

D2. RGB Substring (hard version)

time limit per test: 2 seconds
 memory limit per test: 256 megabytes
 input: standard input
 output: standard output

The only difference between easy and hard versions is the size of the input.

You are given a string s consisting of n characters, each character is 'R', 'G' or 'B'.

You are also given an integer k . Your task is to change the minimum number of characters in the initial string s so that after the changes there will be a string of length k that is a substring of s , and is also a substring of the infinite string "RGBRGBRGB...".

A string a is a substring of string b if there exists a positive integer i such that $a_1 = b_i$, $a_2 = b_{i+1}$, $a_3 = b_{i+2}$, ..., $a_{|a|} = b_{i+|a|-1}$. For example, strings "GBRG", "B", "BR" are substrings of the infinite string "RGBRGBRGB..." while "GR", "RGR" and "GGG" are not.

You have to answer q independent queries.

Input

The first line of the input contains one integer q ($1 \leq q \leq 2 \cdot 10^5$) — the number of queries. Then q queries follow.

The first line of the query contains two integers n and k ($1 \leq k \leq n \leq 2 \cdot 10^5$) — the length of the string s and the length of the substring.

The second line of the query contains a string s consisting of n characters 'R', 'G' and 'B'.

It is guaranteed that the sum of n over all queries does not exceed $2 \cdot 10^5$ ($\sum n \leq 2 \cdot 10^5$).

Output

For each query print one integer — the minimum number of characters you need to change in the initial string s so that after changing there will be a substring of length k in s that is also a substring of the infinite string "RGBRGBRGB...".

Example

input	Copy
3 5 2 BGGGG 5 3 RBRGR 5 5 BBBRR	
output	Copy
1 0 3	

Note

In the first example, you can change the first character to 'R' and obtain the substring "RG", or change the second character to 'R' and obtain "BR", or change the third, fourth or fifth character to 'B' and obtain "GB".

In the second example, the substring is "BRG".

Codeforces Round #575 (Div. 3)

Finished

Practice



→ Virtual participation

Virtual contest is a way to take part in past contest, as close as possible to participation on time. It is supported only ACM-ICPC mode for virtual contests. If you've seen these problems, a virtual contest is not for you - solve these problems in the archive. If you just want to solve some problem from a contest, a virtual contest is not for you - solve this problem in the archive. Never use someone else's code, read the tutorials or communicate with other person during a virtual contest.

Start virtual contest

→ Practice

You are registered for practice. You can solve problems unofficially. Results can be found in the contest status and in the bottom of standings.

→ Clone Contest to Mashup

You can clone this contest to a mashup.

Clone Contest

→ Submit?

Language: GNU G++11 5.1.0

Choose file: 浏览...

Submit

→ Last submissions

Submission	Time	Verdict
58118535	Aug/01/2019 12:45	Accepted
58118408	Aug/01/2019 12:42	Wrong answer on test 1
58118278	Aug/01/2019 12:40	Wrong answer on test 1

→ Problem tags

data structures dp implementation

two pointers *1600

No tag edit access

→ Contest materials

- Announcement #1 (en) 

- | | |
|------------------------|---|
| • Announcement #2 (ru) | × |
| • Tutorial #1 (en) | × |
| • Tutorial #2 (ru) | × |
| • Tutorial #3 (en) | × |
| • Tutorial #4 (en) | × |
| • Tutorial #5 (ru) | × |
| • Tutorial #6 (en) | × |

[Codeforces](#) (c) Copyright 2010-2019 Mike Mirzayanov
The only programming contests Web 2.0 platform
Server time: Aug/01/2019 20:09:19^{UTC+8} (g1).
Desktop version, switch to [mobile version](#).
[Privacy Policy](#)

Supported by

