

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ
ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«ПЕРМСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

ЭЛЕКТРОТЕХНИЧЕСКИЙ ФАКУЛЬТЕТ

КАФЕДРА «ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ И
АВТОМАТИЗИРОВАННЫХ СИСТЕМ»

ОТЧЁТ

«ЛАБОРАТОРНАЯ №9: ОБРАБОТКА ФАЙЛА»

Дисциплина: «Программирование»

Выполнил:

Студент группы ИВТ-21-26

Безух Владимир Сергеевич

Проверил:

Доцент кафедры ИТАС

Полякова Ольга Андреевна

Пермь, 2022

Содержание

1.	Постановка задачи	3
2.	Анализ задачи.....	4
3.	Описание переменных.....	5
4.	Исходный код.....	6
5.	Анализ результатов.....	8

1. Постановка задачи

Скопировать из файла в файл все строки, в которых содержится два одинаковых слова.

Определить номер слова, в котором больше всего символов «а».

2. Анализ задачи

Для поиска строки с двумя одинаковыми словами нужно предварительно отсортировать каждую строку. Затем проверить соответствующие условия.

Поиск слова с наибольшим количеством символов «а» — тривиальная задача.

3. Описание переменных

std::vector<std::vector<std::string>> strings; — для хранения слов и строк.

std::ifstream f_input("FileInput.txt"); — для использования файла с ВХОДНЫМИ данными.

std::ofstream f_output("FileOutput.txt"); — для использования файла с ВЫХОДНЫМИ данными.

4. Исходный код

```
#include<vector>
#include<cstring>
#include<fstream>
#include<sstream>
#include<iostream>
#include<algorithm>

int main()
{
    std::vector<std::vector<std::string>> strings;
    std::ifstream f_input("FileInput.txt");
    std::ofstream f_output("FileOutput.txt");

    // input
    while (!f_input.eof()) {
        std::string word, temp;
        std::vector<std::string> words;

        getline(f_input, temp);
        std::stringstream string(temp);

        while (string >> word)
            words.emplace_back(word);

        strings.emplace_back(words);
    }

    // first task
    std::vector<std::vector<std::string>> firstTaskResult;
    for (const auto& words : strings) {
        std::vector<std::string> temp = words;
        std::sort(temp.begin(), temp.end());

        bool flag_two_equal_words = false;
        auto it = temp.begin(); std::string prev_word = *it; ++it;
        while (it != temp.end()) {
            if (prev_word == *it) {
                if (!flag_two_equal_words)
                    flag_two_equal_words = true;
                else {
                    flag_two_equal_words = false;
                    break;
                }
            }

            prev_word = *it;
            ++it;
        }

        if (flag_two_equal_words)
            firstTaskResult.emplace_back(words);
    }

    for (const auto& words : firstTaskResult) {
        for (const auto& word : words)
            f_output << word << ' ';

        f_output << '\n';
    }
}
```

```

// second task
size_t max = 0, max_number = 1, current_number = 1;
for (const auto& words : strings) {
    for (const auto& word : words) {
        size_t counter = 0;
        for (const auto& symbol : word)
            if (symbol == 'a')
                ++counter;

        if (counter > max)
        {
            max = counter;
            max_number = current_number;
        }

        ++current_number;
    }
}

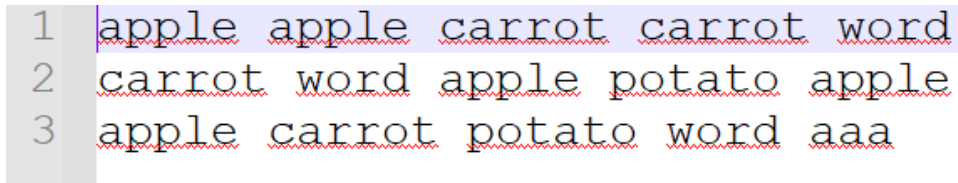
f_output << max_number;

f_input.close();
f_output.close();
}

```

5. Анализ результатов

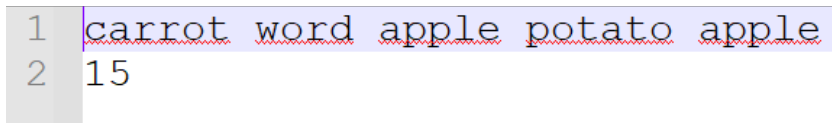
Исходный файл (рис. 1).



```
1 apple apple carrot carrot word
2 carrot word apple potato apple
3 apple carrot potato word aaa
```

Рисунок 1 — Строки и слова

Результирующий файл (рис. 2).



```
1 carrot word apple potato apple
2 15
```

Рисунок 2 — Слова нумеруются с единицы