Пермский национальный исследовательский политехнический университет

Кафедра информационные технологии и автоматизированные системы

Лабораторная работа №6

“**Строки**”

|  |  |  |
| --- | --- | --- |
|  |  | Выполнил:  студент группы: ИВТ-24-2б  К.Д. Горбушин  Проверил:  доцент кафедры ИТАС  О.А. Полякова |

Пермь 2025

# Постановка задачи

Задана строка, состоящая из символов. Символы объединяются в слова. Слова друг от друга отделяются одним или несколькими пробелами. В конце текста ставится

точка. Текст содержит не более 255 символов. Выполнить ввод строки, используя функцию Gets(s) и напечатать все слова которые содержат по одной цифре

**Код программы**

#include <iostream>

#include <stdio.h> // Для fgets

#include <ctype.h> // Для isdigit, isspace

using namespace std;

int main() {

system("cls");

system("chcp 1251");

setlocale(LC\_ALL, "Ru");

char text[256]; // Текст не более 255 символов + 1 для нуля

char current\_word[256]; // Слово не более 255 символов + 1 для нуля

int current\_word\_index = 0;

cout << "Введите текст (не более 255 символов, заканчивается точкой):\n";

fgets(text, sizeof(text), stdin); // Используем fgets

// Удаляем символ новой строки, если он есть

int text\_length = 0;

while (text[text\_length] != '\0') {

text\_length++;

}

if (text\_length > 0 && text[text\_length - 1] == '\n') {

text[text\_length - 1] = '\0';

text\_length--;

}

// Проверка, что текст заканчивается точкой

if (text\_length == 0 || text[text\_length - 1] != '.') {

cerr << "Ошибка: Текст должен заканчиваться точкой." << endl;

return 1;

}

cout << "Слова, содержащие ровно одну цифру:\n";

for (int i = 0; text[i] != '\0'; ++i) {

if (!isspace(text[i]) && text[i] != '.') {

// Символ является частью слова

current\_word[current\_word\_index++] = text[i];

}

else {

// Символ - разделитель (пробел или точка)

if (current\_word\_index > 0) {

// Есть слово для обработки

current\_word[current\_word\_index] = '\0'; // Завершаем слово нулем

int digit\_count = 0;

for (int j = 0; current\_word[j] != '\0'; ++j) {

if (isdigit(current\_word[j])) {

digit\_count++;

}

}

if (digit\_count == 1) {

cout << current\_word << endl;

}

// Сбрасываем для следующего слова

current\_word\_index = 0;

}

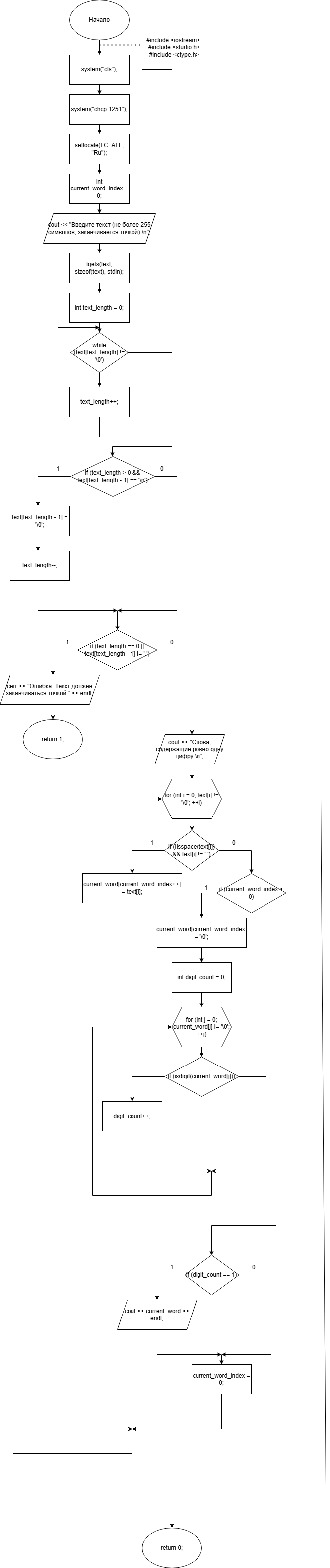
}

}

return 0;

}

**Блок-схема**

****

**Ссылка на GITHUB:** [https://github.com/kirill2209](https://github.com/kirill2209%20)