Sort Command #3

Objective

In this challenge, we practice using the *sort* command to sort input in text or TSV formats.

Resources

The *sort* command is frequently used for sorting input in text or TSV formats in various different ways. These ways may be either lexicographical, case insensitive, based on the numeric field only, based on a particular column, etc.

Sort Options

- The vanilla *sort* command simply sorts the lines of the input file in lexicographical order.
- The **-n** option sorts the file on the basis of the numeric fields available if the first word or column in the file is a number.
- The **-r** option reverses the sorting order to either the reverse of the usual lexicographical ordering or descending order while sorting in numerical mode.
- The **-k** option is useful while sorting a table of data (tsv, csv etc.) based on a specified column (or columns).
- The **-t** option is used while specifying a delimiter in a particular file where columns are separated by tabs, spaces, pipes etc.

A few useful resources to study different variants as well as working examples of the *sort* command are:
A Wikipedia entry for the 'sort' command
How to Sort Files in Linux using Sort Command

Task

You are given a text file where each line contains a number. The numbers may be either an integer or have decimal places. There will be no extra characters other than the number or the newline at the end of each line. Sort the lines in ascending order - so that the first line holds the numerically smallest number, and the last line holds the numerically largest number.

Input Format

A text file where each line contains a positive number (less than 100) as described above.

Output Format

Output the text file with the lines reordered in numerically ascending order.

Sample Input

```
9.1
43.7
2.2
62.1
2.1
9.3
43.5
4.6
44.6
4.7
42.7
47.4
46.6
4.5
```

55.6
4
9.2
66.6
2
2.3

Sample Output

2
2.1
2.2
2.3
4
4.5
4.6
4.7
9.1
9.2
9.3
42.7
43.5
43.7
44.6
46.6
47.4
55.6
62.1
66.6