

# Lost Password

## Problem

Ashish has forgotten his password. He remembers that he used the following algorithm to create his password: Ashish took up to  $k$  consecutive words from a passage of text, and took the first letter from each word. Then, he might have changed some of the letters to their "l33tspeak" equivalents. Specifically, he might have changed "o" to "0", "i" to "1", "e" to "3", "a" to "4", "s" to "5", "t" to "7", "b" to "8" and/or "g" to "9".

For example, if Ashish took his password from the first sentence of *The Fellowship of the Ring* -- *"This book is largely concerned with Hobbits, and from its pages a reader may discover much of their character and a little of their history"* -- Ashish would have reduced that to "tbilcwhafiparmdmotcaaloth". Then the password might be "tbilcwh", "7b1lcwh4f", "a", "4", or "4al07h", etc.

Ashish has a special extension installed in his browser that will prevent his computer from uploading any string that contains his password. In order to figure out which passage of text he took his password from, Ashish has created a webpage to take advantage of this extension. Every second, the webpage will tell the browser to post a "password string" for a new passage of text: a string that contains all of the possible passwords that Ashish could have chosen from that passage of text. As soon as his browser fails to post such a string, Ashish will know where he took his password from.

For example, if  $k = 2$  and the passage of text contains words starting with the letters "google", then one password string for that passage is "goo0og00gle9o909l3". All substrings of length  $\leq 2$  from the original string, and all of their l33tspeak equivalents, are contained in the new string.

Given the first letters of the words in a passage of text, what is the minimum number of characters in the "password string" of that passage?

## Input

The first line of the input gives the number of test cases,  $T$ .  $T$  test cases follow. Each test case consists of two lines. The first line contains the integer  $k$ . The second line contains a string  $S$ , representing the first letters of the words in a passage of text.  $S$  contains only the characters 'a' - 'z', with no spaces.

## Output

For each test case, output one line containing "Case #x: y", where  $x$  is the case number (starting from 1) and  $y$  is the minimum number of characters in the password string for  $S$ .

## Limits

Memory limit: 1GB.

Time limit: 40 seconds per test set.

$1 \leq T \leq 20$ .

$S$  will contain at least  $2 * k$  characters.

There will exist a password string with at most  $10^{18}$  characters.

### Test set 1 (Visible Verdict)

**S** will contain at most 1000 characters.  
 $k = 2$ .

### Test set 2 (Hidden Verdict)

**S** will contain at most 5000 characters.  
 $2 \leq k \leq 500$ .

### Sample

#### Sample Input

```
4
2
poppop
2
google
2
tbilcwhafiparmdmotcaaloth
10
tbilcwhafiparmdmotcaaloth
```

#### Sample Output

```
Case #1: 6
Case #2: 18
Case #3: 53
Case #4: 1136
```

In the first sample input, one possible password string is "0ppop0".

In the second sample input, one possible password string is "goo0og00gle9o909l3".