**МИНОБРНАУКИ РОССИИ**

**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ**

**ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**

**«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)**

**Кафедра САПР**

**ОТЧЁТ**

**По курсовой работе №1**

**По дисциплине «Программирование»**

Студент гр. 3351 \_\_\_\_\_\_\_\_\_\_\_\_ Морозов А.А.

Преподаватель \_\_\_\_\_\_\_\_\_\_\_\_ Рыжов Н.Г.

Санкт-Петербург

2023

цель работы

Составить функцию для сортировки (перестановки строк) матрицы из n строк и m столбцов по возрастанию элементов k-го столбца; void sortmas ( int n, int m, int \*matr, int k); Вариант курсовой работы – 15.

**Текст программы**

#include <iostream>

#include <windows.h>

#include <conio.h>

#include <ctime>

using namespace std;

const int maxrand = 1000; // Максимальная граница псевдорандомных чисел

const int maxscale = 20; // Максимальный размер матрицы

const int language = 1251; // Установка русского языка

void sortmas(int n, int m, int\* matr, int k) { // Функция для сортировки матрицы по столбцу k

bool swapped;

for (int i = 0; i < n - 1 or swapped; i++) { // Условия для работы сортировки

swapped = false;

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

if (matr[j \* m + (k - 1)] > matr[(j + 1) \* m + (k - 1)]) { // Сравниваем элементы по столбцу k

for (int l = 0; l < m; l++) {

swap(matr[j \* m + l], matr[(j + 1) \* m + l]); // Если текущий элемент больше следующего, меняем строки местами

}

swapped = true;

}

}

}

}

int main() {

SetConsoleCP(language);

SetConsoleOutputCP(language);

srand(static\_cast<unsigned int>(time(0))); // Генератор псевдорандомных чисел

int n, m, size = 20, i, j, k; // Объявление переменных

cout << "Выберите размер матрицы." << "\n" << "Задайте количество столбцов m: " << "\n";

cin >> m; // Ввод количества стобцов m

cout << "Задайте количество строк n: " << "\n";

cin >> n; // Ввод количества строк n

while ((n > maxscale) or (m > maxscale) or (n < 0) or (m < 0)) { // Проверка правильности ввода

if (m > maxscale or m < 0) {

cout << "Неверный ввод размеров матрицы. Повторите ввод." << "\n" << "Повторно задайте количество столбцов m: " << "\n";

cin >> m; // Повторный ввод количества стобцов m

}

else {

cout << "Неверный ввод размеров матрицы. Повторите ввод." << "\n" << "Повторно задайте количество строк n: " << "\n";

cin >> n; // Повторный ввод количества строк n

}

}

cout << "Выберите столбец k по которому будет сортироваться массив: " << "\n";

cin >> k; // Ввод столбца сортировки k

while ((k > m) or (k < 0)) { // Проверка правильности ввода

cout << "В заданной матрице такого столбца не существует. Повторите ввод столбца k: " << "\n";

cin >> k; // Повторный ввод столбца сортировки k

}

cout << "Генерируется псевдорандомная матрица размера " << m << "x" << n << ":" << "\n";

int\* matr = new int[m \* n];

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

for (int j = 0; j < m; j++) {

matr[i \* m + j] = rand() % maxrand; // Генерация псевдослучайных чисел в матрице

cout << matr[i \* m + j] << "\t"; // Вывод элементов матрицы

}

cout << "\n";

}

sortmas(n, m, matr, k); // Вызов функции сортировки пузырьком

cout << "Отсортированная матрица по столбцу " << k << ":" << "\n";

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

for (int j = 0; j < m; j++) {

cout << matr[i \* m + j] << "\t"; // Вывод элементов упорядоченной матрицы

}

cout << "\n";

}

delete[] matr; // Освобождение памяти

\_getch();

}

**Описание программы**

Программное обеспечение – Microsoft Visual Studio 2022. Язык программирования – C++. При запуске программа просит пользователя задать необходимые параметры: размер матрицы, столбец, по которому будет производиться сортировка. Если параметры неверны (размер матрицы не может превышать 20х20), то пользователь должен будет повторить ввод. Далее программа создаёт матрицу указанных размеров из случайных (псевдослучайных) целых чисел. Потом выполняется пузырьковая сортировка матрицы по заданному столбцу перестановкой целых строк. В конце выводится отсортированная матрица.

**Руководство пользователя**

При запуске программа выводит сообщения с просьбой ввести параметры программы: количество столбцов m, количество строк n, номер столбца k, по которому матрица будет сортироваться. (Рис. 1)

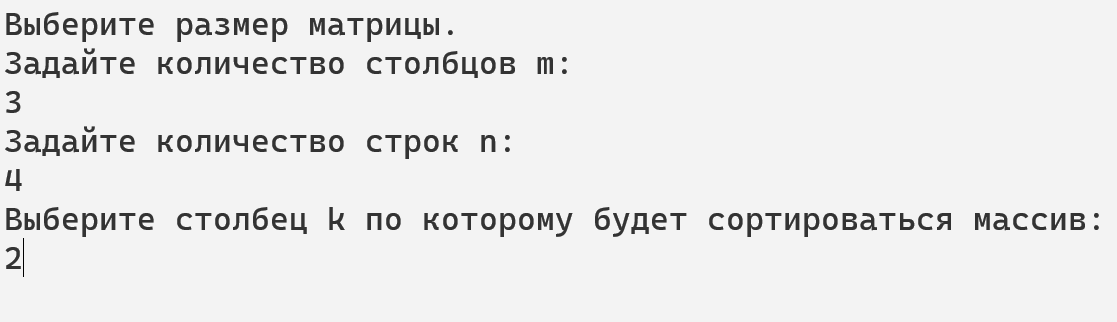


Рис. 1 – Настройка параметров

Если параметры введены неверно, то пользователь должен будет повторить ввод столько раз, пока не будут введены правильные данные. (Рис. 2)

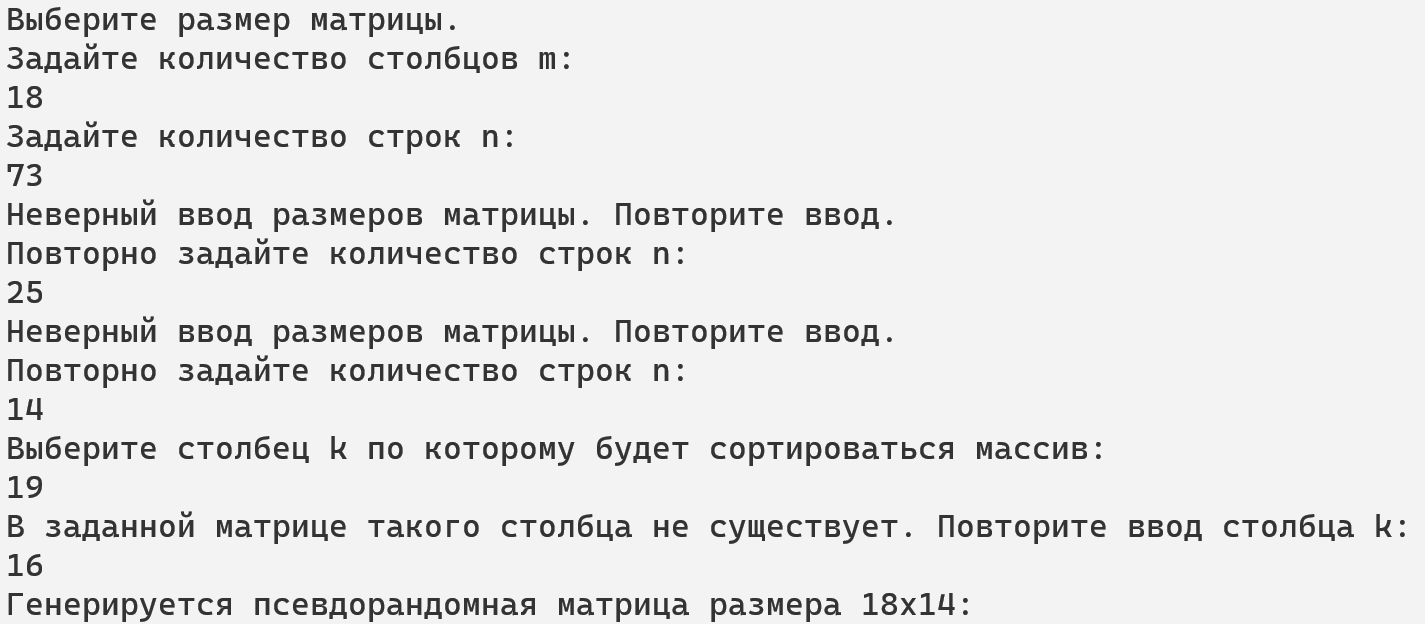


Рис. 2 – Повторный ввод

Далее программа генерирует матрицу из рандомных (псевдорандомных) целых чисел. Сначала выводится изначальная матрица, потом выполняется сортировка по столбцу k и в конце выводится упорядоченная матрица. (Рис. 3, Рис. 4)

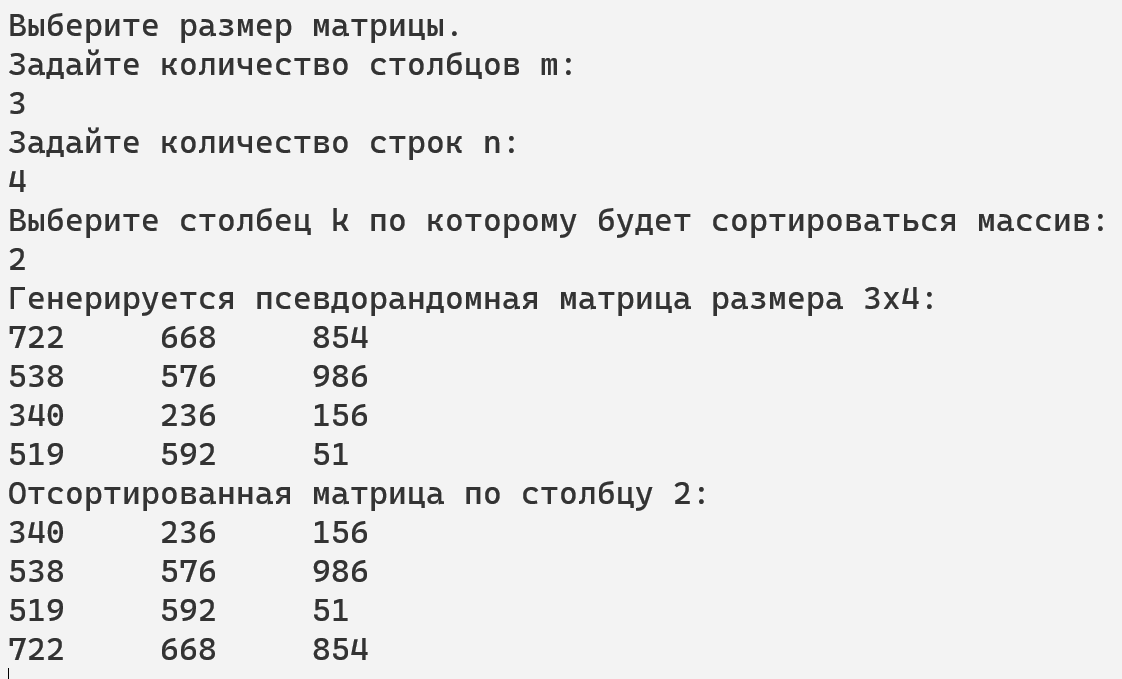


Рис. 3 – Работа программы

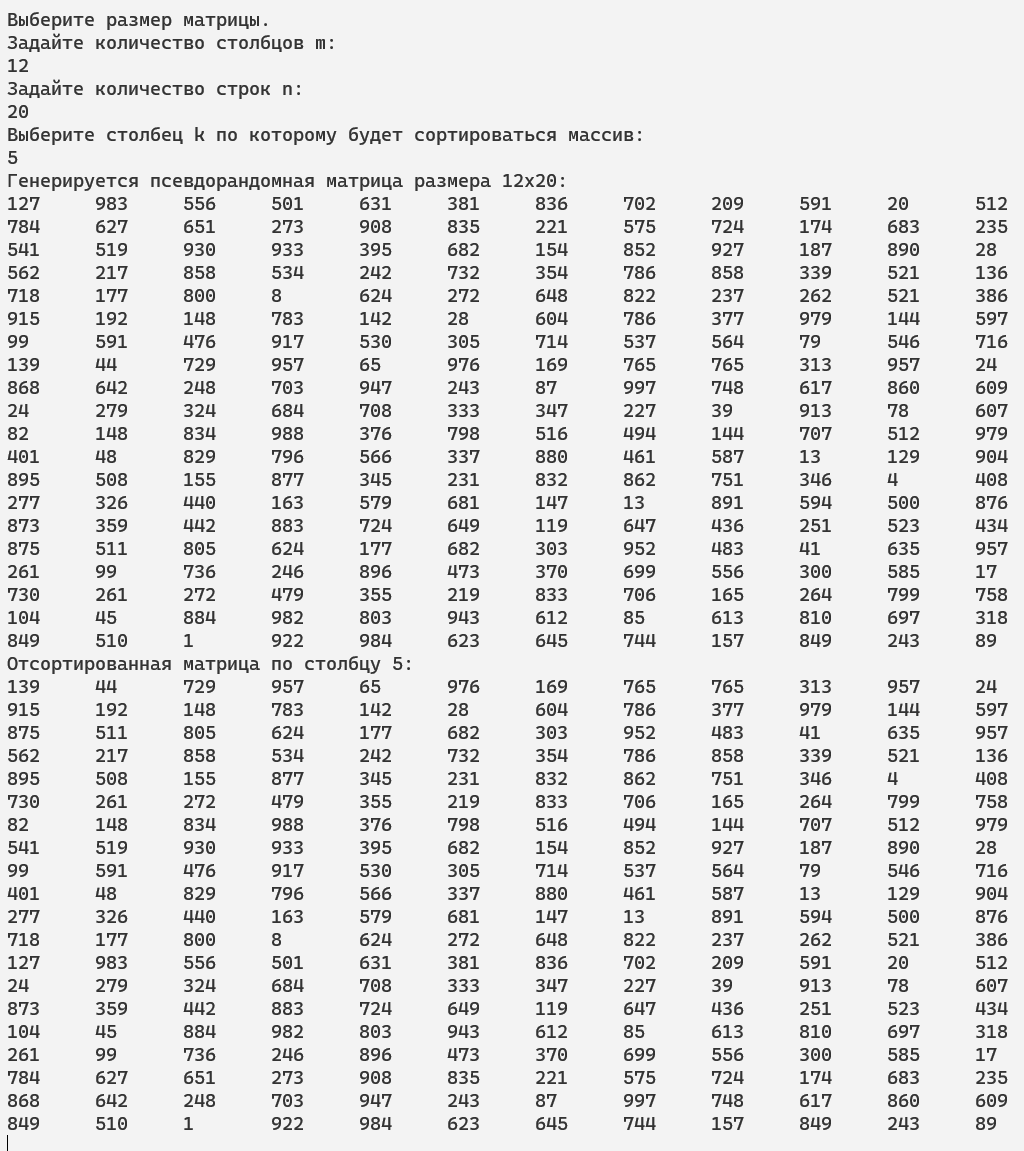


Рис. 4 – Работа программы

**Пути дальнейшего улучшения программы**

1. Улучшение визуальной части программы
2. Создание более удобного интерфейса программы