

Экзаменационный контекст 2

11 янв 2022, 18:13:11

старт: 24 дек 2020, 15:00:00

финиш: 24 дек 2020, 18:00:00

длительность: 03:00:00

начало: 24 дек 2020, 15:00:00

конец: 24 дек 2020, 18:00:00

G. Рождественский челлендж

Ограничение времени	2 секунды
Ограничение памяти	64Mb
Ввод	стандартный ввод или input.txt
Вывод	стандартный вывод или output.txt

На одном популярном сайте для подготовки к техническим собеседованиям запустили следующий рождественский челлендж для программистов на языке C#: дан базовый абстрактный класс, содержащий поле со ссылкой на массив, заголовок индексатора и метода для вычисления некоторой метрики. Необходимо реализовать абстрактные члены класса в классе-наследнике. Существует несколько вариантов того, каким именно образом должны быть реализованы индексатор и метод для вычисления метрики, они распределяются между участниками случайным образом. Вам выпала следующая конфигурация:

- Индексатор должен возвращать ближайшее число в массиве, которое меньше заданного. Если такое число отсутствует, необходимо выбрасывать исключение типа `ArgumentException` с сообщением: "Number does not exist."
- Метод для подсчета метрики должен возвращать отношение количества цифр шесть, встречающихся в элементах массива, к общему количеству цифр в элементах массива.

Формат ввода

Программа получает на вход строку, состоящую из целых неотрицательных чисел, записанных через пробел – элементы массива, а также целое неотрицательное число. Гарантируется, что входные данные корректны.

Формат вывода

Программа должна вывести ближайшее число в массиве, меньше заданного (или выбросить исключение, как описано выше), а также значение метрики (с точностью до трех знаков после десятичного разделителя).

Пример 1

Ввод 52 56 54 58
50Вывод Number does not exist.
0.125

Пример 2

Ввод 11 22 66 55 33
44Вывод 33
0.200

Примечания

В систему необходимо сдавать только файл `ChristmasArray.cs`.

```
1 using System;
2 using System.Linq;
3
4 internal class ChristmasArray : BaseArray
5 {
6
7     public ChristmasArray(int[] array) : base(array)
8     {
9     }
10    public override int this[int number]
11    {
12        get
13        {
14            bool flag = false;
15
16            int[] arrayNew = new int[array.Length];
17
18            int k = 0;
19
20            for (int i = 0; i < array.Length; i++)
21            {
22                if (array[i] < number)
23                {
24                    arrayNew[k++] = array[i];
25                    flag = true;
26                }
27            }
28
29            if (flag == false)
30            {
31                throw new ArgumentException("Number does not exist.");
32            }
33
34            Array.Sort(arrayNew);
35            return arrayNew[arrayNew.Length - 1];
36        }
37    }
38 }
```