|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  | | --- | --- | --- | |  |  |  | | МИНОБРНАУКИ РОССИИ | | | | Федеральное государственное бюджетное образовательное учреждение  высшего образования  **«МИРЭА – Российский технологический университет»**  **РТУ МИРЭА** | | |   Институт информационных технологий |
| Кафедра вычислительной техники |
|  |

|  |  |
| --- | --- |
| **ОТЧЕТ ПО ПРАКТИЧЕСКОЙ РАБОТЕ №6** | |
| **по дисциплине** | |
| **«Алгоритмические основы обработки данных»** | |
|  | |
| Выполнил студент группы ИВБО-11-23  *(учебная группа)* | Туктаров Т.А. |
| Принял старший преподаватель | Асадова Ю.С. |

|  |  |  |
| --- | --- | --- |
| Практическая работа выполнена | «29» сентября 2024г. | *\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*  *(подпись студента)* |
|  |  |  |
| «Зачтено» | «29» сентября 2024г. | *\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*  *(подпись руководителя)* |

|  |  |  |
| --- | --- | --- |
|  |  |  |
| МИНОБРНАУКИ РОССИИ | | |
| Федеральное государственное бюджетное образовательное учреждение  высшего образования  **«МИРЭА – Российский технологический университет»**  **РТУ МИРЭА** | | |

Институт информационных технологий

Кафедра вычислительной техники

Выполнено \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/Т.А. Туктаров/

Зачтено \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/Ю.С. Асадова/

**Задание на практическую работу №6**

Дисциплина: «Алгоритмические основы обработки данных»

Студент Туктаров Тимур Азатович Шифр 23И0087 Группа ИВБО-11-23

**1. Тема**: «использование библиотечных функций для обработки текста».

**2. Срок сдачи студентом законченной работы:** 14.10.2024.

**3. Исходные данные:** целочисленный динамический массив.

**4. Задание:**

Разработать программу, выполняющую операции над целочисленным динамическим массивом из n элементов (n<=100):

* ввод массива;
* вывод массива на экран;
* Поиск номера последнего отрицательного элемента
* Удаление числа после последнего максимального числа

**5. Содержание отчета:**

* титульный лист;
* задание;
* оглавление;
* введение;
* основные разделы отчета;
* заключение;
* список использованных источников;

Руководитель работы Ю.С. Асадова \_\_\_\_\_\_\_\_\_\_\_\_ «14» октября 2024г.

подпись

Задание принял к исполнению Т.А Туктаров \_\_\_\_\_\_\_\_\_\_\_\_ «14» октября 2024г.

подпись

**СОДЕРЖАНИЕ**

[ВВЕДЕНИЕ 4](#_Toc145183808)

[1 ОСНОВНОЙ РАЗДЕЛ 5](#_Toc145183809)

[2 БЛОК-СХЕМА АЛГОРИТМА 6](#_Toc145183810)

[3 ИСХОДНЫЙ КОД 7](#_Toc145183811)

[4 ПРИМЕР РАБОТЫ ПРОГРАММЫ 8](#_Toc145183812)

[ЗАКЛЮЧЕНИЕ 9](#_Toc145183813)

[СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 10](#_Toc145183814)

# ВВЕДЕНИЕ

В данной практической работе требуется применить циклы, работу с динамическим массивами, функциями и условными операторами.

Постановка задачи:

Разработать программу, выполняющую операции над целочисленным динамическим массивом из n элементов (n<=100):

* ввод массива;
* вывод массива на экран;
* поиск номера последнего отрицательного элемента
* удаление числа после последнего максимального числа

# 1 ОСНОВНОЙ РАЗДЕЛ

В начале программы объявляем динамический массив vec, затем запускаем бесконечный цикл. Внутри него выводим инструкцию и описание команд для пользователя. В зависимости от ввода пользователя, программа выполняет одну из 5 операций: считывает массив, выводит массив, удаляет последний элемент после максимального элемента, ищет номер последнего отрицательного элемента.

# 2 БЛОК-СХЕМА АЛГОРИТМА..

Представим описание алгоритма в графическом виде на рисунке 2.2

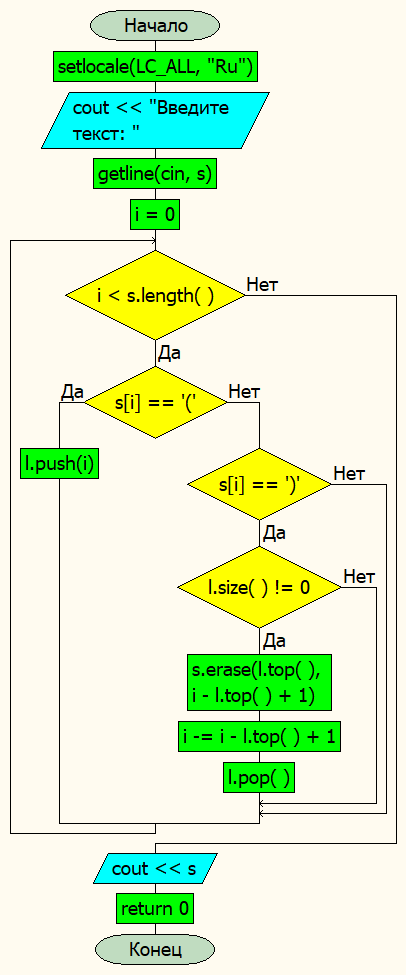


Рисунок 2.1 – Блок – схема алгоритма функции main()

# 3 ИСХОДНЫЙ КОД

Программная реализация алгоритма для решения задачи представлена ниже.

Листинг 3.1 – Исходный код программы

// PR\_6.cpp : Этот файл содержит функцию "main". Здесь начинается и заканчивается выполнение программы.

//

#include <iostream>

#include <vector>

using namespace std;

int find(vector<int>& v) {

for (int i = v.size(); i >= 0; i--)

{

if (i < 0)return i;

}

return -1;

}

void delete\_max(vector<int>& v) {

int mx = v[0];

for (auto a : v) {

if (a > mx)mx = a;

}

int i = 0;

while (i < v.size()) {

if (v[i] == mx) {

v.erase(v.begin() + i);

i--;

}

++i;

}

}

void enter(vector<int>& v) {

while (1) {

int a;

cin >> a;

v.push\_back(a);

}

}

void print\_arr(vector<int>& v) {

for (auto a : v)

{

cout << a << " ";

}

cout << "\n";

}

int main()

{

setlocale(LC\_ALL, "RU");

vector<int>vec;

while (1) {

cout << "Что вы хотите сделать?\n";

cout << "1 - Ввести массив\n2 - найти последний отрицательный элемет\n";

cout << "3 - удалить все элементы перед каждым максимальным числом\n4 - вывести массив";

cout << "0 - выход из программы";

short c;

cin >> c;

switch (c)

{

case 0:

return 0;

break;

case 1:

enter(vec);

break;

case 2:

find(vec);

break;

case 3:

delete\_max(vec);

break;

case 4:

print\_arr(vec);

break;

default:

break;

}

}

}

# 4 ПРИМЕР РАБОТЫ ПРОГРАММЫ

int main()

{

setlocale(LC\_ALL, "RU");

vector<int>vec;

while (1) {

cout << "Что вы хотите сделать?\n";

cout << "1 - Ввести массив\n2 - найти последний отрицательный элемет\n";

cout << "3 - удалить все элементы перед каждым максимальным числом\n4 - вывести массив";

cout << "0 - выход из программы";

short c;

cin >> c;

switch (c)

{

case 0:

return 0;

break;

case 1:

enter(vec);

break;

case 2:

find(vec);

break;

case 3:

delete\_max(vec);

break;

case 4:

print\_arr(vec);

break;

default:

break;

}

}

}

Пример программы, которая удаляет подстроку bc, а также скобочки, которые обрамляют ее.



Рисунок 4.1 – Пример работы программы – удаление подстроки без пробелов

Пример программы, которая удаляет из текста с пробелами подстроку, содержащую пробелы.



Рисунок 4.2 – Пример работы программы – удаление подстроки с пробелами.

# ЗАКЛЮЧЕНИЕ

В ходе выполнения данной практической работы была реализована программа удаления текста между скобочек. Также были приобретены навыки работы со строчными данными на языке программирования C++.

# СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Лозовский В.В. Алгоритмические основы обработки данных: учебное пособие / Лозовский В.В., Платонова О.В., Штрекер Е.Н. — М.: МИРЭА – Российский технологический университет, 2022. – 337 с.

2. Платонова О.В. Алгоритмические основы обработки данных: методические указания / Платонова О.В., Асадова Ю.С., Расулов М.М. — М.: МИРЭА – Российский технологический университет, 2022. — 73 с.

3. Белик А.Г. Алгоритмы и структуры данных: учебное пособие / А.Г. Белик, В.Н. Цыганенко. — Омск: ОмГТУ, 2022. — 104 с. — ISBN 978-5-8149-3498-7. — Текст: электронный // Лань: электронно-библиотечная система. — URL: https://e.lanbook.com/book/343688 (дата обращения: 23.09.2024)

4. Павлов Л.А. Структуры и алгоритмы обработки данных / Л.А. Павлов, Н.В. Первова. — 2-е изд., стер. — Санкт-Петербург: Лань, 2022. — 256 с. — ISBN 978-5-507-44105-1. — Текст: электронный // Лань: электронно-библиотечная система. — URL: https://e.lanbook.com/book/207563 (дата обращения: 23.09.2024)

5. Пантелеев Е.Р. Алгоритмы и структуры данных: учебное пособие / Е.Р. Пантелеев, А.Л. Алыкова. — Иваново: ИГЭУ, 2018. — 142 с. — Текст: электронный // Лань: электронно-библиотечная система. — URL: https://e.lanbook.com/book/154576 (дата обращения: 23.09.2024)