Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования «Тихоокеанский государственный университет»

Высшая школа кибернетики и цифровых технологий

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №6

ПО КУРСУ «ПРОГРАММИРОВАНИЕ»

«РАБОТА СО СТРОКАМИ»

Выполнил: студент 2 курса группы ПО(б)-31

Корейбо Дмитрий Алексеевич

Проверил: ассистент ВШ КЦТ

Крылов Владимир Андреевич

Хабаровск 2024 г.

Цель работы: изучить теоретический материал по лабораторной работе и реализовать на языке программирования C++ программы для решения заданий.

Задание: необходимо изучить теоретический материал по работе со строками и реализовать несколько программ на языке программирования C++, чтобы получить практический опыт работы с этими концепциями.

Ход работы:

1. работа с программой “Visual Studio” для созданий программ со строками.

Заголовочный файл fstream в стандартной библиотеке C++ определяет базовые типы для чтения и записи файлов.

Include <string> в C++ — это подключение заголовочного файла, необходимого для использования объектов класса string.

Класс string стандартной библиотеки С++ реализует различные операции со строками. Например, инициализацию, копирование, доступ к отдельным символам и другие.

Заголовок cctype в C++ содержит прототипы функций, которые используются для преобразования символов в верхний и нижний регистр, а также для проверки, является ли символ цифрой.

isVowel() — это функция в C++, которая принимает символ в качестве аргумента и проверяет, соответствует ли он любому из символов гласных («a», «e», «i», «o», «u») в верхнем и нижнем регистре.

Modifytext — это функция на языке программирования C++, которая позволяет изменять текст.

Toupper в C++ — это функция, которая преобразует строчные буквы в прописные.

Функция ispunct в C++ проверяет, является ли аргумент символом пунктуации.

countPunctuation — это функция для подсчёта количества знаков пунктуации в строке.

Getline() — это встроенная функция в C++, которая позволяет выводить в консоль все вводимые с клавиатуры символы.

Push\_back в C++ — это функция, которая добавляет элемент в конец вектора.

1. составление с помощью данной программы для первого задания (листинг 1).

Листинг 1 — код файла Visual Studio

#include <iostream>

#include <fstream>

#include <string>

#include <vector>

#include <cctype>

bool isVowel(char c) {

return c == 'a' || c == 'e' || c == 'i' || c == 'o' || c == 'u' ||

c == 'A' || c == 'E' || c == 'I' || c == 'O' || c == 'U';

}

std::string modifyText(const std::string& text) {

std::string modifiedText;

std::size\_t g = 0, end;

while ((end = text.find\_first\_of(" ", g)) != std::string::npos) {

std::string word = text.substr(g, end - g);

if (isVowel(word[0])) {

word[0] = toupper(word[0]);

}

modifiedText += word + " ";

g = end + 1;

}

std::string lastWord = text.substr(g);

if (isVowel(lastWord[0])) {

lastWord[0] = toupper(lastWord[0]);

}

modifiedText += lastWord;

return modifiedText;

}

int main() {

setlocale(LC\_ALL, "RUS");

std::ifstream inputFile("input.txt");

std::string line;

if (!inputFile.is\_open()) {

std::cerr << "Не удалось открыть файл." << std::endl;

return 1;

}

while (std::getline(inputFile, line)) {

std::string result = modifyText(line);

std::cout << result << std::endl;

}

inputFile.close();

return 0;

}

Результат работы программы первого задания (рисунок 1).

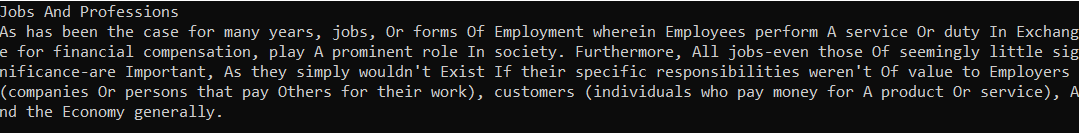


Рисунок 1 – результат работы программы первого задания

1. составление с помощью данной программы для второго задания (листинг 2).

Листинг 2 — код файла Visual Studio

#include <iostream>

#include <fstream>

#include <string>

#include <vector>

#include <cctype>

#include <iostream>

#include <fstream>

#include <string>

#include <vector>

#include <cctype>

bool isPunctuation(char ch) {

return std::ispunct(ch);

}

std::vector<std::string> splitIntoSentences(const std::string& text) {

std::vector<std::string> sentences;

std::string currentSentence;

for (char ch : text) {

currentSentence += ch;

if (ch == '.' || ch == '!' || ch == '?') {

sentences.push\_back(currentSentence);

currentSentence.clear();

}

}

return sentences;

}

int countPunctuation(const std::string& sentence) {

int count = 0;

for (char ch : sentence) {

if (isPunctuation(ch)) {

++count;

}

}

return count;

}

int main() {

setlocale(LC\_ALL, "RUS");

std::ifstream inputFile("input.txt");

if (!inputFile.is\_open()) {

std::cerr << "Не удалось открыть файл!" << std::endl;

return 1;

}

std::string text((std::istreambuf\_iterator<char>(inputFile)), std::istreambuf\_iterator<char>());

inputFile.close();

std::vector<std::string> sentences = splitIntoSentences(text);

int maxPunctuationCount = 0;

std::vector<std::string> resultSentences;

for (const auto& sentence : sentences) {

int punctuationCount = countPunctuation(sentence);

if (punctuationCount > maxPunctuationCount) {

maxPunctuationCount = punctuationCount;

resultSentences.clear();

resultSentences.push\_back(sentence);

}

else if (punctuationCount == maxPunctuationCount && punctuationCount > 0) {

resultSentences.push\_back(sentence);

}

}

std::cout << "Предложения с максимальным количеством знаков пунктуации:" << std::endl;

for (const auto& sentence : resultSentences) {

std::cout << sentence << std::endl;

}

return 0;

Результат работы программы второго задания (рисунок 2).

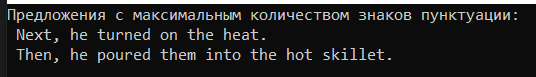


Рисунок 2 – результат работы программы второго задания

Вывод: в ходе выполнения лабораторной работы мы изучили как работать со строками на языке программирования C++. C++ остается актуальным и востребованным языком программирования благодаря своей мощности, универсальности и эффективности.