using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

namespace ConsoleApplication1

{

class Program

{

//NO FUNCIONA CASTEANDO CADA CARACTER A STRING, DA TIME LIMIT EXCEEDED

//static int Comparar(string a, string b)

//{

// if (a.Length > b.Length) return 1;

// else if (b.Length > a.Length) return -1;

// for (int i = 0; i < a.Length; i++)

// {

// if (int.Parse(a[i].ToString()) > int.Parse(b[i].ToString()))

// {

// return 1;

// }

// else if (int.Parse(b[i].ToString()) > int.Parse(a[i].ToString()))

// {

// return -1;

// }

// }

// return 0;

//}

static int Comparar(string a, string b)

{

if (a.Length > b.Length) return 1;

else if (b.Length > a.Length) return -1;

for (int i = 0; i < a.Length; i++)

{

if (a[i] > b[i])

{

return 1;

}

else if (b[i] > a[i])

{

return -1;

}

}

return 0;

}

static int partition(string[] arr, int low, int high)

{

string pivot = arr[high];

string temp;

int i = (low - 1); // index of smaller element

for (int j = low; j <= high - 1; j++)

{

// If current element is smaller than or

// equal to pivot

if(Comparar(arr[j], pivot)<0) //if (arr[j] <= pivot)

{

i++;

// swap arr[i] and arr[j]

temp = arr[i];

arr[i] = arr[j];

arr[j] = temp;

}

}

// swap arr[i+1] and arr[high] (or pivot)

temp = arr[i + 1];

arr[i + 1] = arr[high];

arr[high] = temp;

return i + 1;

}

static void quickSort(string [] array, int start, int end)

{

if (start < end)

{

int pivotIndex = partition(array, start, end);

quickSort(array, start, pivotIndex - 1);

quickSort(array, pivotIndex + 1, end);

}

}

static void Main(string[] args)

{

//int n = int.Parse(Console.ReadLine());

//int[] arr = Array.ConvertAll(Console.ReadLine().Split(' '), e => int.Parse(e));

//quickSort(arr, 0, arr.Length - 1);

//string[] s = {

//"6",

//"31415926535897932384626433832795",

//"1",

//"3",

//"10",

//"3",

//"5"};

//quickSort(s, 0, s.Length - 1);

//foreach (string elem in s)

//{

// Console.WriteLine(elem);

//}

int n = int.Parse(Console.ReadLine());

string[] s = new string[n];

for (int i = 0; i < n; i++)

{

s[i] = Console.ReadLine();

}

quickSort(s, 0, n - 1);

foreach (string elem in s)

{

Console.WriteLine(elem);

}

Console.ReadLine();

}

}

}