

# Name Sorting

The king of Byteland wants to make a family ancestral archive. Now you being the best coder in the country have been called by the king in order to do this task for him. This will require a lot of work, but first the king wants to test your capability. He will give you list of names his ancestors you have to sort them in increasing manner. Now here the catch is that each name is of the format "FirstName Roman-number". Firstly they must be sorted lexicographically as per their FirstName, and in case the FirstName is same they must be sorted as per the Roman-number accredited to them. For sorting as per the Roman-number consider the integer value they represent.

## Facts about Roman-numbers:

- The Roman numerals for 1 through 10 are I, II, III, IV, V, VI, VII, VIII, IX, and X.
- The Roman numerals for 20, 30, 40, and 50 are XX, XXX, XL, and L.
- The Roman numeral for any other two-digit number less than 50 can be constructed by concatenating the numeral for its tens and the numeral for its ones.
- For example, 47 is  $40 + 7 = \text{"XL"} + \text{"VII"} = \text{"XLVII"}$ .

## Input

The first line contains  $T(\leq 10)$ : number of test cases.

Each test on the first line will have  $N(\leq 1000)$ , the number of names you need to sort.

The next  $N$  lines contain  $N$  names in the above specified manner

(The integral value of the Roman-numbers will be in range of 1 to 50 both inclusive.)

## Output

For each of the  $T$  test cases print  $N$  line having the names sorted in ascending order.

## Example

Sample Input:

```
2
2
Louis IX
Louis VIII
6
Philippe VI
```

Jean II  
Charles V  
Charles VII  
Charles VI  
Louis XI

Sample Output:

Louis VIII  
Louis IX  
Charles V  
Charles VI  
Charles VII  
Jean II  
Louis XI  
Philippe VI