НИЖЕГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ

УНИВЕРСИТЕТ

ИМ. Р.Е. АЛЕКСЕЕВА

Кафедра "Вычислительные системы и технологии"

**ПРОГРАММИРОВАНИЕ**

**Отчёт**

**по лабораторной работе № 4**

Синтез и использование функций,

указатели, динамическое управление памятью

Вариант № 20

Выполнил студент группы 19-ИВТ-3

Сухоруков Валерий Алексеевич

«4»ноября 2019 г.

Проверил ст. преподаватель кафедры ВСТ

Мартынов Д.С.

«\_\_\_» \_\_\_\_\_\_\_ 2019 г.

Нижний Новгород 2019

Условие задачи:

Заданы матрицы A и C с размерами M×N и L×L соответственно.Написать функции умножения матрицы на скаляр и вычисления следа квадратной матрицы. Вычислить и напечатать Slc - след матрицы C и матрицу R = A×Slc.

Код задачи на языке С++

#include<iostream>

#include<vector>

#include<iomanip>

using namespace std;

vector<vector<int>> filling(int d, int c) {/\*функция заполнения массива\*/

char b;

vector<vector<int>>a(d);

cout << "choose how to fill the matrix : manually (enter m) or randomly (enter r) ";

cin >> b;

if (b == 'm')

{

cout << "enter the elements of the matrix ";

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

{

a[i].resize(c);

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

{

cin >> a[i][j];

}

}

}

if (b == 'r')

{

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

{

a[i].resize(c);

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

{

a[i][j]=rand()%200;

}

}

}

return a;

}

int proverka() {/\*функция проверки правильности ввода разрера массива\*/

int n = 0;

char b = 1;

while (cin.good()) {

if (n == 0) {

while (b != 0)

{

b = cin.get();

if (b >= '0' && b <= '9')

{

n = n \* 10 + (b - 48);

}

else {

if (b != '\n')

{

cout << "error, write again " << endl;

n = 0;

}

else

{

break;

}

}

}

}

else {

break;

}

}

return n;

}

int trace(vector<vector<int>>a,int q,int b) {/\*функция вычисления следа матрицы С\*/

int t=0;

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

{

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

{

if (i==j)

{

t = t + a[i][j];

}

}

}

return t;

}

vector<vector<int>> composition(int trace,int row,int column, vector<vector<int>>a) {/\*функция вычисления матрицы R (матрица А \* след матрицы С)\*/

vector<vector<int>>r(row);

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

{

r[i].resize(column);

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

{

r[i][j] = a[i][j] \* trace;

}

}

return r;

}

char logo[] = "\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n"

"\* Nizhniy Novgorod Technical University \*\n"

"\* Study work number 4. Task number 1. \*\n"

"\* Performed student 19-IVT-3 Sukhorukov V.A. \*\n"

"\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n";

int main() {

int m, n,l,tr;

cout << logo;

cout << "enter the number of rows in the matrix a"<<endl;

m = proverka();

cout << "enter the number of columns in the matrix a"<<endl;

n = proverka();

cout << "enter the number of rows and columns in the matrix C" << endl;

l = proverka();

vector<vector<int>>a = filling(m,n);

vector<vector<int>>c = filling(l, l);

tr = trace(c,l,l);

vector <vector<int>>r = composition(tr,m,n,a);

cout <<"trace matrix's C = "<< tr<<endl<<"matrix a is : "<<endl;

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

{

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

{

cout << setw(3) << a[i][j] << " ";

if (j == n - 1)

{

cout << endl;

}

}

}

cout<<"matrix C is : " << endl;

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

{

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

{

cout << setw(3) << c[i][j] << " ";

if (j == n - 1)

{

cout << endl;

}

}

}

cout<< "matrix R is : " << endl;

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

{

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

{

cout <<setw(5) <<r[i][j]<<" ";

if (j==n-1)

{

cout << endl;

}

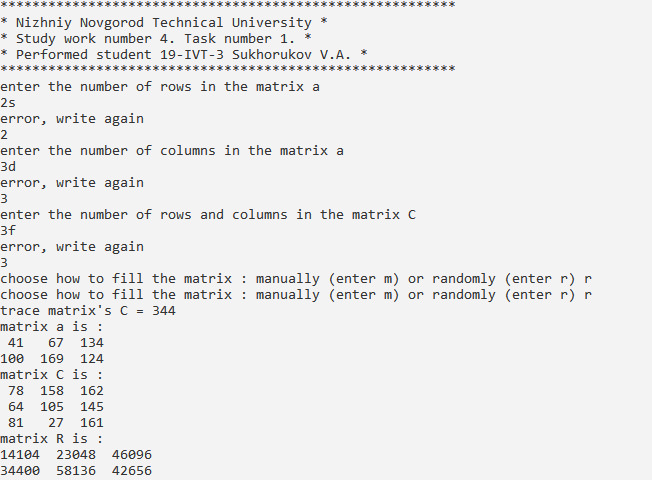
}

}

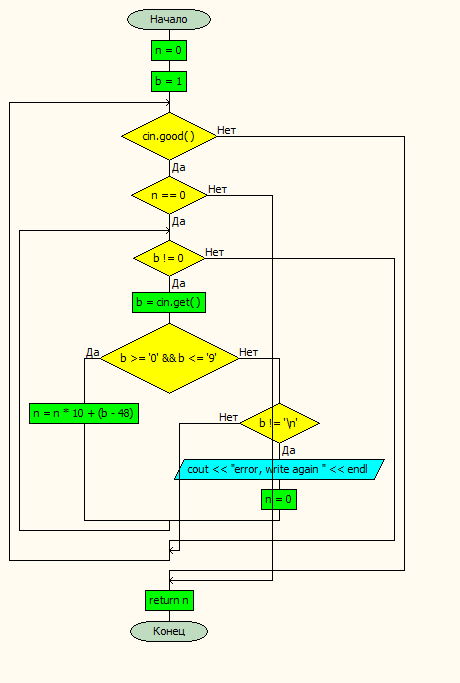
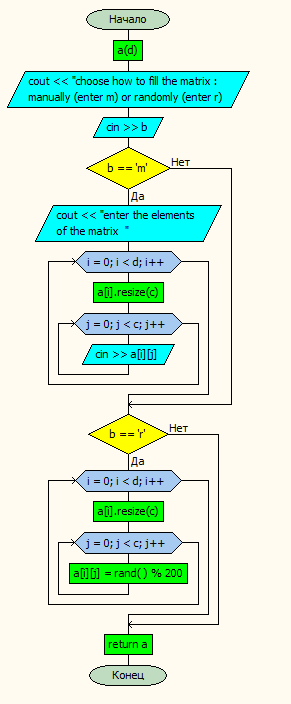
return 0;

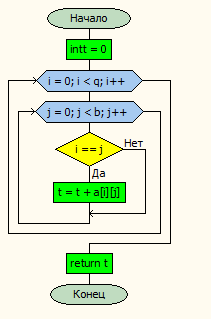
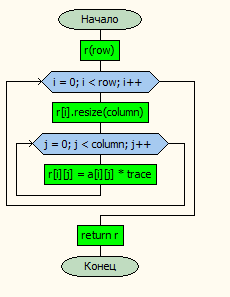
}

Результат работы программы



Блок схемы

Функция filling Функция proverka

Функция trace Функция composition

Функция main

