Министерство науки и высшего образования Российской Федерации Федеральное государственное автономное образовательное учреждение высшего образования

**«Пермский национальный исследовательский политехнический университет»**

Электротехнический факультет

Кафедра «Информационные технологии и автоматизированные системы»

Направление подготовки: 09.03.01 – «Информатика и вычислительная техника»

**Лабораторная работа № 1**

по дисциплине

**«Основы алгоритмизации и программирования»**

Выполнил студент гр. ИВТ-21-1б

Мифтахов Ильяс Гарифжанович

Проверил:

ст. преп. каф. ИТАС

Яруллин Денис Владимирович

(оценка) (подпись)

(дата)

**Постановка задачи**

Написать перегруженные функции и основную программу, вызывающую их.

**Задание**

а) для массива целых чисел найти минимальный элемент;

б) для строки найти длину самого короткого слова.

**Основная часть**

Текст программы представлен на рисунке 1.

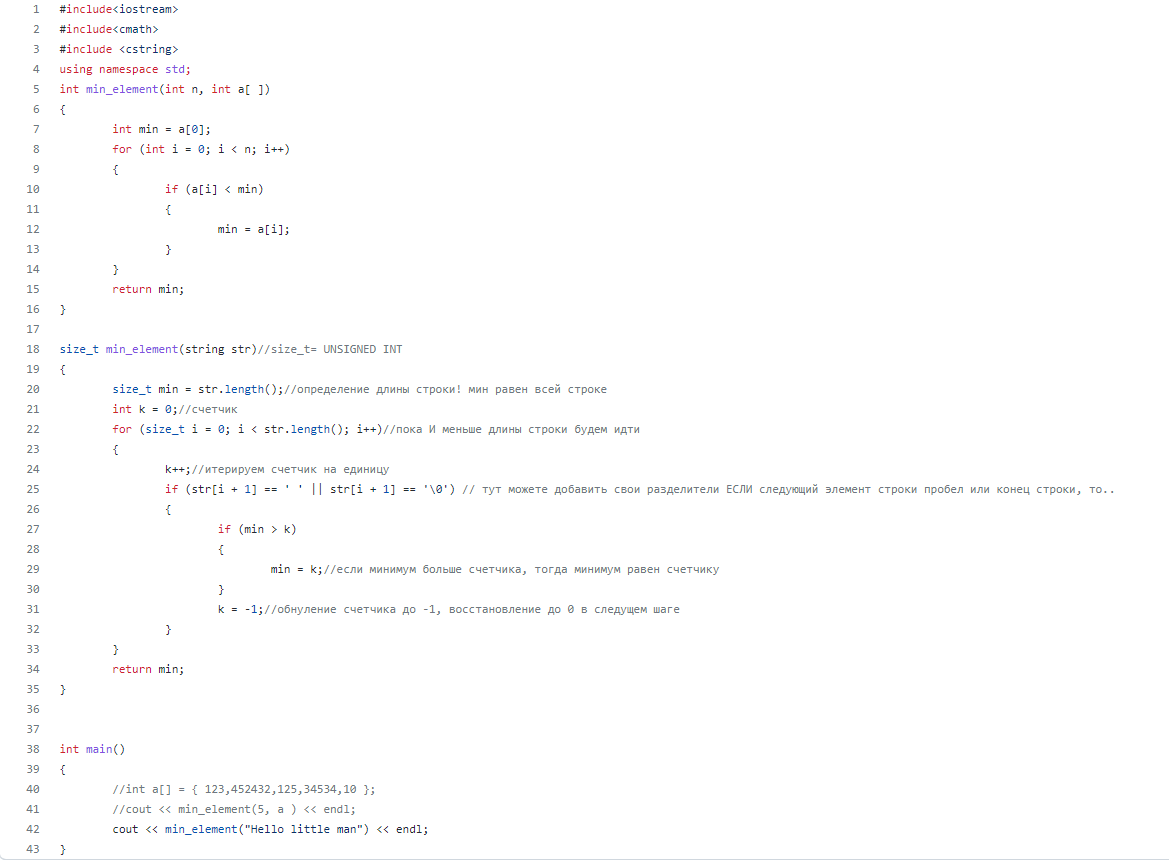


Рисунок 1-текст программы

Для корректной работы программы со строкой была подключена дополнительная директива препроцессора – «cstring»

Итак, для начала были подключены стандартные директивы препроцессора, указано пространство «std». Далее, идет первая функция «min\_element» с целочисленным типом данных(integer), которая принимает в себя параметр n – количество чисел в массиве и сам массив a[], далее, пусть минимальным элементов будет элемент с индексом 0(номер 1), цикл for будет исполнен ровно столько раз, сколько в массиве указано элементов, начиная с элемента с нулевым индексом(номером 1). Далее, встречаем условие, если элемент на данном проходе цикла меньше минимального(первого), то минимальным становится уже этот элемент, по окончании проходов выходим из цикла и функция возвращает значение минимального элемента. Вызов этой функции из главной требует указания в скобках количества элементов массива и указания самих элементов массива.

Также, можем перегрузить нашу функцию «min\_element» с помощью вызова из главной этой функции, только вместо прежних параметров необходимо в двойных кавычках написать слова в строке. Этой функции я присвоил тип «size\_t» для комфортной работы и дебага, этот тип гарантирует содержание в себе индексы всех элементов массива. Вводим новую переменную минимума, которой присваиваем значение длины строки, ибо длиной самого большого слова может быть длина самой строки. Вводим переменную, которая будет играть роль счетчика в нашем цикле for. Пока переменная i, равная нулю, увеличивающаяся на 1 с каждым проходом цикла, будет меньше длины строки, цикл будет выполняться. Сразу же увеличиваем счетчик на 1, так как встретили элемент строки, и ЕСЛИ следующий элемент будет пробелом, либо концом строки, тогда мы переходим к еще одному условию – ЕСЛИ минимум больше счетчика, тогда присваиваем значение счетчика минимуму и сбрасываем счетчик до -1, так как при следующем цикле произойдет его увеличение на 1 и он вновь будет равен 0. Возвращаем также длину самого короткого слова в строке.

Блок-схема данной программы представлена ниже, на рисунке 2.

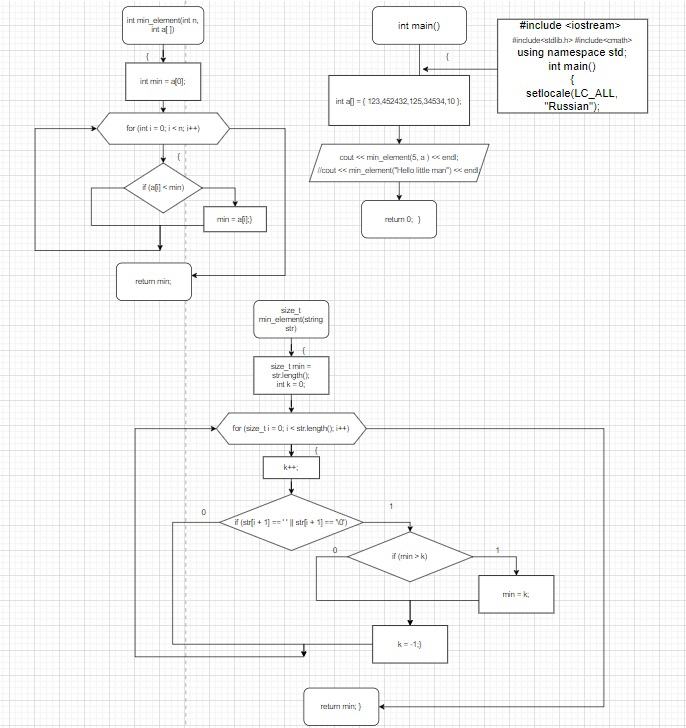


Рисунок 2- блок-схема

На рисунке 3 представлен результат выполнения программы для решения задачи в пункте «б)»

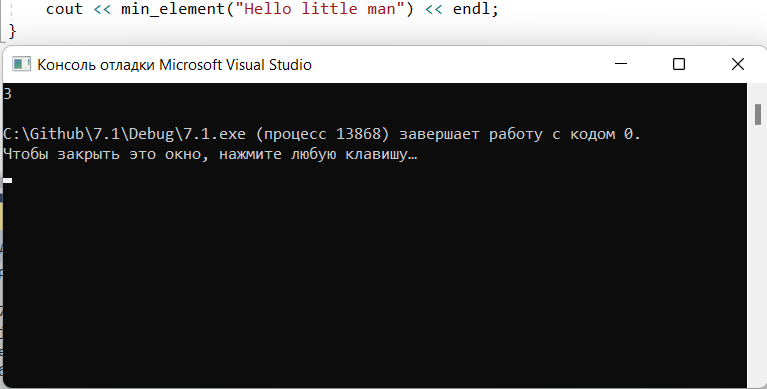


Рисунок 3- результат выполнения программы

На рисунке 4 представлен результат выполнения программы для решения задачи в пункте «а)»

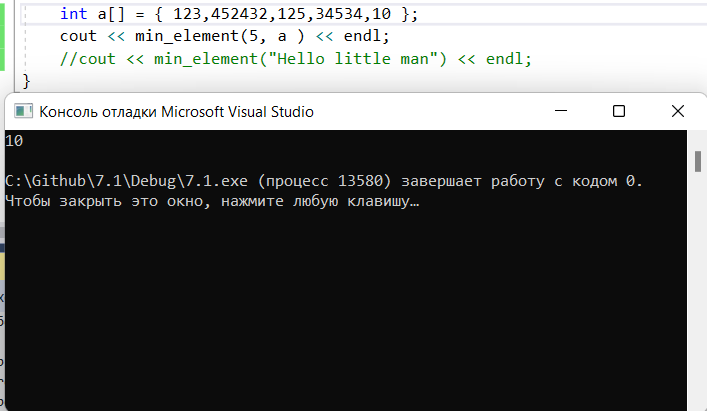


Рисунок 4- результат выполнения программы

**Вывод**

В лабораторной работе №1 поставленная задача выполнена, перегрузка функции реализована успешно на основе заданий варианта 19. Вывод на экран корректных значений.