ЛАБОРАТОРНАЯ РАБОТА № 4

ИССЛЕДОВАНИЕ МЕХАНИЗМА ДРУЖЕСТВЕННОСТИ

1 ЦЕЛЬ РАБОТЫ

Приобретение практических навыков при написании объектно-ориентированных программ с использованием механизма множественного наследования.

2 ПОСТАНОВКА ЗАДАЧИ

Описать заданные по варианту классы (содержащие private поля и методы). Для каждого класса описать конструктор по умолчанию и конструктор с параметрами, а также деструктор (по необходимости). Создать функцию, дружественную обоим классам, и в ней обратиться к их закрытым полям и методам.

Вариант 3

Создать два класса: Вектор (int \*) и Матрица (int \*\*). Описать функцию, определяющую сумму их максимальных значений, как дружественную.

3 ТЕКСТ ПРОГРАММЫ

#include <conio.h>

#include <stdlib.h>

#include <iostream>

using namespace std;

class Matrix;

class Vector

{

int n;

int \*V;

public:

Vector(int n1)

{

n=n1;

V = new int [n];

cout<<"Введите значения элементов массива 1: "<<endl;

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

cin>>V[i];

cout<<endl;

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

cout<<V[i]<<" ";

cout<<endl<<endl;

}

~Vector()

{

delete V;

}

friend int Sum(Vector V, Matrix M);

};

class Matrix

{

int n, m;

int \*\*M;

public:

Matrix(int n2, int m1)

{

n=n2;

m=m1;

M = new int\* [n];

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

M[i] = new int [m];

cout<<"Введите значения элементов массива 2: "<<endl;

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

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

cin>>M[i][j];

cout<<endl;

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

{

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

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

cout<<endl<<endl;

}

}

~Matrix()

{

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

delete M[i];

delete M;

}

friend int Sum(Vector V, Matrix M);

};

int Sum(Vector ob1, Matrix ob2)

{

int max1=0, max2=0, sum;

for(int i=0; i<ob1.n; i++)

if(max1<ob1.V[i]) max1=ob1.V[i];

for(int i=0; i<ob2.n; i++)

for(int j=0; j<ob2.m; j++)

if(max2<ob2.M[i][j]) max2=ob2.M[i][j];

sum=max1+max2;

return sum;

}

int main ()

{

setlocale(0, "RUS");

Vector ob1(4);

Matrix ob2(3, 3);

cout<<"Сумма масимальных элементов массивов: "<<Sum(ob1, ob2)<<endl;

return 0;

}

4 ТЕСТИРОВАНИЕ

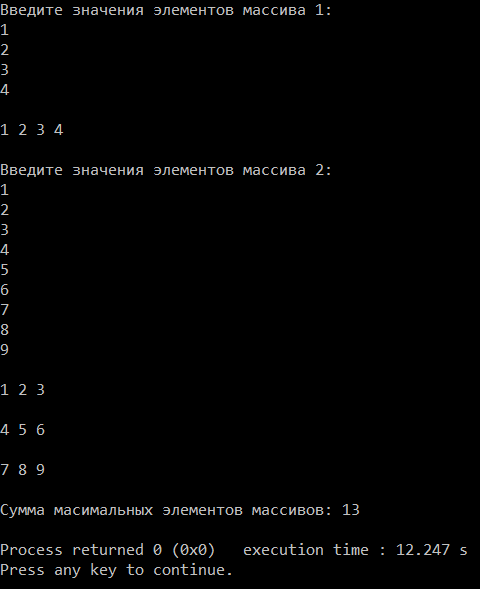


Рисунок 1 – Тестирование дружественной функции нахождения суммы максимальных элементов массивов.

ВЫВОД

В ходе выполнения лабораторной работы были приобретены практические навыки при написании объектно-ориентированных программ с использованием механизма множественного наследования.