**Министерство науки и высшего образования Российской Федерации**Федеральное государственное бюджетное образовательное учреждение высшего образования

**«Владимирский государственный университет**

**имени Александра Григорьевича и Николая Григорьевича Столетовых»**

**(ВлГУ)**

Институт прикладной математики, физики и информатики

Кафедра физики и прикладной математики

**ЛАБОРАТОРНАЯ РАБОТА №9**

по дисциплине  
«Основы программирования»  
на тему:  
**«Строки. Алгоритмы поиска подстроки в строке»**

Выполнил:

ст. гр. ПМИ-123

Рушев А.М.

Принял:  
ст. преподаватель

каф. ФиПМ

Черников А.С.

Владимир, 2023 г.

**Цель работы:** освоение способов работы со строками в языке программирования С++. Изучение алгоритмов поиска подстроки в строке. **Задание:**

1. Создать строку как массив символов. Проинициализировать этот массив при объявлении. Выполнить сортировку элементов массива по алфавиту. Вывести на экран отсортированный массив.
2. Объявить массив из десяти строк. Проинициализировать элементы массива вводом с клавиатуры. Выполнить сортировку элементов массива в алфавитном порядке. Вывести на экран отсортированный массив.
3. Объявить и проинициализировать три строки, таким образом, чтобы из них можно было составить осмысленную фразу, каждую строку начинать с прописной буквы, заканчивать пробелом. Объединить строки в одну. Удалить из полученной строки последний пробел, заменив его на соответствующий знак препинания. Первую буквы полученной строки сделать заглавной. Вывести на экран полученный результат.
4. Реализовать алгоритм прямого поиска.

**Практическая часть:**

**Задание 1.**

Листинг программы:

void task1() {

/\*

Создать строку как массив символов. Проинициализировать этот массив при объявлении.

Выполнить сортировку элементов массива по алфавиту. Вывести на экран отсортированный массив.

\*/

cout << "\nЗадание №1" << endl;

char str[10] = { "dcbvaanm" };

cout << "Исходный массив: " << str << endl;

//сортировка по алфавиту

for (int i = 0; i < sizeof(str) - 1; i++) {

for (int j = sizeof(str) - 1; j > 0; j--) {

if (str[j - 1] > str[j]) {

char temp = str[j - 1];

str[j - 1] = str[j];

str[j] = temp;

}

}

}

cout << "Отсортированный массив: ";

for (int i = 0; i < sizeof(str);i++) {

cout << str[i];

}

}

Результат выполнения кода:

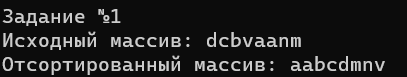


Рисунок 1 – Результат выполнения кода задания 1

Задание 2.

Листинг программы:

void task2() {

/\*

Объявить массив из десяти строк. Проинициализировать элементы массива вводом с клавиатуры.

Выполнить сортировку элементов массива в алфавитном порядке. Вывести на экран отсортированный массив.

\*/

cout << "\nЗадание №2" << endl;

const int n = 10;

string arrStr[n];

//заполнение строки

for (int i = 0; i < n; i++) {

cout << "Заполните строку: ";

cin >> arrStr[i];

}

cout << "Исходный массив: " << endl;

for (int i = 0; i < n; i++) {

cout << arrStr[i] << endl;

}

//сортировка по алфавиту

for (int i = 0; i < n - 1; i++) {

for (int j = 0; j < n - 1 - i; j++) {

if (arrStr[j] > arrStr[j + 1]) {

string temp = arrStr[j];

arrStr[j] = arrStr[j + 1];

arrStr[j + 1] = temp;

}

}

}

cout << "Отсортированный массив: " << endl;

for (int i = 0; i < n; i++) {

cout << arrStr[i] << endl;

}

}

Результат выполнения кода:

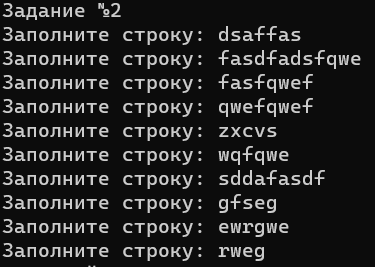


Рисунок 2 – Заполнение массива строк

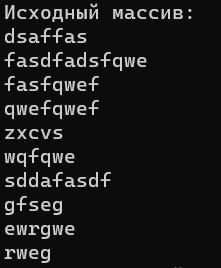


Рисунок 3 – Исходный массив

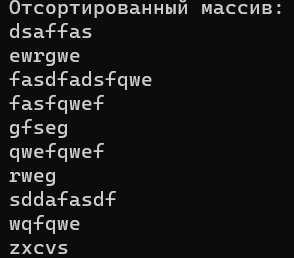


Рисунок 4 – Отсортированный массив по алфавиту

Задание 3.

Листинг программы:

void task3() {

/\*

Объявить и проинициализировать три строки, таким образом, чтобы из них можно было составить осмысленную фразу,

каждую строку начинать с прописной буквы, заканчивать пробелом. Объединить строки в одну. Удалить из полученной строки последний пробел,

заменив его на соответствующий знак препинания. Первую буквы полученной строки сделать заглавной. Вывести на экран полученный результат.

\*/

cout << "\nЗадание №3" << endl;

char str1[] = "Hello ";

char str2[] = "World ";

char str3[] = "People ";

const int k = sizeof(str1) + sizeof(str2) + sizeof(str3) + 1;

char strFull[k];

strcpy\_s(strFull, str1);

strcat\_s(strFull, str2);

strcat\_s(strFull, str3);

//переход в нижний регистр во всем предложении

for (int i = 0; i < sizeof(strFull); i++) {

strFull[i] = tolower(strFull[i]);

}

//замена последнего символа с пробела на знак препинания

if (strFull[strlen(strFull) - 1] == ' ') {

strFull[strlen(strFull) - 1] = '.';

}

else cout << "В конце предложения нет пробела." << endl;

//проверка на заглавную букву первого символа

if (strFull[0] == tolower(strFull[0])) {

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

}

cout << strFull << endl;

}

Результат выполнения кода:

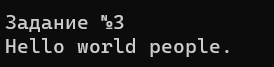


Рисунок 5 – Результат выполнения кода задания 3

Задание 4.

Листинг программы:

void task4() {

/\*

Реализовать алгоритм прямого поиска.

\*/

cout << "\nЗадание №4" << endl;

char str1[50] = "HelloworldPeoplewowowowowowowowowoowwwowowow";

char str2[50] = "wo";

int number = 0;

for (int i = 0; i < strlen(str1) - strlen(str2); i++) {

int j = 0;

while ((j < strlen(str2) && str2[j] == str1[i + j])) {

j = j + 1;

}

if (j == strlen(str2)) {

cout << "Слово: " << str2 << " появилось на позиции: " << i << endl;

number++;

}

}

cout << "Слово встретилось: " << number << " раз(а)" << endl;

}

Результат выполнения кода:

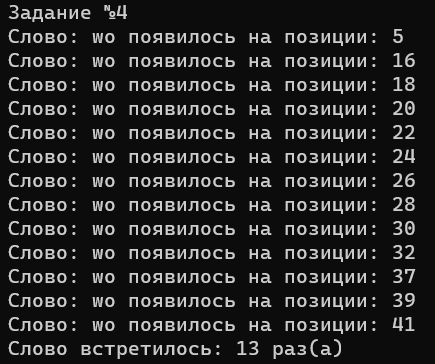


Рисунок 6 – Результат выполнения кода задания 4

Листинг программы:

#include <iostream>

using namespace std;

int main()

{

setlocale(LC\_ALL, "rus");

cout << "Лабораторная работа №9." << endl;

task1();

task2();

task3();

task4();

system("pause");

return 0;

}

**Вывод:** были закреплены навыки работы со строками и их функциями, также был изучен алгоритм поиска в языке программирования C++.