and the racket must be located inside the window with all its parts. In other words, it is assumed that the flies located at the sides of the racket will have enough time to fly away.

### SCORING

In test cases worth 50% of all points, it is sufficient to only have the first line of output correct.

### SAMPLE TESTS

<p><b>input</b></p> <pre>3 5 3 ..... .*.*. .....</pre> <p><b>output</b></p> <pre>1 +-+..  * *. +-+..</pre>	<p><b>input</b></p> <pre>7 6 4 ..... .*.*.* ..... .*.*.. ..*... ..*... *.....*</pre> <p><b>output</b></p> <pre>2 ..... .*.*.* +--+..  * .  . * . +--+.. *.....*</pre>	<p><b>input</b></p> <pre>9 9 6 ***..... .....*.* .*.....* ..*.....* ..*.*.... ..*.....* .....*... .*.....** .....</pre> <p><b>output</b></p> <pre>6 ***..... .....*.* .*.....* ..*+----+ ..* *...  ..* ...*  ... . *..  .* .***  ...+----+</pre>
--	---	--

Due to the frantical usage of the racket to kill flies, Marin has sustained a serious bodily injury known to the medical community as *epicondylitis lateralis humeri*. His grandma has advised smearing rakija over it, the doctor has prescribed a strong painkiller, but Marin has ignored every single advice and decided to look for the answer in integer sequences.

He has discovered a previously undiscovered sequence of integers and called it the *xorbonacci* sequence. The  $n^{\text{th}}$  element in the sequence is denoted with  $x_n$ . The sequence is defined recursively in the following way:

$$\begin{aligned}x_1 &= a_1, \\x_2 &= a_2, \\&\dots \\x_k &= a_k, \\x_n &= x_{n-1} \oplus x_{n-2} \oplus \dots \oplus x_{n-k}, \quad n > k\end{aligned}$$

Because of a reason only known to Marin, he determined that all his sorrows will go away if you answer his  $Q$  queries defined with numbers  $l$  and  $r$ . The answer to the query is represented with the value

$$x_l \oplus x_{l+1} \oplus \dots \oplus x_{r-1} \oplus x_r$$

Help Marin and answer his queries.

**Please note:** The operation  $\oplus$  is the operation of binary **XOR**.

### INPUT

The first line of input contains the integer  $K$  ( $1 \leq K \leq 100\,000$ ) from the task.

The following line contains  $K$  integers that represent the first  $K$  elements in the xorbonacci sequence. All numbers are smaller than  $10^{18}$ .

The following line contains the integer  $Q$  ( $1 \leq Q \leq 10^6$ ) from the task.

The  $i^{\text{th}}$  of the following  $Q$  lines contains two integers  $l_i$  and  $r_i$  ( $1 \leq l_i \leq r_i \leq 10^{18}$ ) that represent Marin's  $i^{\text{th}}$  query.

### OUTPUT

Each of the following  $Q$  lines of output must contain the answers to Marin's queries, the order being the same as the input.

### SAMPLE TESTS

<b>input</b> 4 1 3 5 7 3 2 2 2 5 1 5	<b>input</b> 5 3 3 4 3 2 4 1 2 1 3 5 6 7 9
<b>output</b> 3 1 0	<b>output</b> 0 4 7 4

Mirko has been a very good boy, so he got exactly what he wanted for his birthday, a "Young physicist" kit! In the kit, there are  $N$  types of resistors, connecting wires and an ohmmeter. If a resistor is of type  $i$ , it provides a resistance of precisely  $R_i$  ohms.

As we all know, resistors can be connected in two different ways, in **series** and **parallel**. Also, we know that the complete circuit of resistors can be replaced with **one resistor** of the resistance equal to the **equivalent resistance of the circuit**. When the circuit is **series**, it is written in the following way:

$$(R1-R2-R3-...-RK)$$

and the following formula holds:

$$R_{ekv} = R_1 + R_2 + R_3 + \dots + R_K$$

When the circuit is **parallel**, it is written in the following way:

$$(R1|R2|R3|...|RK)$$

and the following formula holds:

$$R_{ekv} = \frac{1}{\frac{1}{R_1} + \frac{1}{R_2} + \frac{1}{R_3} + \dots + \frac{1}{R_K}}$$

Mirko was excited to bring his new toy to school, where he showed it to all his friends. Unfortunately for him, the teacher also noticed the toy. She quickly connected a circuit and is now asking Mirko to tell her the equivalent resistance of the circuit. Help Mirko and answer the teacher's question!

#### INPUT

The first line of input contains the integer  $N$  ( $1 \leq N \leq 9$ ) from the task.

The following line contains  $N$  **real** numbers  $R_i$  ( $0 < R_i < 100$ ) from the task.

The last line contains the circuit  $S$ , that will not be longer than 100 000 characters.  $S$  will only consist of the following characters: 'R', '1' - '9', '(', ')', '-' and '|'. The number of pairs of brackets will be minimal, and the brackets will be paired properly. Inside of a pair of brackets, there will not be the character '-' and the character '|' at the same time. The circuit will only consist of the resistors from the kit and it is possible that not all types of resistors are used, or that one type of resistor is used several times.

#### OUTPUT

The first and only line of output must contain the number from the task. A deviation from the official solution of  $\pm 0.00001$  is tolerated.

#### SCORING

In test cases worth 20% of total points, it will hold:

- $N = 3$
- $S$  is going to consist of exactly three resistors,  $R_1$ ,  $R_2$  and  $R_3$ , not necessarily in that order

In test cases worth 40% of total points,  $S$  is going to have exactly one pair of brackets.

#### SAMPLE TESTS

<p><b>input</b></p> <p>3 2 3 6 (R1-R2-R3)</p> <p><b>output</b></p> <p>11.00000</p>	<p><b>input</b></p> <p>3 2 3 6 (R1 R2 R3)</p> <p><b>output</b></p> <p>1.00000</p>	<p><b>input</b></p> <p>5 5.0 5.0 5.0 5.0 5.0 ((R1-(R2 R3)-(R4 R5))  (R1-(R2 R3)-(R4 R5)))</p> <p><b>output</b></p> <p>5.00000</p>
--	---	---

**Clarification of the third example:** The circuit from the input is broken in two rows in order to maximize readability.

Mirko and his older brother Slavko are playing a game. At the beginning of the game, they pick three numbers  $K, L, M$ . In the first and only step of the game, each of them picks their own  $K$  consecutive integers.

Slavko always picks the first  $K$  integers (numbers 1, 2, ...,  $K$ ). Mirko has a special demand – he wants to choose his numbers in a way that there are exactly  $L$  happy numbers among them. He considers a number happy if it meets at least one of the following requirements:

- the number is smaller than or equal to  $M$
- the number is **prime**

Out of respect to his older brother,  $L$  will be **smaller than or equal to** the total number of happy numbers in Slavko's array of numbers.

They will play a total of  $Q$  games with different values  $K, L, M$ . For each game, help Mirko find an array that meets his demand.

#### INPUT

The first line of input contains  $Q$  ( $1 \leq Q \leq 100\,000$ ). Each of the following  $Q$  lines contains three integers, the  $i^{\text{th}}$  line containing integers  $K_i, L_i, M_i$  ( $1 \leq K_i, M_i \leq 150, 0 \leq L_i \leq K_i$ ) that determine the values  $K, L, M$  that will be used in the  $i^{\text{th}}$  game.

#### OUTPUT

Output  $Q$  lines, the  $i^{\text{th}}$  line containing an integer, the initial number of Mirko's array in the  $i^{\text{th}}$  game. If an array with the initial number being smaller than or equal to 10 000 000 does not exist, output  $-1$ . If there are multiple possible solutions, output any.

#### SAMPLE TESTS

<b>input</b> 3 1 1 1 2 0 2 3 1 1	<b>input</b> 3 4 1 1 5 2 3 5 0 3	<b>input</b> 4 7 2 5 6 1 1 10 4 5 6 2 2
<b>output</b> 1 8 4	<b>output</b> 6 4 24	<b>output</b> 6 20 5 4

Young Luka is about to enter a house with the evil witch Marica inside. As soon as he enters the house, she asks him questions about her array of  $N$  numbers. Luka fearfully asks for a clarification of the questions. Marica explains to him that each query consists of two integers  $L$  and  $R$  which represent the positions of a contiguous sub-array in her array.

It is Luka's task to answer for each query what the longest **contiguous** sub-array of that contiguous sub-array (it can be the entire sub-array) having the property of being magical. An array is called magical if all the values are between the values of the first and last number in that array. For example, [1 3 1 2 4] is magical, the same as [4 1 1 2 1], whereas [3 3 4 1] is not magical.

### INPUT

The first line of input contains the integer  $N$  ( $1 \leq N \leq 500\,000$ ), the number of numbers in the array. The second line contains  $N$  integers  $a_i$  ( $1 \leq a_i \leq 10^9$ ).

The third line contains the integer  $Q$  ( $1 \leq Q \leq 500\,000$ ), the number of queries.

Each of the following  $Q$  lines contains two integers,  $L$  and  $R$  ( $1 \leq L \leq R \leq N$ ), representing the sub-array from the query.

### OUTPUT

The  $i^{\text{th}}$  line of output must contain a single integer – the answer to the  $i^{\text{th}}$  query.

### SCORING

In test cases worth 50% of total points, it will hold  $N, Q \leq 30\,000$ .

### SAMPLE TESTS

<b>input</b> 5 5 4 3 3 2 3 1 2 1 1 2 4	<b>input</b> 6 6 6 5 1 6 2 3 4 5 4 6 1 4
<b>output</b> 2 1 3	<b>output</b> 2 2 4