ЗВІТ

про виконання лабораторної роботи № 7.1

«Пошук заданих елементів та впорядкування рядків / стовпчиків матриці»

з дисципліни

«Алгоритмізація та програмування»

студента групи ІК-11

Зробка Івана Михайловича

**Умова завдання**

Написати програму, яка:

1. за допомогою генератора випадкових чисел формує вказану матрицю – так, що

значення її елементів належать заданому діапазону. Обчислити кількість та суму

тих елементів, які задовольняють вказаному критерію; а також замінити нулями ці

елементи;

2. впорядковує матрицю, переставляючи місцями її рядки / стовпчики відповідно до

вказаних ключів впорядкування.

Пояснити призначення змінних.

Необхідно:

- (завдання 1 і 2) сформувати матрицю;

- (завдання 1 і 2) вивести її на екран у вигляді таблиці, використовуючи форматне

виведення;

- (завдання 2) впорядкувати матрицю, переставляючи місцями її рядки / стовпчики

відповідно до вказаних ключів впорядкування;

- (завдання 2) вивести на екран модифіковану матрицю у вигляді окремої таблиці,

використовуючи виведення з тими самими специфікаціями формату;

- (завдання 1 і 2) виконати вказані у завданні дії (обчислити кількість та суму тих

елементів, які задовольняють вказаному критерію; а також замінити нулями ці

елементи);

- (завдання 1 і 2) вивести результат – обчислену кількість і суму, та вивести на

екран модифіковану матрицю у вигляді окремої таблиці, використовуючи вивення

з тими самими специфікаціями формату.

Усі вказані дії необхідно реалізувати за допомогою окремих функцій. Інформація у

функції має передаватися лише за допомогою параметрів. Використання глобальних змінних

– не допускається.

Кожна функція має виконувати лише одну роль, і ця роль має бути відображена у

назві функції.

Рекурсивний та ітераційний способи – це 2 різні проекти, для яких потрібно 2 різних

unit-тести і 2 різних звіти.

«Функція, яка повертає / обчислює / шукає ...» – має не виводити ці значення, а

повернути їх у місце виклику як результат функції або як відповідний вихідний параметр.

Діапазон значень елементів матриці: (7, …, 65)

Матриця A розмірністю (5  9) із елементів цілого типу.

Критерій – всі парні і крім кратних 7.

Ключі впорядкування: ↓ ↓ | ↑

– переставляються місцями рядки матриці – так, щоб елементи першого стовпчика

стали впорядкованими за спаданням;

– якщо в першому стовпчику наявні однакові елементи, то переставляються рядки

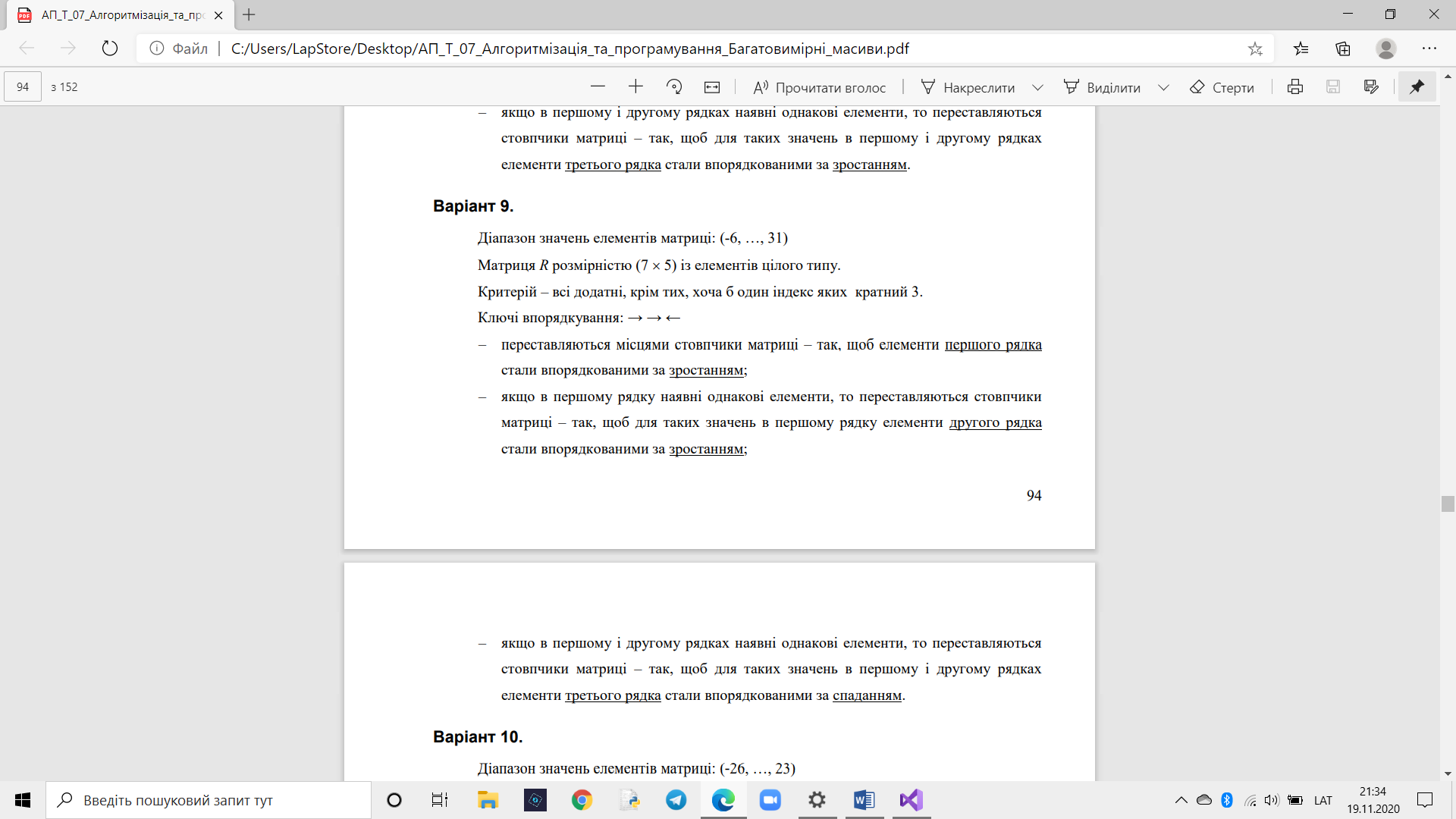
матриці – так, щоб для таких значень в першому стовпчику елементи другого

стовпчика стали впорядкованими за спаданням;

– якщо в першому і другому стовпчиках наявні однакові елементи, то

переставляються рядки матриці – так, щоб для таких значень в першому і другому

стовпчиках елементи четвертого стовпчика стали впорядкованими за зростанням.



// lab7.01.cpp : Этот файл содержит функцию "main". Здесь начинается и заканчивается выполнение программы.

//

#include <iostream>

#include<Windows.h>

#include<iomanip>

using namespace std;

void create(int\*\* arr, const int row, const int col, const int low, const int high)

{

for (int i = 0; i < row; i++)

{

for (int j = 0; j < col; j++)

{

arr[i][j] = low + rand() % (high - low + 1);

}

}

}

void print(int\*\* arr, const int row, const int col);

void sort(int\*\* arr, const int row,const int col);

void change(int\*\* arr,const int row1,const int row2,const int col);

void calc(int\*\* arr, const int row, const int col,int &K,int &S);

int main()

{

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

int Low = -6;

int High = 31;

int row = 7;

int col = 5;

int\*\* arr = new int\* [row];

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

arr[i] = new int[col];

}

create(arr, row, col, Low, High);

print(arr, row, col);

sort(arr, row, col);

print(arr, row, col);

int k = 0;

int s = 0;

calc(arr, row, col, s, k);

print(arr, row, col);

cout << endl<<"\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_";

cout << "Сума елемантів масиву "<<s << endl<< "Кількість елемменів масиву, ккі відповідають критерію "<< endl;

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

delete[] arr[i];

}

delete[] arr;

}

void print(int\*\* arr, const int row, const int col)

{

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

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

//if (arr[i][j] > 0 && i % 3 != 0 && j % 3 != 0)

cout << setw(4) << arr[i][j];

}

cout << endl;

}

cout << endl;

}

void sort(int\*\* arr, const int row,const int col)

{

for (int i = 0; i < row; i++)

for (int j = 0; j < row; j++)

if

change(arr, j, j + 1, col);

}

void change(int\*\* arr, const int row1,const int row2, const int col)

{

int tmp = 0;

for (int i = 0; i < col; i++)

{

tmp = arr[row1][i];

arr[row1][i] = arr[row2][i];

arr[row2][i] = tmp;

}

}

void calc(int\*\* arr, const int row, const int col, int& S, int& K)

{

K = 0;

S = 0;

for (int i = 0; i < row; i++)

{

for (int j = 0; i < col; i++)

{

if (arr[i][j] % 3 == 0)

K += 1;

S += arr[i][j];

arr[i][j] = 0;

}

}

}

