Проверил:

\_\_\_\_\_\_\_\_\_Антонов М.Б.

Дата: «\_\_» \_\_\_\_\_\_\_ 20 г.

Оценка:

Выполнила:

Студент группы Y2433

\_\_\_\_\_\_\_\_ Максимов Д.О

ОТЧЕТ

о лабораторной работе №2

09.02.07 Информационные системы и программирование

по теме: Одномерные и двумерные массивы

по дисциплине: Разработка программных модулей

УНИВЕРСИТЕТ ИТМО

Факультет среднего профессионального образования

Санкт-Петербург 2019

**Цель работы:** создать программу работающую с двумерными массивами.

**Задачи:**

* научиться работать с двумерными массивами;
* научиться передавать и возвращать несколько значений функции;
* научиться создавать динамические массивы.

**Задание:**

Пользователь должен задать размер двух матриц. Ввод с клавиатуры:

* количество строк первой матрицы;
* количество столбцов первой матрицы;
* количество строк второй матрицы;
* количество столбцов второй матрицы.

Требуется реализовать обязательно в виде отдельных функций:

* сумму двух матриц;
* произведение двух матриц;
* транспонирование матрицы;
* сложение и умножение матрицы на число.

Если какое-либо действие невозможно выполнить — написать соответствующее сообщение об ошибке.

**Текст программы**:

#include <iostream>

using namespace std;

void SumOfMatrix(int \*\*a, int \*\*b,int \*\*c,int& len1, int& wid1){

for(int i = 0; i < len1; i++){

for(int j = 0; j < wid1; j++){

c[i][j] = a[i][j] + b[i][j];

}

}

}

void MultiplyOfMatrix(int \*\*a, int \*\*b, int \*\*d,int width, int height){

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

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

int val = 0;

for(int v = 0; v < height; v++){

val += a[i][v]\*b[v][j];

}

d[i][j] = val;

}

}

}

void PrintMas(int \*\*a, const int& n1, const int& m1){

for(int i = 0; i < n1; i++){

for(int j = 0; j < m1; j++){

if(a[i][j] < 10){

cout.width(2);

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

}

if(a[i][j] >= 10){

cout.width(1);

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

}

}

cout << endl;

}

}

void FillArray(int\*\* array, int len,int wid){

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

array[i] = new int[wid];

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

array[i][j] = rand()%9+1;

} }

}

int\*\* Transp(int\*\* array,int n, int m)

{ int i,j;

int \*\*b;

b=new int \*[n];

if(b==NULL){

cout<< "Error" << endl;

exit(1);

}

for (i = 0 ;i < n ; i++){// n = length; m = width

b[i] = new int[m];

if(b[i] == NULL){

cout<< "Error" << endl;

exit(1);

}

}

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

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

b[j][i] = array[i][j];

}

}

return b;

}

int main(){

int len1, wid1, len2, wid2;

cout << "Enter width of first matrix" << endl;

cin >> wid1;

cout << "Enter length of first matrix" << endl;

cin >> len1;

int \*\*a;

a = new int\*[len1];

FillArray(a,len1,wid1);

cout << "First matrix" << endl;

PrintMas(a, len1, wid1);

cout << "Enter width of second matrix" << endl;// wid = m

cin >> wid2;

cout << "Enter length of second matrix" << endl;// len = n

cin >> len2;

int \*\*b;

b = new int\*[len2];

FillArray(b,len2,wid2);

cout << "Second matrix" << endl;

PrintMas(b, len2, wid2);

int \*\*transp1;

transp1 = Transp(a, len1, wid1);

cout << "The first transported matrix" << endl;;

PrintMas(transp1, wid1, len1);

int \*\*transp2;

transp2 = Transp(b, len2, wid2);

cout << "The second transported matrix" << endl;;

PrintMas(transp2, wid2, len2);

if ( (len1 == len2) && (wid1 == wid2)){

cout << "Sum of introduced matrixes is possible" << endl;

int \*\*c;

c = new int\*[len1];

for (int i = 0; i < len1 ;i ++){

c[i] = new int[wid1];

}

SumOfMatrix(a, b,c, len1, wid1);

cout << "Sum of first and second matrix is" << endl;

PrintMas(c, len1, wid1);

}else{

cout << "Sum of introduced matrixes is impossible" << endl;

}

if((wid1 == len2)||(wid2 == len1)){

cout << "multiplication of introduced matrixes is possible" << endl;

int \*\*d;

if(wid1 == len2){

d = new int\*[len2];

for (int i = 0; i < len2 ;i ++){

d[i] = new int[wid1];}

MultiplyOfMatrix(a, b, d , wid1,len2);

cout << "multiplication of introduced matrixes is "<< endl;

PrintMas(d, len2, wid1);

}

else

{

d = new int\*[len1];

for (int i = 0; i < len1 ;i ++)

{

d[i] = new int[wid2];

}

MultiplyOfMatrix(a, b, d , wid2 ,len1);

cout << "multiplication of introduced matrixes is " << endl;

PrintMas(d, len1, wid2);}

}

else

{

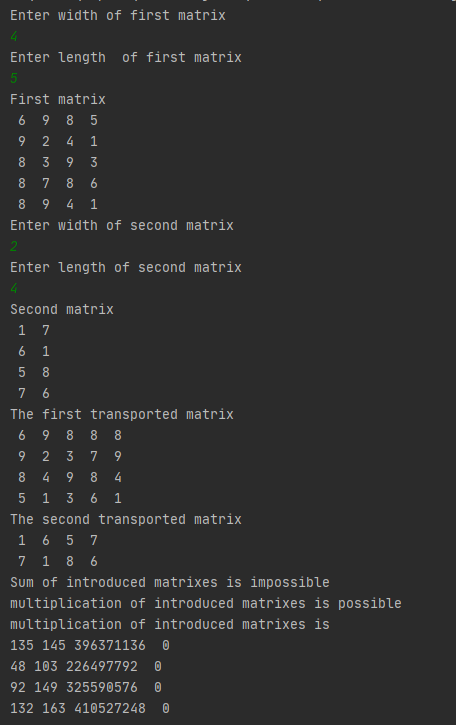
cout << "multiplication of introduced matrixes is impossible" << endl;

}

return 0;

}

Протокол работы программы представлен на рисунке 1.



*Рисунок 1- Протокол работы программы*