

Given a string S and an integer K , another string T is obtained by **concatenating** S , K **times**. How many **distinct substrings** are there in the **string** T ?

For example, when $S = \text{"ab"}$, $K = 2$: $T = \text{"abab"}$ and there are **7** distinct substrings in the string T and they are: **"a"**, **"b"**, **"ab"**, **"ba"**, **"aba"**, **"bab"** and **"abab"**.

Input

First line of input contains an integer T (< 101) which is the number of test cases. Each of the following T lines contain a string S and an integer K ($2 \leq K \leq 10^9$). The length of S is at most 50000 and it consists of **lowercase** letters only and the string is non-empty.

Output

For each test case, output the case number followed by the number of distinct substrings. The input will be such that the result will **always fit into a 64-bit** signed integer number.

Sample Input

```
3
ab 3
abc 5
aba 4
```

Sample Output

```
Case 1: 11
Case 2: 42
Case 3: 32
```