

Контест 10

11 янв 2022, 18:21:48

старт: 16 мар 2021, 12:30:00

финиш: 22 мар 2021, 23:59:59

длительность: 6д. 11ч.

начало: 16 мар 2021, 12:30:00

конец: 22 мар 2021, 23:59:59

C. Change Letters

| | |
|---------------------|------------|
| Ограничение времени | 2 секунды |
| Ограничение памяти | 64Mb |
| Ввод | input.txt |
| Вывод | output.txt |

Мальчик Петя очень любит все сокращать. Проанализировав некоторое количество текстов, он заметил, что есть буквы, которые встречаются очень редко. Чтобы сократить используемый алфавит он решил заменять все вхождения наиболее редко встречающейся в тексте буквы самой популярной буквой текста. Помогите Пете и напишите программу, которая решает данную задачу. Необходимо учитывать, что строчное и заглавное написание символа считается одной буквой.

Формат ввода

На вход подается файл `input.txt`, содержащий строки, состоящие из латинских букв, цифр и знаков препинания.

Формат вывода

Файл `output.txt`, содержащий исходный текст с заменой самой редко встречающейся буквы на самую часто встречающуюся. Если самых редко/часто встречающихся букв несколько, выбирается первый такой символ по порядку в тексте.

Пример 1

| | |
|---|---|
| Ввод | Вывод |
| <div>It was the night before Christmas.</div> | <div>It tas the night before Christmas.</div> |

Пример 2

| | |
|--|--|
| Ввод | Вывод |
| <div>The world is closing in And did you ever think? That we could be so close? Like brothers The future's in the air I can feel it everywhere Blowing with the wind of change</div> | <div>The world is closing in And did eou ever think? That we could be so close? Like brothers The future's in the air I can feel it everewhere Blowing with the wind of change</div> |

Примечания

В систему необходимо сдавать только файл `ReadWriter.cs` или его содержимое.
Кодировка входного файла – UTF-8.

```
1 using System;
2 using System.IO;
3 using System.Text;
4 using System.Collections.Generic;
5 using System.Linq;
6
7 public class ReadWriter
8 {
9     public static Tuple<char, char> GetMostAndLeastCommonLetters(string path)
10    {
11        Dictionary<char, int> dict = new Dictionary<char, int>();
12
13        bool flag = false;
14
15        string[] data = File.ReadAllLines(path, Encoding.UTF8);
16
17        for (int i = 0; i < data.Length; i++)
18        {
19            string[] current = data[i].ToLower().Split(' ', StringSplitOptions.RemoveEmptyEntries);
20
21            for (int j = 0; j < current.Length; j++)
22            {
23                for (int z = 0; z < current[j].Length; z++)
24                {
25                    if (current[j][z] >= 'a' && current[j][z] <= 'z')
26                    {
27                        flag = true;
28
29                        if (!dict.ContainsKey(current[j][z]))
30                        {
31                            dict.Add(current[j][z], 1);
32                        }
33                        else
34                        {
35                            dict[current[j][z]]++;
36                        }
37                    }
38                }
39            }
40        }
41    }
42 }
```