If a string is in the form **UVU**, where **U** is not empty, and **V** has exactly **L** characters, we say **UVU** is an **L**-Gap string. For example, **abcbabc** is a **1**-Gap string. **xyxyxyxyxy** is both a **2**-Gap string and also a **6**-Gap string, but not a **10**-Gap string (because **U** is non-empty).

Given a string **s**, and a positive integer g, you are to find the number of g-Gap substrings in **s**. **s** contains lower-case letters only, and has at most **50,000** characters.

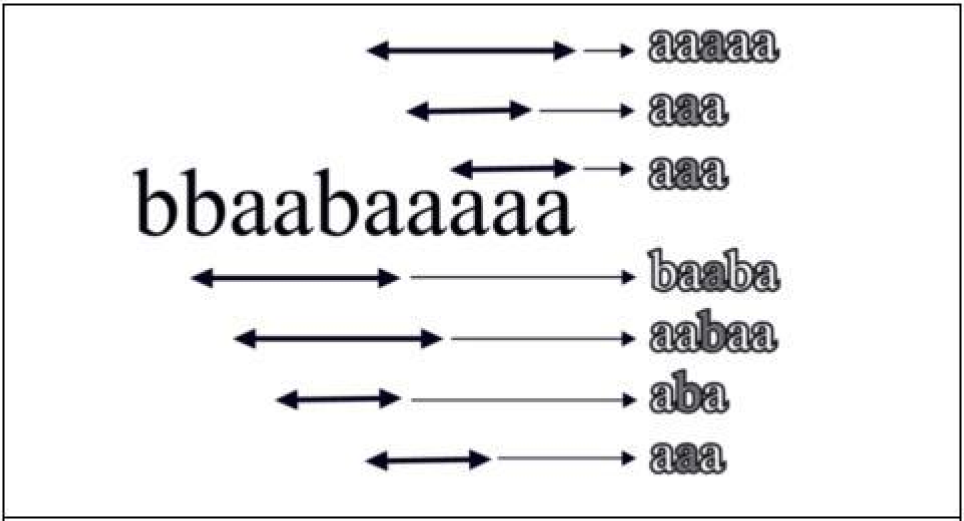

Figure: This figure demonstrates the first sample input. It shows all the 1-gap substrings of the string "bbaabaaaaa". In each individual substring, U & V are shown in light and dark letters respectively.

Input

The first line contains a single integer t ($1 \leq t \leq 10$), the number of test cases. Each of the t followings contains an integer g ($1 \leq g \leq 10$) followed by a string **s**.

Output

For each test case, print the case number and the number of g-Gap substrings. Look at the output for sample input for details.

Sample Input

```
2
1 bbaabaaaaa
5 abxxxxxab
```

Sample Output

```
Case 1: 7
Case 2: 1
```