SQRT Contest #02 - A - Alice's game

Time limit: 1.0s **Memory limit:** 256M

Alice has n stones. She wants to play a game with Bob as follows:

- First, Bob must divide the n stones into m piles such that each pile contains at least one stone.
- Then, they play a game where Alice goes first:
  - On their turn, a player chooses one pile (that contains at least one stone) and removes any number of stones from it (they can take all the stones but cannot take none).
  - A player loses if they cannot make a move on their turn.

Help Bob find a way to divide the stones so that he can always guarantee a win if he plays optimally.

Input

- The first line contains an integer T, the number of test cases $(1 \leq T \leq 1000)$.
- The next T lines each contain two positive integers n and m, where:
  - $1 \leq n \leq 10^9$, $1 \leq m \leq 10^5$, and $n \geq m$.
- The total sum of all m across all test cases does not exceed 2×10^5.

Output

For each test case:

- Print `NO` if there is no way for Bob to guarantee a win.
- Print `YES` followed by m positive integers $a_1, a_2, \ldots, a_m$ (the number of stones in each pile) if there is a way for Bob to guarantee a win.

Example

Input

```
3
3 1
4 2
5 5
```

Output

```
NO
YES 2 2
NO
```

Explanation

- In the first test case, the only possible division is to put all 3 stones into one pile. Alice can take all 3 stones on her first turn to ensure a win.
- In the second test case, if Bob divides the stones into two piles, each with 2 stones, he can always mirror Alice's moves (for example, if Alice takes 1 stone from pile 1, Bob will take 1 stone from pile 2), ensuring his victory.
- In the third test case, the only possible division is to put 5 stones into 5 separate piles. In this case, Bob cannot win, as Alice can always take one stone from a pile and Bob will have no moves left.

SQRT Contest #02 - B - Beautiful pairs

Time limit: 0.5s **Memory limit:** 256M

A pair of numbers (a, b) has a beauty value calculated by the formula $\lfloor \frac{a}{b} \rfloor + a \mod b$, where $\lfloor a \rfloor$ is the greatest integer less than or equal to a. Given a number k, find a pair (a, b) with $2 \leq a, b \leq 10^9$ such that the beauty value equals k.

Input

- The first line contains a positive integer T which is the number of test cases $(1 \leq T \leq 1000)$.
- Each of the following T lines contains a positive integer k $(1 \leq k \leq 10^9)$.

Output

For each test case, print two positive integers a, b that satisfy the condition. If no such pair exists, print `-1 -1`.

Example

Input

```
3
1
2
3
```

Output

```
5 5
3 2
5 3
```

Explanation

- In the first test case, $\lfloor \frac{5}{5} \rfloor + 5 \mod 5 = 1 + 0 = 1$.
- In the second test case, $\lfloor \frac{3}{2} \rfloor + 3 \mod 2 = 1 + 1 = 2$.
- In the third test case, $\lfloor \frac{5}{3} \rfloor + 5 \mod 3 = 1 + 2 = 3$.

SQRT Contest #02 - C - Card game

Time limit: 1.0s **Memory limit:** 256M

This is an interactive problem

Alice has n cards arranged in a stack, among which there is a special card located at position k from the top. Alice and Bob will play a game with Alice taking the first turn, following these rules:

- In each turn, a player can take up to r cards from either the top or the bottom of the stack.
- The player who picks the special card wins the game.

You are given the values of n, k, r. Determine who will win if both players play optimally. Then, act as that player and play the game against the judge. You will play T rounds for each testcase.

Interaction

- First, your program will read an integer T denoting the number of rounds to play $(1 \leq T \leq 10)$.
- Each round proceeds as follows:
  1. Read three integers n, k, r $(1 \leq k, r \leq n \leq 10^5)$.
  2. Print either `Alice` or `Bob` to indicate the character you want to play as.
  3. The game begins, with the first turn going to the judge (if you choose `Bob`) or to you (if you choose `Alice`).
     - On your turn, output:
       - $1\ x$ if you want to take x cards from the top.
       - $2\ x$ if you want to take x cards from the bottom.
     - On the judge's turn, read the move in the same format.
  4. Detect the game's end condition yourself. As soon as the game ends, move on to the next round or exit the program if all T rounds are complete or if you lose.

Example Interaction

Program	Judge	Explanation
	`1`	You need to play 1 round with the judge.
	`5 3 2`	$n = 5, k = 3, r = 2$
`Bob`		You choose to play as Bob. The game starts with Alice (the judge) taking the first turn.
	`1 2`	The judge (Alice) takes 2 cards from the top.
`2 1`		You take 1 card from the bottom.

Program	Judge	Explanation
	`2 1`	The judge takes 1 card from the bottom.
`1 1`		You take 1 card from the top, which is the special card. The game ends with you winning.

SQRT Contest #02 - D - Divisor sequence

Time limit: 1.0s **Memory limit:** 256M

Alice has just learned about divisors and multiples of integers in class. If an integer a is divisible by another integer b, we say that a is a multiple of b and b is a divisor of a.

To test the knowledge and calculation ability of the students, Alice's teacher gave the students a long sequence of numbers $a_1, a_2, \ldots, a_n$ and asked them to select as many elements as possible so that when they combine them into a sequence while keeping the order of the numbers in the original sequence, denoted as $b_1, b_2, \ldots, b_k$, for all integers $2 \leq i \leq k$, b_i is either a divisor or a multiple of b_{i-1}. The student who selects the longest sequence (i.e., the largest k) will receive a prize from the teacher.

Alice really wants to win this game. Help Alice find the longest sequence b.

Input

- The first line contains a positive integer n ($1 \leq n \leq 10^5$).
- The second line contains n positive integers $a_1, a_2, \ldots, a_n$ ($1 \leq a_i \leq 10^5$).

Output

- The first line should contain the positive integer k — the length of the longest sequence b that you can find.
- The second line should contain n integers $x_1, x_2, \ldots, x_k$, with $x_1 < x_2 < \ldots < x_k$, which represent the indices of the elements of the sequence b in the original sequence a, i.e., $b_i = a_{x_i}$. If there are multiple results, print any of them.

Example

Input

```
5
1 2 3 4 5
```

Output

```
3
1 2 4
```

Explanation

- We have $b = (a_1, a_2, a_4) = (1, 2, 4)$, where 2 is a multiple of 1 and 4 is a multiple of 2. It is easy to see that this is the longest sequence b that satisfies the condition.

SQRT Contest #02 - E - Encode and decode

Time limit: 2.0s **Memory limit:** 256M

You are likely familiar with the Caesar cipher, one of the simplest and most well-known encryption techniques. In this cipher, each character in the plaintext is replaced by another character that is a fixed number of positions `k` away in the alphabet. For example, with `k = 3`, `a` is replaced with `d`, `b` with `e`, `c` with `f`, ..., `y` with `b`, and `z` with `c`.

Although this technique is popular, it is very easy to break. To improve it, Alice has an idea: concatenate the plaintext and the encrypted text into a single string, and add some noise characters before the plaintext, between the plaintext and the encrypted text, or after the encrypted text. For example, starting with the plaintext `abc` and $k = 3$, she encrypts it to `def` and then adds some noise characters to form the string `xabcyzdeft`.

You are given the final string s obtained after this process. Determine the maximum possible length of the plaintext.

Input

- A single string s of length at most 2×10^5.

Output

- A single integer representing the maximum possible length of the plaintext.

Example

Input

```
sqrtonlinejudge
```

Output

```
2
```

Explanation

The longest possible plaintext is `sq`, which is encrypted using the Caesar cipher with $k = 21$ to form `nl`.

SQRT Contest #02 - F - Frog counting

Time limit: 2.0s **Memory limit:** 1G

On a lake, there are many lotus leaves arranged in a long row. One Sunday, Alice brought some frogs to the lakeside to study how they jump on these leaves.

Alice numbered the leaves in increasing order from left to right. She plans to conduct the experiment over q minutes. In each minute, Alice will perform one of the following two actions:

- Release a frog with a jump step of k on leaf number i. Each minute, the frog will jump k leaves to the right. In other words, if at minute t the frog is on leaf x, then at minute $t+1$, it will jump to leaf $x+k$.
- Count the number of frogs on the leaves $l, l+1, \ldots, r$.

Since Alice knows the jump steps of the frogs in advance, she wants to pre-calculate the results of the counting actions to compare with the actual results. Please help Alice with this task.

Input

- The first line contains a positive integer q $(1 \leq q \leq 10^5)$.
- The next q lines describe Alice's actions at minute i and are in one of the following two formats:
  - $1\ k\ i$: release a frog with a jump step of k on leaf i $(1 \leq i, k \leq 10^5)$.
  - $2\ l\ r$: count the number of frogs on the leaves in the range $[l, r]$ $(1 \leq l \leq r \leq 10^5)$.

Output

- Print non-negative integers, each being the answer to a counting action of type 2, one per line.

Example

Input

```
5
1 1 2
1 2 1
2 1 4
1 1 1
2 2 4
```

Output

```
2
1
```

Explanation

Below is a table showing the position of each frog at each point in time:

Minute	1	2	3	4	5
Frog 1	2	3	4	5	6
Frog 2		1	3	5	7
Frog 3				1	2

SQRT Contest #02 - G - Great string

Time limit: 0.5s **Memory limit:** 256M

Alice has many friends. Today, each of Alice's friends gives her a string. It is known that all the strings have the same length. Alice wants to sort and concatenate the strings in some order such that the resulting string has the smallest lexicographical order. Please help Alice accomplish this task.

Note: A string a is considered lexicographically smaller than string b if and only if there exists an index x such that:

- $a_i = b_i$ for all $1 \leq i < x$,
- $a_x < b_x$.

Input

- Multiple lines, each containing a string that Alice's friends have given her. The total length of all strings does not exceed 10^5.

Output

- A single line containing the string with the smallest lexicographical order that can be formed.

Example

Input

```
ab
bc
ac
```

Output

```
abacbc
```

SQRT Contest #02 - H - Hidden word search

Time limit: 1.0s **Memory limit:** 256M

Alice has a character grid of size $n \times m$. Alice randomly draws a path on the grid such that two consecutive cells on the path share an edge and no cell appears more than once on the path. She records the characters encountered along the path in order from the start to the end, then removes the path from the grid. She gives the grid and the string she recorded to Bob and challenges him to find the path she drew. Can you help Bob find the path?

Input

- The first line contains two integers n and m $(1 \leq n, m \leq 5)$.
- The second line contains a string s, which consists of lowercase English letters, representing the sequence of characters Alice recorded. The length of the string s is at most 15.
- The next n lines each contain a string of m lowercase English letters.

Output

- For each i-th character in the string s, print the position of the character on the path. The position of each character is represented by a pair of integers denoting the row and column of the character (1-based index). If there are multiple solutions, print any valid solution. If no solution exists, print "No solution".

Example

Input

```
3 5
dine
sqrto
nline
judge
```

Output

```
3 3
2 3
2 4
2 5
```

SQRT Contest #02 - I - Integer splitting

Time limit: 10.0s **Memory limit:** 1G

Note the unusual time (and memory) limits for this task.

Given the following expression:

$$\pm 1^k \pm 2^k \pm 3^k \pm \ldots \pm m^k = n$$

A math teacher has proven that for every pair of positive integers (n, k), there always exists a positive integer m and a way to assign the signs ($\pm$) such that the expression is satisfied. For example, with $(n, k) = (7, 3)$, we have $-1^3 + 2^3 = 7$.

The teacher finds this problem interesting, so he decide to turn it into a test for their students. Specifically, the teacher will give each student a pair of positive integers (n, k). Each student is tasked with finding a value of m and a sequence of signs ($\pm$) that satisfy the given pair.

Your task is to solve these problems for the teacher.

Input

- The first line contains a single integer T $(1 \leq T \leq 100)$, the number of test cases.
- Each of the next T lines contains two integers n, k $(0 \leq n \leq 10^{\min(10, 2k)}, 1 \leq k \leq 7)$.

Output

For each test case, output a string consisting of `+` or `-` signs representing how to assign the signs for $1^k, 2^k, \ldots, m^k$ such that the given expression is satisfied.

Example

Input

```
2
0 2
7 3
```

Output

```
++-+--+
-+
```

SQRT Contest #02 - J - Jewels

Time limit: 0.5s **Memory limit:** 256M

A gemstone shop sells a necklace consisting of n red gemstones on the left and m blue gemstones on the right (this necklace is a straight chain, not a circular loop). Alice wants to buy many such necklaces to form one long chain. She wants to know how many red gemstones and blue gemstones are in the first k gemstones of the resulting chain. Can you help Alice calculate this?

Input

- A single line containing three positive integers n, m, k $(1 \leq n, m, k \leq 10^9)$.

Output

- A single line containing two integers: the number of red gemstones and the number of blue gemstones.

Example

Input

```
4 2 6
```

Output

```
4 2
```

SQRT Contest #02 - K - Key

Time limit: 0.5s **Memory limit:** 256M

Alice wants to send Bob a secret number. However, sending it in the usual way is vulnerable to hacker attacks. By chance, today Alice learned about the Fibonacci sequence, which is defined as follows:

- $f_n = 1$ for $1 \leq n \leq 2$.
- $f_n = f_{n-1} + f_{n-2}$ for $n > 2$.

Inspired by the Fibonacci sequence, Alice created a sequence m as follows:

- $m_1 = a, m_2 = b, m_3 = c$.
- $m_n = m_{n-1} \oplus m_{n-2} \oplus m_{n-3}$ for $n > 3$, with $\oplus$ is the XOR bitwise operator.

The number Alice wants to send to Bob is the value of $m_l + m_{l+1} + \ldots + m_r$. Alice will send Bob the five numbers (a, b, c, l, r). Your task is to help Bob find the number Alice wants to send.

Input

- Five positive integers a, b, c, l, r $(1 \leq a, b, c, l, r \leq 10^{12}; l \leq r)$.

Output

- Output the remainder of the number you found when divided by 1234567891.

Example

Input

```
1 2 3 4 5
```

Output

```
1
```

Explanation

- Based on the sequence rules, we can easily calculate $m_4 = 0, m_5 = 1$. From there, we get the number Alice wants to send, which is $m_4 + m_5 = 1$.

SQRT Contest #02 - L - Lights

Time limit: 1.0s **Memory limit:** 256M

Alice has a series of lights, each light can have one of three colors: red, yellow, or green. Initially, the lights are in colors denoted as $a_1, a_2, \ldots, a_n$, where:

- $a_i = 0$ if the i^{th} light is green,
- $a_i = 1$ if the i^{th} light is yellow,
- $a_i = 2$ if the i^{th} light is red.

The lights are connected in a sequence. There are n buttons, and pressing the i^{th} button toggles the state of all lights from position 1 to i as follows:

- A green light turns yellow.
- A yellow light turns red.
- A red light turns green.

Alice wants to know the minimum number of button presses required to turn all the lights green. However, she finds this task too simple, so she decides to make it more challenging. Specifically, Alice will perform q operations to modify the state of the lights. In each operation, she selects three integers l, r, k and modifies the states of the lights from position l to r as follows:

- The l^{th} light is toggled 1^k times,
- The $(l+1)^{th}$ light is toggled 2^k times,
- The $(l+2)^{th}$ light is toggled 3^k times,
- ...
- The r^{th} light is toggled $(r-l+1)^k$ times.

After each operation, Alice wants to know the minimum number of button presses required to turn all lights green.

Input

- The first line contains two positive integers n and q ($1 \le n, q \le 10^5$).
- The second line contains n integers $a_1, a_2, \ldots, a_n$ ($0 \le a_i \le 2$).
- The next q lines each contain three positive integers l, r, k ($1 \le l \le r \le n, 0 \le k \le 10^9$).

Output

The output consists of $q+1$ lines:

- The first line contains a single integer: the minimum number of button presses required to turn all lights green for the initial state.

- The next q lines, where the i^{th} line contains a single integer: the minimum number of button presses required to turn all lights green after the i^{th} operation.

Example

Input

```
5 2
0 1 1 2 2
4 5 1
1 1 2
```

Output

```
3
6
5
```

Explanation

- **Initial State:** The lights are $[0, 1, 1, 2, 2]$. The minimum presses are: press buttons $1, 3, 5$ once each. Sequence of changes:
  - $[0, 1, 1, 2, 2] \to [1, 1, 1, 2, 2] \to [2, 2, 2, 2, 2] \to [0, 0, 0, 0, 0]$.
- **After Operation 1:** Lights become $[0, 1, 1, 0, 1]$. Press buttons $1, 3, 4, 5$ as follows:
  - Button 1 once, button 3 twice, button 4 once, button 5 twice.
- **After Operation 2:** Lights become $[1, 1, 1, 0, 1]$. Press buttons $3, 4, 5$ with the same pattern.