

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования «Тульский государственный университет»

КАФЕДРА ИНФОРМАЦИОННОЙ БЕЗОПАСНОСТИ

СОРТИРОВКА ВСТАВКАМИ

отчет о
лабораторной работе №4

по дисциплине
ТЕХНОЛОГИИ И МЕТОДЫ ПРОГРАММИРОВАНИЯ

ВАРИАНТ 9

Выполнила:	ст. гр. 230711	Павлова В.С.
Проверил:	асс. каф. ИБ	Курбаков М.Ю.

Тула, 2022 г.

ЦЕЛЬ И ЗАДАЧА РАБОТЫ

Цель: изучить принципы сортировки массивов методом включений (методом вставки).

Задача: в данной работе требуется написать программу, демонстрирующую использование изученных принципов.

ЗАДАНИЕ НА РАБОТУ

Задание из варианта 9: отсортировать по возрастанию массив, состоящий букв вашего имени и фамилии, считая, что а = 1, б = 2, в = 3, ... я = 33.

СХЕМА ПРОГРАММЫ

Схема алгоритма для сортировки входной последовательности вставками в лексикографическом порядке (а = 1, б = 2, в = 3, ... я = 33), представлена на рисунке 1.

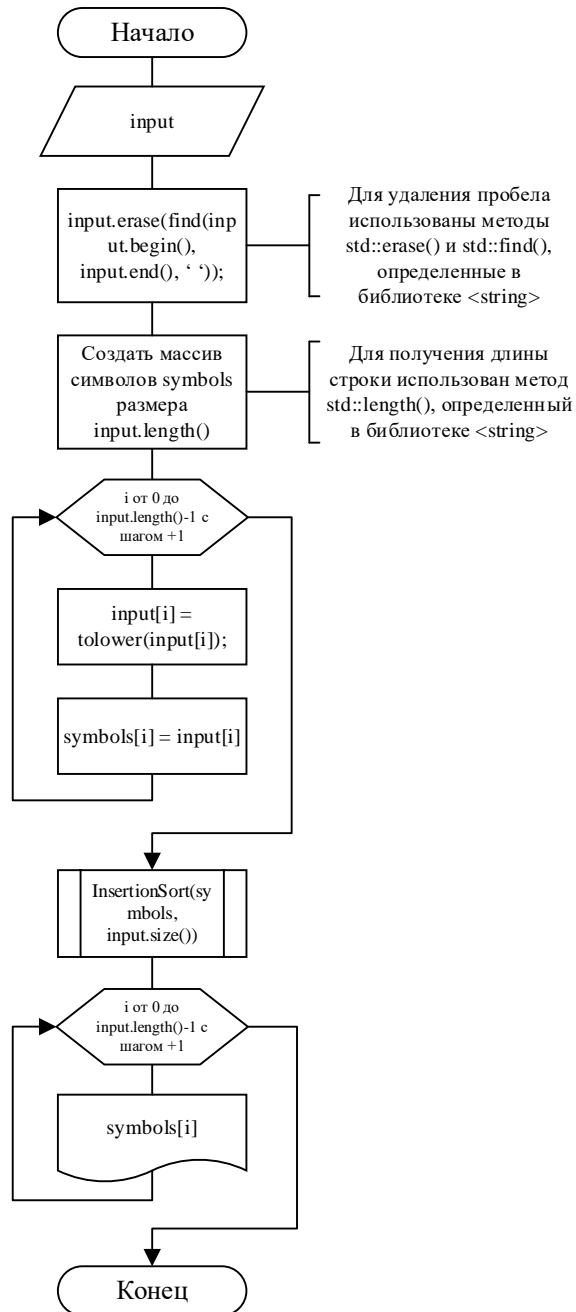


Рисунок 1 – Схема алгоритма сортировки массива вставками

Схема алгоритма сортировки вставками представлена на рисунке 2.

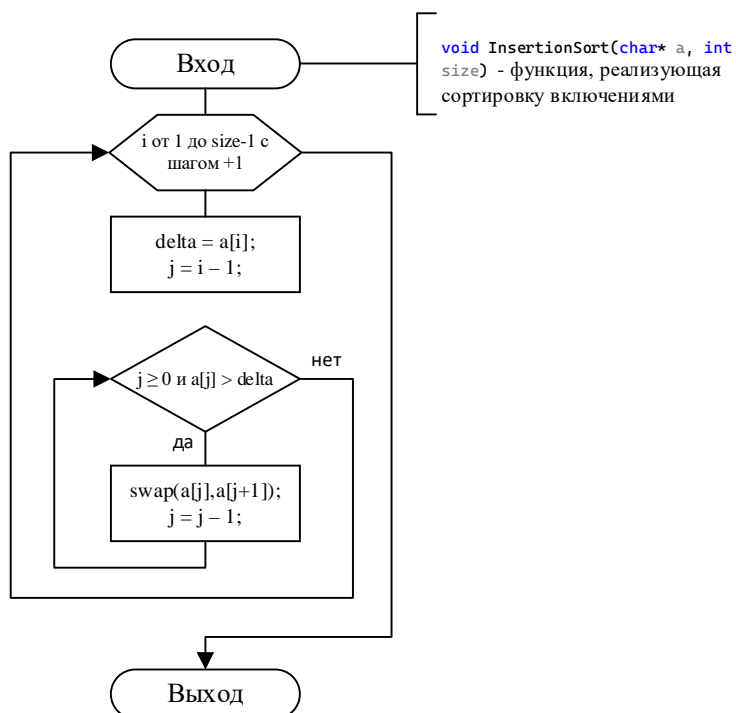


Рисунок 2 – Схема алгоритма сортировки

ТЕКСТ ПРОГРАММЫ

Текст программы на языке программирования C++ для сортировки массива в лексикографическом порядке представлен в листинге 1.

Листинг 1. Текст программы

```

#include <iostream>
#include <string>
#include <windows.h>
void InsertionSort(char* a, int size)
{
    char delta;
    int j;
    for (int i = 1; i < size; i++)
    {
        delta = a[i];
        j = i - 1;
        while (j >= 0 && a[j] > delta)
        {
            std::swap(a[j], a[j + 1]);
            j--;
        }
    }
    return;
}
int main()
{
    setlocale(LC_ALL, "Russian");
    SetConsoleCP(1251);
    SetConsoleOutputCP(1251);
}
  
```

Листинг 1. Текст программы (продолжение)

```
std::string input;
std::cout << "Введите ваше имя и фамилию: ";
std::getline(std::cin >> std::ws, input);
input.erase(std::find(input.begin(), input.end(), ' '));
char* symbols = new char[input.length()];
for (int i = 0; i < input.length(); i++)
{
    input[i] = tolower(input[i]);
    symbols[i] = input[i];
}

InsertionSort(symbols, input.size());
std::cout << "Последовательность после сортировки: ";
for (int i = 0; i < input.length(); i++)
    std::cout << symbols[i];

return 0;
}
```

ИНСТРУКЦИЯ ПОЛЬЗОВАТЕЛЯ

Данная программа предназначена для сортировки массива в лексикографическом порядке. При запуске программы пользователю предлагается ввести строку со своим именем и фамилией. После программа преобразовывает строку к нижнему регистру, убирает пробелы и сортирует массив методом вставки, а после выводит в консоль уже отсортированную последовательность символов.

Сортировка осуществляется следующим образом: массив разбивается на две области: упорядоченную и неупорядоченную. Изначально весь массив является неупорядоченной областью, кроме первого элемента. При первом проходе первый элемент из неупорядоченной области изымается и помещается в правильное положение в упорядоченной области. На каждом проходе размер упорядоченной области возрастает на 1, а размер неупорядоченной области сокращается на 1.

ИНСТРУКЦИЯ ПРОГРАММИСТА

Данная программа предназначена сортировки массива в лексикографическом порядке. Структуры данных, используемые в программе, приведены в таблице 1.

Таблица 1 – Структуры данных в программе

Имя	Тип (класс)	Предназначение
input	string	Входная строка
symbols	char*	Сортируемый массив символов

В программе имеется следующая функция:

`void InsertionSort(char* a, int size)` – функция, реализующая сортировку.

Таблица 2 – Структуры данных, используемые в подпрограмме InsertionSort

Имя	Тип	Предназначение
<i>формальные параметры</i>		
a	char*	Ссылка на массив для сортировки
size	int	Размер массива
<i>локальные переменные</i>		
delta	char	Отсортированный элемент
j	int	Индекс левой границы неупорядоченной области

ДЕМОНСТРАЦИОННЫЙ ПРИМЕР

Сперва отсортируем входную последовательность вручную. Согласно заданию варианта, заданный массив должен быть отсортирован по возрастанию с учётом того, что символ «а» = 1, «б» = 2, «в» = 3, ... «я» = 33. Пусть имеется входная последовательность **«Павлова Виктория»**. Тогда после форматирования к нижнему регистру, удаления пробелов и сортировки она должна выглядеть следующим образом **«аавввиикклоопртя»**.

Теперь проверим результат работы программы на соответствие полученной последовательности. Он представлен на рисунке 3. Как видно по рисунку, последовательность отсортирована верно.

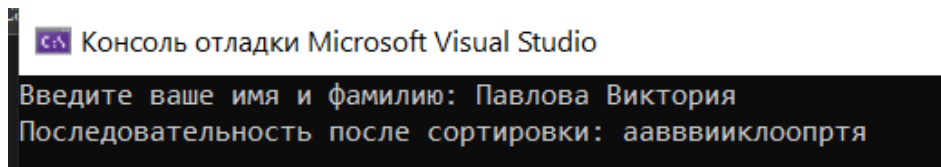


Рисунок 3 – Результат работы программы

ВЫВОДЫ

В ходе данной лабораторной работы был изучен принцип работы сортировки вставками. Для демонстрации полученных знаний была написана программа для сортировки указанным методом, результат работы которой был проверен аналитически. По результатам проверки можно сделать вывод о том, что программа работает корректно.