

Given a string containing only alpha-numerals (0-9, A-Z and a-z) you have to count the frequency (the number of times the character is present) of all the characters and report only those characters whose frequency is a prime number. A prime number is a number, which is divisible by exactly two different integers. Some examples of prime numbers are **2, 3, 5, 7, 11** etc.

Input

The first line of the input is an integer T ($0 < T < 201$) that indicates how many sets of inputs are there. Each of the next T lines contains a single set of input.

The input of each test set is a string consisting alpha-numerals only. The length of this string is positive and less than **2001**.



Output

For each set of input produce one line of output. This line contains the serial of output followed by the characters whose frequency in the input string is a prime number. These characters are to be sorted in lexicographically ascending order. Here “lexicographically ascending” means ascending in terms of the **ASCII** values. Look at the output for sample input for details. If none of the character frequency is a prime number, you should print ‘empty’ (without the quotes) instead.

Sample Input

```
3
ABCC
AABBBBDDDDD
ABCDFFFF
```

Sample Output

```
Case 1: C
Case 2: AD
Case 3: empty
```