**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ**

**РОССИЙСКОЙ ФЕДЕРАЦИИ**

федеральное государственное автономное образовательное учреждение высшего образования «Санкт-Петербургский политехнический университет Петра Великого»

(ФГАОУ ВО «СПбПУ»)

**Институт среднего профессионального образования**

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ № 7

по дисциплине: «Основы алгоритмизации и программирования»

**Студент группы № 219/2**

**Клопова Ульяна**

**Вариант 15**

**Преподаватель: Рябова В.В.**

Санкт-Петербург

2024 г.

**Задание 1.** Создать динамический одномерный массив, заполнить случайными целыми числами. Вычислить количество положительных элементов в первой и во второй части массива. Ответ вывести с новой строки.

**#**include <iostream>

#include <iostream>

#include <ctime>

using namespace std;

int main()

{

setlocale(0,"");

cout << "Дан одномерный динамический массив" << endl;

int ahah, i, l =0, r =0, m= 0 ;

cin >> ahah;

int\* a = new int[ahah];

srand(time(0));

for ( i =0 ; i < ahah; i++)

{

a[i] = rand() % 210 - 100;

cout << a[i] << " ";

}

m = i / 2;

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

if (a[i] > 0) {

l++;

}

}

for (int i = m; i < ahah; i++) {

if (a[i] > 0) {

r++;

}

}

cout << endl << endl;

cout << "Количество положительных элементов в первой части: " << l << endl;

cout << "Количество положительных элементов во второй части: " << r << endl;

}

**Запуск 1.**

**Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание**

Результат при ahah = 7

**Запуск 2.**

**Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание**

Результат при ahah = 8

**Запуск 3.**

**Изображение выглядит как снимок экрана, текст, Шрифт

Автоматически созданное описание**

Результат при ahah = 33

**Задание 2.** Дан динамический двумерный массив a, размером (n×m). Заполнить одномерный массив, найдя количество положительных элементов в каждом столбце матрицы.

#include <iomanip>

#include <iostream>

#include <ctime>

#include <iomanip>

using namespace std;

int main() {

setlocale(0, "");

int r, c, z{};

cout << "Дан динамический двумерный массив a, размером (nxm). " << endl;

cout << "Заполнить одномерный массив, найдя количество положительных элементов в каждом столбце матрицы. " << endl;

cout << "Введите количество строк: ";

cin >> r;

cout << "Введите количество столбцов: ";

cin >> c;

cout << endl;

int\*\* a(new int\* [r] {});

srand(time(0));

for (int i{}; i < r; i++)

{

a[i] = new int [c] {};

}

for (int i{}; i < r; i++)

{

for (int j{}; j < c; j++)

{

a[i][j] = -10 + rand() % ( 20);

cout << setw(5) << a[i][j];

}

cout << endl;

}

int\* num = new int[c];

cout << endl;

cout << "Результат в каждом столбце: " << endl;

for (int j{}; j < c; j++)

{

for (int i{}; i < r; i++)

{

if (a[i][j] > 0)

{

z ++;

}

}

num[j] = z;

cout << setw(5) << z;

z = 0;

}

cout << endl << endl;

cout<< "Одномерный динамический массив: "<<endl;

for (int j{}; j < c; j++)

{

cout << setw(5)<< num[j];

}

for (int j{}; j < c; j++)

{

delete[] a[j];

}

delete a;

}

**Запуск 1.**

**Изображение выглядит как текст, снимок экрана, программное обеспечение, Мультимедийное программное обеспечение

Автоматически созданное описание**

Результат при r = 6 и c = 7

**Запуск 2.**

**Изображение выглядит как текст, снимок экрана, программное обеспечение, Мультимедийное программное обеспечение

Автоматически созданное описание**

Результат при r = 4 и c = 5

**Запуск 3.**

**Изображение выглядит как текст, снимок экрана

Автоматически созданное описание**

Результат при r = 10 и c = 5

**Задание 3.** Дан динамический двумерный массив a, размером (n×n) (или квадратная матрица. Найти количество нулевых элементов, расположенных выше побочной диагонали, включая саму побочную диагональ

#include <iomanip>

#include <iostream>

#include <ctime>

#include <iomanip>

using namespace std;

int main() {

setlocale(0, "");

int r, z{};

cout << "Дан динамический двумерный массив a, размером (n×n) (или квадратная матрица. " << endl;

cout << "Найти количество нулевых элементов, расположенных выше побочной диагонали, включая саму побочную диагональ" << endl;

cout << "Введите количество: ";

cin >> r;

cout << endl;

int\*\* a(new int\* [r] {});

srand(time(0));

for (int i{}; i < r; i++)

{

a[i] = new int [r] {};

}

for (int i{}; i < r; i++)

{

for (int j{}; j < r; j++)

{

a[i][j] = -10 + rand() % (20);

cout << setw(5) << a[i][j];

}

cout << endl;

}

cout << "Результат: " << endl;

for (int i{}; i < r; ++i)

{

for (int j=r-1-i; j < r; ++j)

{

if (a[i][j] == 0 )

{

z++;

}

}

}

cout << z << endl;

for (int i{}; i < r; ++i)

{

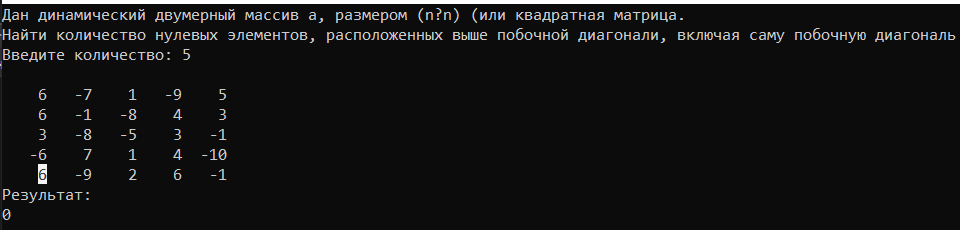
delete[] a[i];

}

delete a;

}

**Запуск 1.**

****

Результат при r = 5

**Запуск 2.**

**Изображение выглядит как текст, снимок экрана

Автоматически созданное описание**

Результат при r = 4

**Запуск 3.**

**Изображение выглядит как текст, снимок экрана, программное обеспечение, Мультимедийное программное обеспечение

Автоматически созданное описание**

Результат при r = 6

**Задание 4.** Дан динамический двумерный массив a, размером (n×n). Упорядочить по возрастанию последний столбец матрицы.

#include <iomanip>

#include <iostream>

#include <ctime>

#include <iomanip>

using namespace std;

void sortLastColumn(int\*\* a, int r) {

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

for (int j = i + 1; j < r; ++j) {

if (a[j][r - 1] < a[i][r - 1]) {

swap(a[i][r - 1], a[j][r - 1]);

}

}

}

}

int main() {

setlocale(0, "");

int r, z{};

cout << "Дан динамический двумерный массив a, размером (n×n). " << endl;

cout << "Упорядочить по возрастанию последний столбец матрицы." << endl;

cout << "Введите количество: ";

cin >> r;

cout << endl;

int\*\* a(new int\* [r] {});

srand(time(0));

for (int i{}; i < r; i++)

{

a[i] = new int [r] {};

}

for (int i{}; i < r; i++)

{

for (int j{}; j < r; j++)

{

a[i][j] = -10 + rand() % (20);

cout << setw(5) << a[i][j];

}

cout << endl;

}

cout << endl;

sortLastColumn(a, r);

cout << "Матрица после сортировки: " << endl;

for (int i{}; i < r; i++)

{

for (int j{}; j < r; j++)

{

cout << setw(5) << a[i][j];

}

cout << endl;

}

for (int i{}; i < r; ++i)

{

delete[] a[i];

}

delete a;

}

**Запуск 1.**

**Изображение выглядит как текст, снимок экрана, Шрифт, дизайн

Автоматически созданное описание**

Результат при r = 7

**Запуск 2.**

**Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание**

Результат при r = 4

**Запуск 3.**

**Изображение выглядит как текст, снимок экрана, Шрифт

Автоматически созданное описание**

Результат при r = 6