C - Classy Problem

In his memoir *So, Anyway*, comedian John Cleese writes of the class difference between his father (who was *middle-middle-middle-lower-middle class* and his mother (who was *upper-upper-lower-middle class*). These fine distinctions between classes tend to confuse American readers, so you are to write a program to sort a group of people by their classes to show the true distinctions.

For this problem, there are three main classes: upper, middle, and lower. Obviously, the highest is upper and the lowest is lower. But there can be distinctions within a class, so upper-upper is a higher class than middle-upper, which is higher than lower-upper. However, all of the upper classes (upper-upper, middle-upper, and lower-upper) are higher than any of the middle classes.

Within a class like middle-upper, there can be further distinctions as well, leading to classes like lower-middle-upper-middle-upper. When comparing classes, once you've reached the lowest level of detail, you should assume that all further classes are the same as the middle level of the previous level of detail. So upper class and middle-upper class are equivalent, as are middle-middle-lower-middle and lower-middle.

The Input

The first line of input will indicate the number of test cases. For each test case that follows, the first line of input contains n, the number of names to follow. Each of the following n lines contains the name of a person as a sequence of 1 or more lower case letters, then a colon, then a single space, and then the class of the person. The class of the person will include one or more modifiers, separated by a single space, followed by a single space and then the word **class**. The entire line will be no longer than 256 characters.

The Output

For each test case, the output will consist of a list of names from highest to lowest class, one name per line. If two people have the same class, they should be listed in alphabetical order by name. Print one blank line after each test case.

Sample Input

```
2
5
mom: upper upper lower middle class
dad: middle middle lower middle class
queenelizabeth: upper upper class
chair: lower lower class
unclebob: middle lower middle class
10
rich: lower upper class
mona: upper upper class
dave: middle lower class
charles: middle class
tom: middle class
william: lower middle class
carl: lower class
violet: middle class
frank: lower class
mary: upper class
```

Sample Output

```
queenelizabeth
mom
dad
unclebob
chair

mona
mary
rich
charles
tom
violet
william
carl
dave
frank
```