



Tricolor Lights

Anna and Bruno, experts in gambling, are about to compete against Dealer D-taro in a game.

During the game, Anna and Bruno are isolated in separate rooms and can only communicate through Dealer D-taro.

In this game, they use a row of N lights. These lights are numbered from 1 to N from left to right and can be illuminated in one of three colors: red, green, or blue.

At the beginning of the game, Anna illuminates each of these lights in either red, green, or blue. Additionally, Dealer D-taro specifies one **prohibited color** for each light, represented by a string S of length N . Let S_i denote the i -th character of S ($1 \leq i \leq N$). If S_i is 'R', the prohibited color for light i is red; if S_i is 'G', it's green; and if S_i is 'B', it's blue. Anna cannot illuminate light i in its prohibited color. For example, if S_1 is 'R', Anna cannot illuminate light 1 in red. Dealer D-taro conveys the information of the prohibited color for each light to Anna but not to Bruno.

After illuminating each light, Anna selects an integer l satisfying $1 \leq l \leq \min(N, 130)$ and informs Dealer D-taro. Dealer D-taro then informs Bruno of the total number of lights N and the integer l chosen by Anna. Subsequently, they proceed with Q rounds as follows:

1. Dealer D-taro selects an integer a_j between 1 and $N - l + 1$ and shows Bruno the sequence of colors illuminated for lights $a_j, a_j + 1, \dots, a_j + l - 1$.
2. Bruno responds with a single integer to Dealer D-taro based on the sequence of colors shown. If the integer matches a_j , Anna and Bruno win the round.

However, Dealer D-taro may vary the selection of a_1, a_2, \dots, a_Q based on the sequence of colors illuminated by Anna and the chosen integer l .

You are required to implement a program that allows Anna and Bruno to win all Q rounds.



Implementation Details

You need to submit 2 files.

The first file is `Anna.cpp`. It should implement Anna's strategy. It should implement the following functions. The program should include `Anna.h` using the preprocessing directive `#include`.

- `std::pair<std::string, int> anna(int N, std::string S)`

This function is called once initially.

- The parameter N is the number of lights.
- The parameter S is a string of length N representing the prohibited colors specified by Dealer D-taro.
- The return value is a pair consisting of a string t representing the colors Anna illuminates for each light and an integer l chosen by Anna. Let t_i denote the i -th character of t ($1 \leq i \leq N$). t_i signifies Anna illuminating the i -th lamp with color t_i . If t_i is 'R', the prohibited color for light i is red; if t_i is 'G', it's green; and if t_i is 'B', it's blue.
- The length of string t must be N . Otherwise, it will be judged as **Wrong Answer [1]**.
- Each character in string t must be 'R', 'G', or 'B'. Otherwise, it will be judged as **Wrong Answer [2]**.
- Each character of string t must be different from the corresponding character in S . Otherwise, it will be judged as **Wrong Answer [3]**.
- l must be between 1 and $\min(N, 130)$. Otherwise, it will be judged as **Wrong Answer [4]**.

The second file is `Bruno.cpp`. It should implement Bruno's strategy. It should implement the following functions. The program should include `Bruno.h` using the preprocessing directive `#include`.

- `void init(int N, int l)` This function is called once initially.
 - The parameter N is the number of lights.
 - The parameter l is the integer chosen by Anna.
- `int bruno(std::string u)` This function is called Q times after function `init` is called and corresponds to steps 1 and 2 of each round in the game.
 - The argument u is a string of length l consisting of 'R', 'G', 'B', representing the sequence of colors illuminated for lights $a_j, a_j + 1, \dots, a_j + l - 1$.
 - Let u_k denote the k -th character of u ($1 \leq k \leq l$). If u_k is 'R', the color of $a_j + k - 1$ -th light that Anna illuminated is red; if u_k is 'G', it's green; and if u_k is 'B', it's blue.
 - The return value is the integer Bruno responds with.
 - The return value must match a_j . Otherwise, it will be judged as **Wrong Answer [5]**.



Important Notices

- Your program can implement other functions for internal use, or use global variables. Submitted files will be compiled with the grader, and become a single executable file. All global variables and internal functions should be declared in an unnamed namespace to avoid conflict with other files. When it is graded, it will be executed as two processes of Anna and Bruno. The process of Anna and the process of Bruno cannot share global variables.
- Your program must not use the standard input and the standard output. Your program must not communicate with other files by any methods. However, your program may output debugging information to the standard error.

Compilation and Test Run

You can download an archive file from the contest webpage which contains the sample grader to test your program. The archive file also contains a sample source file of your program.

The sample grader is the file `grader.cpp`. In order to test your program, put `grader.cpp`, `Anna.cpp`, `Bruno.cpp`, `Anna.h`, `Bruno.h` in the same directory, and run the following command to compile your programs. Instead, you may run `compile.sh` contained in the archive file.

```
g++ -std=gnu++20 -O2 -o grader grader.cpp Anna.cpp Bruno.cpp
```

When the compilation succeeds, the executable file `grader` is generated.

Note that the actual grader is different from the sample grader. The sample grader will be executed as a single process, which will read input data from the standard input and write the results to the standard output and the standard error output.



Input of the Sample Grader

The sample grader reads the following data from the standard input.

```
N
S
Q
a1 a2 ⋯ aQ
```

Unlike the actual grading program, the sample grading program must have fixed answers. a_j ($1 \leq j \leq Q$) represents the integer a_j chosen by Dealer D-taro in the j th round. In your program, a_j must satisfy $1 \leq a_j \leq N - l + 1$ for the integer l chosen by Anna.

Output of the Sample Grader

The sample grader outputs the following information to the standard output and the standard error output (quotes for clarity).

- For correct answers, “Accepted: 130” is output for the integer l chosen by Anna.
- For wrong answers, the type of wrongness is output, such as “Wrong Answer [1]”.

If your program meets the conditions of several types of Wrong Answer, the sample grader reports only one of them.

Grading Considerations

The actual grading program determines a_1, a_2, \dots, a_Q based on the sequence of colors illuminated by Anna and the chosen integer l . But Bruno’s response does not affect the selection of a_1, a_2, \dots, a_Q .

Constraints

- $1 \leq N \leq 500\,000$.
- $1 \leq Q \leq 10\,000$.
- S is a string of length N consisting of ‘R’, ‘G’, or ‘B’.
- N and Q are integers.



Subtasks

1. (5 points) $N \leq 131$.
2. (5 points) $N \leq 250$.
3. (5 points) $N \leq 380$.
4. (15 points) $N \leq 7\,000$.
5. (70 points) No additional constraints. In this subtask, points are awarded as follows:
 - Let l^* denote the maximum value of l chosen by Anna in all test cases of this subtask.
 - If any test case in this subtask is judged as **Wrong Answer [1] ~ [5]** (see implementation details) or exceeds the time limit, memory limit, or encounters a runtime error, the subtask score is 0 points.
 - If all test cases in this subtask are correct, the subtask score is determined as follows:

the value of l^*	score
$61 < l^* \leq 130$	10 points
$41 < l^* \leq 61$	20 points
$34 < l^* \leq 41$	$25 + 3 \times (41 - l^*)$ points
$28 < l^* \leq 34$	$46 + 4 \times (34 - l^*)$ points
$l^* \leq 28$	70 points



Sample Communication

Here is a sample input for the sample grader and corresponding function calls.

Sample Input 1	Sample Function Calls	
	Call	Return value
8	<code>anna(8, "RGGBRBBG")</code>	<code>("BBRGBGRR", 5)</code>
RGGBRBBG	<code>init(8, 5)</code>	
2	<code>bruno("RGBGR")</code>	3
3 1	<code>bruno("BBRGB")</code>	1

In this example, Anna receives the number of lights $N = 8$ and the string representing prohibited colors $S = \text{"RGGBRBBG"}$. Anna chooses to illuminate each light with the string t representing colors `"BBRGBGRR"` and selects the integer l as 5, which she communicates to Dealer D-taro. Then, Dealer D-taro informs Bruno of the numbers $N = 8$ and $l = 5$.

In the first round, Dealer D-taro chooses $a_1 = 3$. Bruno receives the sequence of colors illuminated for lights 3, 4, 5, 6, and 7, represented by the string $u = \text{"RGBGR"}$, and responds with the integer 3, which matches a_1 .

Input example 1 satisfies all constraints for all subtasks. The file `sample-01-in.txt` downloadable from the contest site corresponds to input example 1.