10
LIVE EVENTS

Holi And Colorful houses

Attempted by: **735** / Accuracy: **83%** / Maximum Score: **20** / ★★★⯨☆ 3 Votes

Tag(s): Basic Programming, Basics of Implementation, Easy, Implementation

PROBLEM **EDITORIAL** **MY SUBMISSIONS**

This Holi, Monk wants to distribute sweets to the houses of his colony. The houses of the colony are present in a circular order (i.e. house 1 and house N are adjacent to each other).

Kids love to play with colors but currently there is a supply of only 2 types of colors (Red represented by R and Green represented by G). Due to festive season each of the kids have colored their houses with either green or red color.

Monk is given a task of distributing sweets Q number of times. Every time he is asked to travel from x^{th} house to y^{th} house to distribute the sweets.
The distribution strategy is that he starts from x^{th} house and he has to give 1 sweet to a house only when he travels from **green house to a red house or vice-versa**. Monk can travel from x^{th} house to y^{th} house either in **clockwise direction or in anti-clockwise direction**.
Monk wants your help to find the **minimum** number of sweets he should carry to complete his journey.

Input Format

First line contains an integer T denoting the number of test cases.

Each test case contains an integer N and Q.

Next line contains a string S (representing the house colors) of length N.

Then Q lines follow, each containing 2 integers x and y.

Output Format

For each test case print Q lines containing the minimum number of sweets he must carry with himself.

Input Constraints

$1 \leq T \leq 100$

$1 \leq N, Q \leq 1000$

$1 \leq x, y \leq N$

S consists of only letters 'G' and 'R'

SAMPLE INPUT

```
2
5 2
RRRGG
1 5
3 2
5 2
GGRGG
2 5
3 1
```

SAMPLE OUTPUT

```
1
0
0
1
```

Explanation

Case 1:

For 1st query, if we travel in the direction 1->2->3->4->5, we need only one sweet . If we travel in the direction 1->5, we need only one sweet. So he has to carry only 1 sweet.

For 2nd query, if we travel in the direction 3->4->5->1->2, we need two sweets. If we travel in the direction 3->2, we need no sweets. So he has to carry 0 sweets.

Case 2:

For 1st query house, **2** and **5** can be travelled using path 2->1->5 without any change of colors so he has to carry 0 sweets.

For 2nd query house **3** and **1** can be travelled using path 3->2->1 with a change of color so he has to carry 1 sweet.

Time Limit:	1.0 sec(s) for each input file.
Memory Limit:	256 MB
Source Limit:	1024 KB
Marking Scheme:	Marks are awarded when all the testcases pass.
Allowed Languages:	C, C++, C++14, Clojure, C#, D, Erlang, F#, Go, Groovy, Haskell, Java, Java 8, JavaScript(Rhino), JavaScript(Node.js), Julia, Kotlin, Lisp, Lisp (SBCL), Lua, Objective-C, OCaml, Octave, Pascal, Perl, PHP, Python, Python 3, R(RScript), Racket, Ruby, Rust, Scala, Swift, Visual Basic