Министерство образования Республики Беларусь

Учреждение образования

«Брестский Государственный технический университет»

Кафедра ИИТ

**Лабораторная работа №1**

По дисциплине «Проектирование программ в ИС»

Тема: «Среда программирования С++. Разработка и документирование консольных приложений. Классы.»

**Выполнил:**

Студент 2 курса

Группы ИИ-21

Карагодин Д.Л.

**Проверил:**

Монтик Н.С.

Брест 2022

**Цель работы:** Изучение и решение индивидуальных заданий по разработке классов, описанию классов.

**Ход работы:**

Разработать неарифметический класс. Он должен в итоговом варианте содержать следующие члены:

1. 2-3 свойства-атрибута (один массивного типа).

2. Три конструктора (по одному каждого типа).

3. Методы, достаточные для установки и получения свойств объекта класса (например, TO\_SET(), TO\_SHOW() и т.п.).

4. Три перегруженных оператора (операторы типа “=”, а также двуместный и одноместный операторы).

Разработать арифметический класс с аналогичными требованиями к неарифметическому классу.

**Код программы:**

#include <iostream>

#include <iomanip>

using namespace std;

class Student {

private:

int age;

double score;

char name[64];

public:

Student() { this->age = 0; this->score = 0; strcpy\_s(this->name, "Noname"); }

Student(int age, double score, const char\* name) {

this->age = age;

this->score = score;

strcpy\_s(this->name, name);

}

Student(Student& st2) {

this->age = st2.age;

this->score = st2.score;

strcpy\_s(this->name, st2.name);

}

void To\_Set(int age, float score, const char\* name)

{

this->age = age;

this->score = score;

strcpy\_s(this->name, name);

}

void To\_Show()

{

cout << setw(-20) << "Age of student: " << age << endl;

cout << setw(-20) << "Academic score: " << score << endl;

cout << setw(-20) << "His Name and Surname: "; puts(name);

cout << endl;

}

int GetAge() { return this->age;}

float GetScore() {return this->score;}

char\* GetName() {return this->name;}

friend double operator-(Student& first, Student& second) {

return first.score -= second.score;

}

double operator+(double up) {return this->score += up;}

friend bool operator==(Student& first, Student& second) {

return !strcmp(first.name, second.name);

}

};

class Vector {

private:

double x, y, z;

string real\_or\_no;

public:

Vector() {

this->x = 0;

this->y = 0;

this->z = 0;

this->real\_or\_no = "false";

}

Vector(double x, double y, double z, string real\_or\_no = "true") {

this->x = x;

this->y = y;

this->z = z;

this->real\_or\_no = real\_or\_no;

}

Vector(const Vector& v2) {

this->x = v2.x;

this->y = v2.y;

this->z = v2.z;

this->real\_or\_no = v2.real\_or\_no;

}

void Set(double x, double y, double z, string real\_or\_no = "true") {

this->x = x;

this->y = y;

this->z = z;

this->real\_or\_no = real\_or\_no;

}

void Show() {

setprecision(3);

cout << "First coordinate(X): " << x << endl;

cout << "Second coordinate(Y): " << y << endl;

cout << "Third coordinate(Z): " << z << endl;

cout << "He is real?: " << real\_or\_no << endl << endl;

setprecision(6);

}

double GetX() {return this->x;}

double GetY() {return this->y;}

double GetZ() {return this->z;}

string GetRoN() {return this->real\_or\_no;}

friend Vector operator+(Vector& V1, Vector& V2) {

return Vector(V1.x + V2.x, V1.y + V2.y, V1.z + V2.z);

}

Vector operator-(double d) {

return Vector(this->x - d, this->y - d, this->z - d);

}

friend double operator\*(Vector& V1, Vector& V2) {

return (V1.x \* V2.x + V1.y \* V2.y + V1.z \* V2.z);

}

};

int main()

{

Student petya(18, 0, " ");

petya.To\_Set(18, 5.4, "Petya Popniy");

Student vasya(petya);

petya.To\_Show();

vasya.To\_Show();

char\* ch = new char[64];

cout << "vasya.name: ";

ch = vasya.GetName();

cout << ch << endl;

double d = vasya - petya;

cout << "Difference between score: " << d << endl

<< "Up for Petya: " << petya + 1 << endl

<< "This is one man?: " << (vasya == petya) << endl << endl;

Vector V1(2.5, 3.66, 8.65), V2(1.1, 2.2, 3.3), V3;

V1.Show(); V2.Show();

cout << "Addition: " << endl;

(V1 + V2).Show();

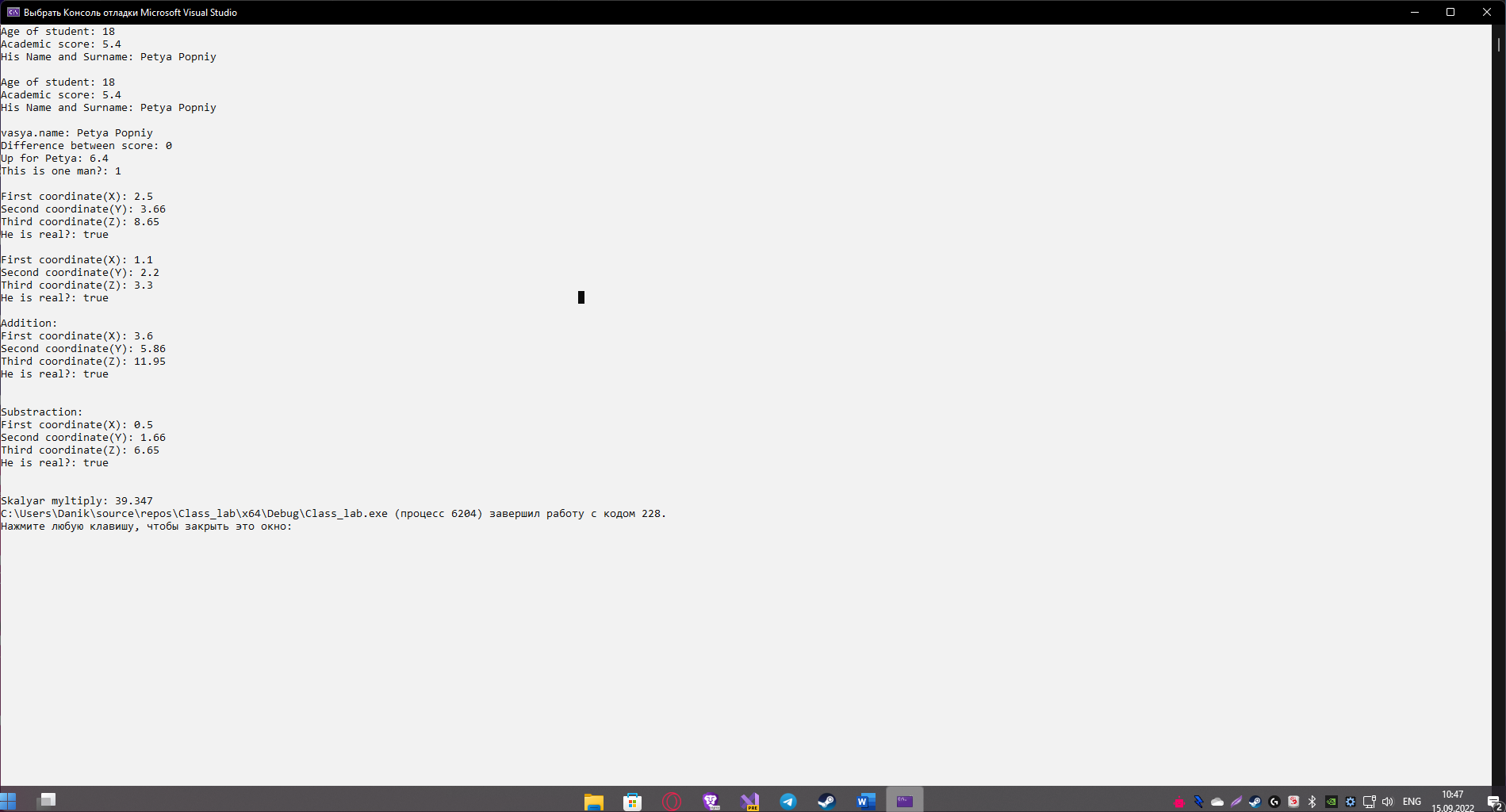
cout << endl << "Substraction: " << endl;

(V1 - 2).Show();

cout << endl << "Skalyar myltiply: " << V1 \* V2;

return 228;}

**Результат программы:**



**Вывод:** изучил и решил индивидуальные задания по разработке классов, описанию классов.