

## Carry Bit (carry)

Valerio is currently visiting Poland, and on the street he found two strings  $A$  and  $B$  of length  $N$  each, consisting of 0's and 1's.



Figure 1: One of the strings found by Valerio.

When he returned home, he decided to write a program that chooses two contiguous substrings of length  $L$ , one from  $A$  and one from  $B$ , then interprets them as binary numbers (i.e. numbers written in base 2) and computes their sum. Unfortunately Valerio made a mistake: he decided to store the result of the sum in an  $L$ -bit variable (i.e., a variable that can store numbers from 0 to  $2^L - 1$  inclusive), but in some cases the sum can be too large! Help him to verify the correctness of the result.

You are given  $Q$  queries, each consisting of 3 integers  $X$ ,  $Y$  and  $L$ . For each query, determine whether Valerio's program is correct for the strings  $A[X \dots X + L - 1]$  and  $B[Y \dots Y + L - 1]$ . In other words, you need to determine whether the sum of the numbers with binary representation  $A[X \dots X + L - 1]$  and  $B[Y \dots Y + L - 1]$  is strictly less than  $2^L$ .

Among the attachments of this task you may find a template file `carry.*` with a sample incomplete implementation.

## Input

The input file consists of:

- a line containing integer  $N$ .
- a line containing string  $A$  of length  $N$ .
- a line containing string  $B$  of length  $N$ .
- a line containing integer  $Q$ .
- $Q$  lines, the  $i$ -th of which consisting of integers  $X_i$ ,  $Y_i$  and  $L_i$ .

## Output





The output file must contain a single line consisting of the  $Q$  integers: the  $i$ -th value must be 1 if Valerio's program gives the correct result to query  $i$ , otherwise the value must be 0.

## Constraints

- $1 \leq N \leq 200\,000$ .
- $A$  and  $B$  consist of characters '0' and '1'.
- $1 \leq Q \leq 200\,000$ .
- $0 \leq X_i, Y_i < N$  for each  $i = 0 \dots Q - 1$ .
- $1 \leq X_i + L \leq N$  for each  $i = 0 \dots Q - 1$ .
- $1 \leq Y_i + L \leq N$  for each  $i = 0 \dots Q - 1$ .

## Scoring

Your program will be tested against several test cases grouped in subtasks. In order to obtain the score of a subtask, your program needs to correctly solve all of its test cases.

- **Subtask 1** (0 points)      Examples.  

- **Subtask 2** (40 points)       $N \leq 4000$  and  $Q \leq 4000$ .  

- **Subtask 3** (30 points)       $X_i = Y_i$  for each  $i = 0 \dots Q - 1$ .  

- **Subtask 4** (30 points)      No additional limitations.  


## Examples

input	output
7 1001010 0111101 5 0 0 1 0 0 3 0 0 4 0 4 3 4 5 2	1 1 0 0 1

input	output
12 101000010101 011110101110 7 1 5 4 8 2 3 9 4 2 10 3 1 0 0 3 0 0 6 3 5 6	1 0 0 1 0 0 1

## Explanation

In the **first sample case** there are 5 queries:

- in the first query the 2 substrings are '1' and '0', and their sum is 1, less than  $2^1$ ;
- in the second query the 2 substrings are '100' and '011', and their sum is 7, less than  $2^3$ ;
- in the third query the 2 substrings are '1001' and '0111', and their sum is 16, not less than  $2^4$ ;
- in the fourth query the 2 substrings are '100' and '101', and their sum is 9, not less than  $2^3$ ;
- in the fifth query the 2 substrings are '01' and '01', and their sum is 2, less than  $2^2$ .