In programming multiple whitespaces are used to only to make code more readable, so mostly all programming languages totally ignore multiple spaces in code (except for some esoteric ones). In general there are different types of whitespace characters: space itself, tabs, newline symbol, various control characters, etc. Tabs and spaces bring one or the biggest holywar to a programmers world as there is no common rule what to use for code indentation — tab or space characters.

In this holywar you stand for tab side and your project code convention requires to use only them for code indentation. However you have recently spotted that someone is using space characters instead of it. Four spaces and tab character look the same in our text editor, so you have decided to write a parser that will change all consequent space characters to one. After that you would be able to determine amount of corrupted code.

Input

The number of tests T ($T \leq 100$) is given on the first line. At first line of each test there is integer N ($N \leq 50$). Next N lines with text that must be processed for extra spaces follow. Maximal line length is equal to 500 characters.

Output

For each test case output a single line 'Case T:'. Where T is the test case number (starting from 1). Next N lines must be output with input text having no consequent spaces. By the way, always leave a blank line between tests. Please refer to the sample output for clarity.

Sample Input

```
2
3
Sample test one:
  there was 2 spaces and
here are also  2  spaces
2
Sample test two:
     there was 4 spaces
```

Sample Output

```
Case 1:
Sample test one:
 there was 2 spaces and
here are also 2 spaces

Case 2:
Sample test two:
 there was 4 spaces
```