|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  | | --- | --- | --- | |  |  |  | | 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 №9** | |
| **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 | «6» ноября 2024г. | *\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*  *(подпись студента)* |
|  |  |  |
| «Зачтено» | «13» ноября 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 №9**

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:** матрица n на m.

**4. Задание:**

Создать шаблоны функций, выполняющих ввод, вывод и упорядочивание матрицы. Протестировать шаблоны для матриц с элементами различных типов: int, float и char.

Разработать программу, которая вводит матрицу из n строк и m столбцов (n<=100, m<=50) и упорядочивает элементы матрицы. таким образом, чтобы при чтении матрицы по строкам ее элементы образовывали отсортированный по возрастанию массив.

**5. Содержание отчета:**

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

Руководитель работы Ю.С. Асадова \_\_\_\_\_\_\_\_\_\_\_\_ «4» ноября 2024г.

подпись

Задание принял к исполнению Т.А Туктаров \_\_\_\_\_\_\_\_\_\_\_\_ «28» октября 2024г.

подпись

**СОДЕРЖАНИЕ**

[ВВЕДЕНИЕ 4](#_Toc181801359)

[1 ОСНОВНОЙ РАЗДЕЛ 5](#_Toc181801360)

[2 БЛОК-СХЕМА АЛГОРИТМА.. 6](#_Toc181801361)

[3 ИСХОДНЫЙ КОД 9](#_Toc181801362)

[4 ПРИМЕР РАБОТЫ ПРОГРАММЫ 11](#_Toc181801363)

[ЗАКЛЮЧЕНИЕ 12](#_Toc181801364)

[СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 13](#_Toc181801365)

# ВВЕДЕНИЕ

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

Постановка задачи:

Разработать программу, которая вводит матрицу из n строк и m столбцов (n<=100, m<=50) и упорядочивает элементы матрицы таким образом, чтобы при чтении матрицы по строкам ее элементы образовывали отсортированный по возрастанию массив. Программа должна предоставлять пользователю выбор типа элементов матрицы: целого или строкового. Операции ввода, вывода и упорядочивания элементов матрицы должны быть реализованы в виде функций с шаблонами.

# 1 ОСНОВНОЙ РАЗДЕЛ

Программа состоит из 7-и функций. Главная – main(), в ней выполняется весь управляющий код. enter\_mat() – функция для ввода матрицы, принимает ссылку на матрицу. print\_mat() – функция для вывода матрицы, которая передается аргументу по ссылке. sort\_mat() – функция сортировки матрицы, которая передается аргументу по ссылке. Функции реализованы с помощью щаблонов, поэтому нам не надо создавать перегрузку для каждого возможного типа данных в матрице.

enter\_mat(), и print\_mat() работают по одинаковому принципу – с помощью вложенного цикла считывают/выводят данные.

sort\_mat() – Сортирует матрицу пузырьком, сравнивая каждый элемент с каждым.

# 2 БЛОК-СХЕМА АЛГОРИТМА..

Представим описание алгоритма в графическом виде на рисунках 2.1 – 2.5

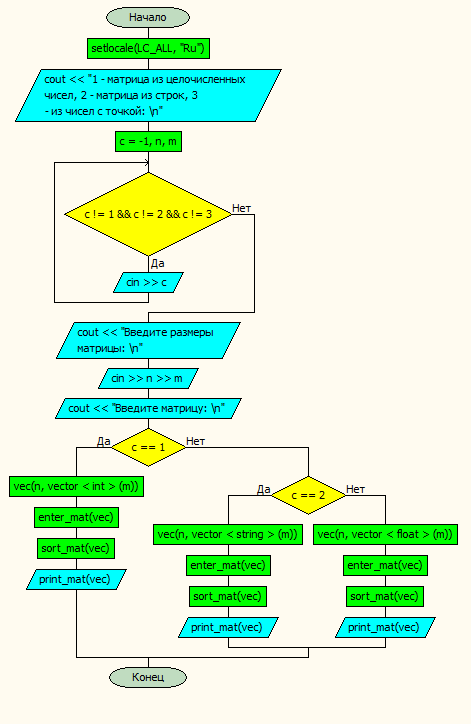


Рисунок 2.1 – Блок – схема алгоритма функции main()

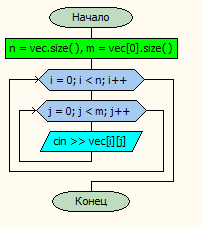


Рисунок 2.2 – Блок-схема функции enter\_mat()

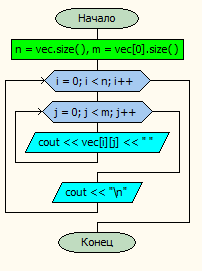


Рисунок 2.3 – Блок-схема функции print\_mat()

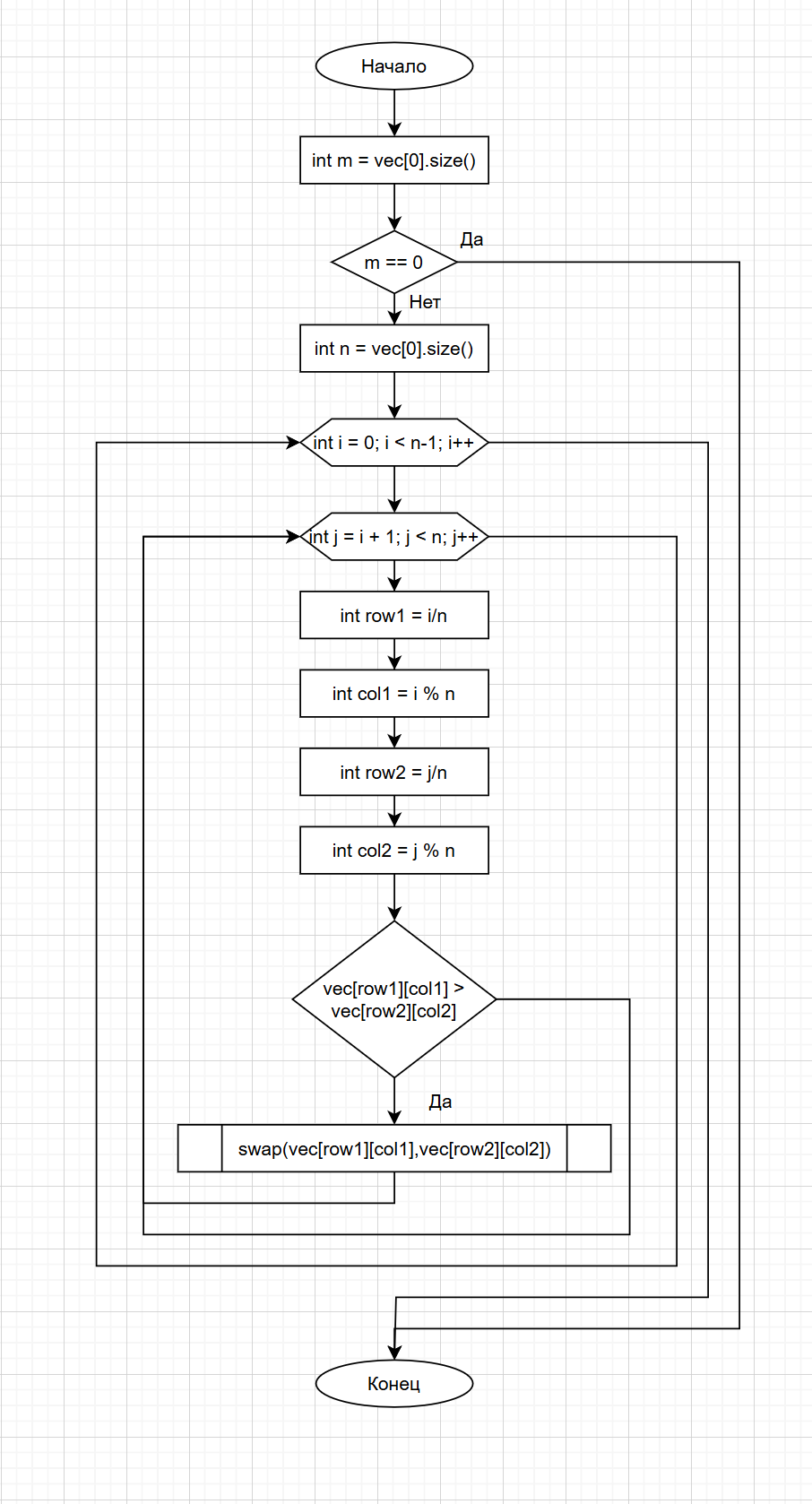


Рисунок 2.4 – Блок-схема функции sort\_mat()

# 3 ИСХОДНЫЙ КОД

Программная реализация алгоритма для решения задачи представлена ниже.

Листинг 3.1 – Исходный код программы

|  |
| --- |
| #include <iostream>  #include <string>  #include <vector>  #include <algorithm>  using namespace std;  template<typename Type>  void enter\_mat(vector<vector<Type>>& vec);  template<typename Type>  void sort\_mat(vector<vector<Type>>& vec);  template<typename Type>  void print\_mat(vector<vector<Type>>& vec);  int main()  {  setlocale(LC\_ALL, "Ru");  cout << "1 - матрица из целочисленных чисел, 2 - матрица из строк, 3 - из чисел с точкой: \n";  int c = -1, n, m;  while (c != 1 && c != 2 && c != 3)cin >> c;  cout << "Введите размеры матрицы: \n";  cin >> n >> m;  cout << "Введите матрицу: \n";  if (c == 1) {  vector<vector<int>>vec(n, vector<int>(m));  enter\_mat(vec);  sort\_mat(vec);  print\_mat(vec);  }  else if(c == 2){  vector<vector<string>>vec(n, vector<string>(m));  enter\_mat(vec);  sort\_mat(vec);  print\_mat(vec);  }  else {  vector<vector<float>>vec(n, vector<float>(m));  enter\_mat(vec);  sort\_mat(vec);  print\_mat(vec);  }  }  template<typename Type>  void enter\_mat(vector<vector<Type>>& vec) {  int n = vec.size(), m = vec[0].size();  for (int i = 0; i < n; i++)  {  for (int j = 0; j < m; j++) {  cin >> vec[i][j];  }  }  }  template<typename Type>  void sort\_mat(vector<vector<Type>>& vec) {  int m = vec.size();  if (m == 0) return;  int n = vec[0].size();  for (int i = 0; i < m \* n - 1; i++) {  for (int j = i + 1; j < m \* n; j++) {  int row1 = i / n, col1 = i % n;  int row2 = j / n, col2 = j % n;  if (vec[row1][col1] > vec[row2][col2]) {  swap(vec[row1][col1], vec[row2][col2]);  }  }  }  }  template<typename Type>  void print\_mat(vector<vector<Type>>& vec) {  int n = vec.size(), m = vec[0].size();  for (int i = 0; i < n; i++)  {  for (int j = 0; j < m; j++) {  cout << vec[i][j] << " ";  }  cout << "\n";  }  } |

# 4 Пример работы программы

Пример программы в которой выполняются все команды из условия.

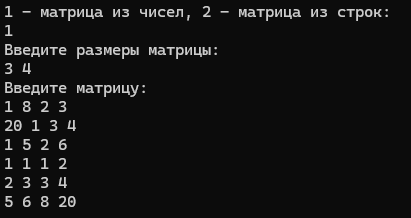


Рисунок 4.1 – Пример работы программы – сортировка числовой матрицы

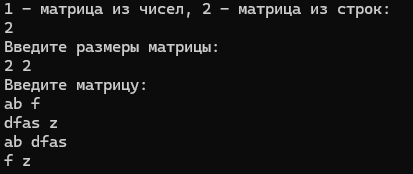


Рисунок 4.2 – Пример работы программы – сортировка строковой матрицы

# ЗАКЛЮЧЕНИЕ

В ходе выполнения данной практической работы была реализована программа для работы с матрицами. Также были приобретены навыки работы со массивами данных, перегрузкой функций и циклами. Были использованы шаблоны.

# 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)