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-21-2б

Безух Владимир Сергеевич

Проверил:

Доцент кафедры ИТАС

Полякова Ольга Андреевна

Пермь, 2022

Содержание

[1. Постановка задачи 3](#_Toc97827178)

[2. Анализ задачи 4](#_Toc97827179)

[3. Описание переменных 5](#_Toc97827180)

[4. Исходный код 6](#_Toc97827181)

[5. Анализ результатов 8](#_Toc97827182)

# Постановка задачи

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

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

# 

# Анализ задачи

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

Поиск слова с наибольшим количество символов «a» — тривиальная задача.

# 

# 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.

# 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();

}

# Анализ результатов

Исходный файл (рис. 1).

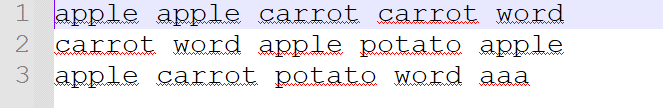


Рисунок 1 — Строки и слова

Результирующий файл (рис. 2).

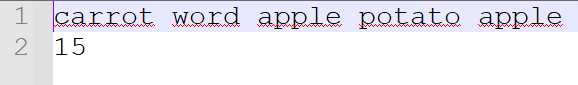


Рисунок 2 — Слова нумеруются с единицы