Big Sorting

Problem Submissions Leaderboard

Consider an array of numeric strings where each string is a positive number with anywhere from 1 to 10^6 digits. Sort the array's elements in *non-decreasing*, or ascending order of their integer values and return the sorted array.

Example

$unsorted = [\text{'1'}, \text{'200'}, \text{'150'}, \text{'3'}]$

Return the array ['1', '3', '150', '200'].

Function Description

Complete the *bigSorting* function in the editor below.

bigSorting has the following parameter(s):

- *string unsorted[n]:* an unsorted array of integers as strings

Returns

- *string[n]:* the array sorted in numerical order

Input Format

The first line contains an integer, n, the number of strings in $unsorted$.
Each of the n subsequent lines contains an integer string, $unsorted[i]$.

Constraints

- $1 \leq n \leq 2 \times 10^5$
- Each string is guaranteed to represent a positive integer.
- There will be no leading zeros.
- The total number of digits across all strings in $unsorted$ is between 1 and 10^6 (inclusive).

Sample Input 0

```
6
31415926535897932384626433832795
1
3
10
3
5
```

Sample Output 0

```
1
3
3
5
```

```
10
31415926535897932384626433832795
```

Explanation 0

The initial array of strings is $unsorted = [31415926535897932384626433832795, 1, 3, 10, 3, 5]$. When we order each string by the real-world integer value it represents, we get:

$$1 \leq 3 \leq 3 \leq 5 \leq 10 \leq 31415926535897932384626433832795$$

We then print each value on a new line, from smallest to largest.

Sample Input 1

```
8
1
2
100
123034798498573417183401923 71
3084193741082937
3084193741082938
111
200
```

Sample Output 1

```
1
2
100
111
200
3084193741082937
3084193741082938
123034798498573417183401923 71
```


Python 3

```python
#!/bin/python3

import math
import os
import random
import re
import sys

#
# Complete the 'bigSorting' function below.
#
# The function is expected to return a STRING_ARRAY.
# The function accepts STRING_ARRAY unsorted as parameter.
#

def bigSorting(unsorted):
    return sorted(unsorted,key=lambda x: (len(x), x))

if __name__ == '__main__':
```

```
    fptr = open(os.environ['OUTPUT_PATH'], 'w')

    n = int(input().strip())

    unsorted = []

    for _ in range(n):
        unsorted_item = input()
        unsorted.append(unsorted_item)

    result = bigSorting(unsorted)

    fptr.write('\n'.join(result))
    fptr.write('\n')

    fptr.close()

```

Line: 1 Col: 1

Upload Code as File Test against custom input Run Code Submit Code