


Sort of Sorting

Problem ID: sortofsortin**CPU Time limit:** 2 seconds**Memory limit:** 1024 MB**Author:** Greg Hamerly**Source:** Baylor Competitive Programming course**License:** 

Can you believe school has already started? It seems like we were just finishing last semester. Last semester was tough because the administration had a hard time keeping records of all the students in order, which slowed everything down. This year, they are going to be on top of things. They have recognized that you have the skills to help them get into shape with your programming ability, and you have volunteered to help. You recognize that the key to getting to student records quickly is having them in a sorted order. However, they don't really have to be perfectly sorted, just so long as they are sort-of sorted.

Write a program that sorts a list of student last names, but the sort only uses the first two letters of the name. Nothing else in the name is used for sorting. However, if two names have the same first two letters, they should stay in the same order as in the input (this is known as a 'stable sort'). Sorting is *case sensitive* based on ASCII order (with uppercase letters sorting before lowercase letters, i.e., $A < B < \dots < Z < a < b < \dots < z$).

Input

Input consists of a sequence of up to 500 test cases. Each case starts with a line containing an integer $1 \leq n \leq 200$. After this follow n last names made up of only letters (a–z, lowercase or uppercase), one name per line. Names have between 2 and 20 letters. Input ends when $n = 0$.

Output

For each case, print the last names in sort-of-sorted order, one per line. Print a blank line between cases.

Sample Input 1

```
3
Mozart
Beethoven
Bach
5
Hilbert
Godel
Poincare
Ramanujan
Pochhammer
0
```

Sample Output 1

```
Bach
Beethoven
Mozart

Godel
Hilbert
Poincare
Pochhammer
Ramanujan
```