**MINISTERUL EDUCAŢIEI, CULTURII ȘI CERCETĂRII AL REPUBLICII MOLDOVA**

**UNIVERSITATEA TEHNICĂ A MOLDOVEI**

**Facultatea Calculatoare, Informatică și Microelectronică**

**Departamentul Ingineria Software și Automatică**

**RAPORT**

**la lucrarea de laborator nr.5**

**Tema: „Polimorfism. Funcții virtuale”**

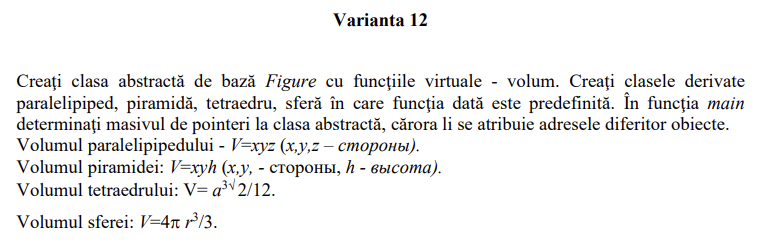
**Disciplina: „Clase și obiecte. Constructori și destructor pentru clasa”**

**A elaborat *st. gr. SI-211, Vozian Vladimir***

**A verificat *lect.univ S. Scrob***

**Chișinău 2022**

**Sarcina**



**Codul programului:**

#include <iostream>

#include <string>

#include <cmath>

using namespace std;

class Figura {

public:

    virtual float Volum(){};

private:

    float volum;

    friend class Parapelipiped;

    friend class Piramida;

    friend class Tetraedul;

    friend class Sfera;

};

class Parapelipiped : public Figura {

private:

    float x, y, z;

public:

    float Volum() override {

        volum = x \* y \* z;

        return volum;

    }

    Parapelipiped() {

        x = y = z = 0;

    }

    Parapelipiped(float x, float y, float z) {

        this->x = x;

        this->y = y;

        this->z = z;

    }

};

class Piramida : public Figura {

private:

    float x, y, h;

public:

    Piramida(){

        x=y=h=0;

    }

    Piramida(float x,float y,float inaltime){

        this->x = x;

        this->y = y;

        this->h = inaltime;

    }

    float Volum() override {

        volum = x \* y \* h;

        return volum;

    }

};

class Tetraedul:public Figura{

private:

    float a;

public:

    Tetraedul(){

        a = 0;

    }

    Tetraedul(float a){

        this->a=a;

    }

    float Volum() override{

        volum = pow(a,3)\*sqrt(2);

        return volum;

    }

};

class Sfera:public Figura{

private:

    float raza;

public:

    float Volum() override {

        volum = 4\*M\_PI\*pow(raza,3)/3;

        return volum;

    }

    Sfera(){

        raza=0;

    }

    Sfera(float raza){

        this->raza=raza;

    }

};

int main() {

    Figura \*arr\_fig[5]={

            new Sfera(8),

            new Tetraedul(5),

            new Piramida(5.3,8,1.9),

            new Parapelipiped(6,3.9,12),

            new Parapelipiped(9,1.4,7),

            };

    for (int i=0;i<5;i++){

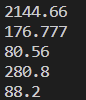
        cout<<arr\_fig[i]->Volum()<<endl;

    }

    return 0;

}

Rezultatele:



**Concluzii:**

Deci sa lucrat cu moștenirea multiplă in limbajul C++, sa creat o clasa abstractă principală cu o funcție virtuală și clasele ei derivate. Sa acumulat o experiență de lucru cu moștenirea și sa exersat asupra problemei ce poate apărea la moștenire, problema diamantului.