Problem Description

The input contains N four-digit integers. For example, if N=9, the input may look like

3412
5232
9128
3456
1324
1928
3344
7631
4343

For each integer, we rearrange its digits from large to small and obtain a new integer. As a result, the nine integers above become

4321
5322
9821
6543
4321
9821
4433
7631
4433

For output, we list the modified integers in a decreasing order line by line. Each integer only needs to be listed once.

9821
7631
6542
5322
4433
4321

<Hint>
You may consider the following algorithm:
0. Create a character array (char str[5];) for storing the input integers and create an array (int HIT[10000];) to record the occurrences of input integers.
1. Read the number of input integers and store it in variable N (scanf("%d", &N);).
2. For each of the next N integers, do

2.1 Read the four-digit integer and store it as a four-character string in str.

2.2 Use the bubble sort algorithm to rearrange the four characters.

2.3 Convert the rearranged string to an integer m and set the value of the corresponding element of the HIT array to 1 (HIT[m]=1;).

3. Check each element of the HIT array; if its value is 1, print the corresponding integer.

Note that the first digit is nonzero.

The bubble sort algorithm:

```
/* Using bubble sort to rearrange an array A[n] */


for (i = 0; i < n; i++) {
   for (j = 1; j < n; j++) {
      if (A[j-1] > A[j]) {
         /* swap A[j-1] A[j] */
      }
   }
}
```

Input

The first line contains a number N (1<=N<=100).

The next N lines provide the N four-digit integers.

Output

Each line shows a modified unique integer as explained in the problem description. Remember to print a newline at the end of each line.

Sample Input

```
5
1234
5678
2468
1357
4123
```

Sample Output

```
8765
8642
7531
4321
```