Министерство образования и науки Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования

**«Пермский национальный исследовательский политехнический университет»**

Кафедра **«**Информационные технологии и автоматизированные системы**»**

**ОТЧЕТ**

**ПО ЛАБОРАТОРНОЙ РАБОТЕ №1**

Дисциплина: «Основы алгоритмизации и программирования»

Семестр 2

Тема: Классы и объекты. Инкапсуляция.

Вариант 6.

Выполнил работу

Студент группы РИС-22-1Б

Поважный В. Е.

Проверил

Доцент кафедры ИТАС

Полякова О.А.

г. Пермь-2023

**Введение**

Для выполнения лабораторной работы требуется сделать отчет программы. Создание программ – отличный способ практики программирования.

**Постановка задачи**

1. Реализовать определение нового класса. Для демонстрации работы с объектами написать главную функцию. Продемонстрировать разные способы создания объектов и массивов объектов.
2. Структура-пара – структура с двумя полями, которые обычно имеют имена first и second. Требуется реализовать тип данных с помощью такой структуры. Во всех заданиях должны присутствовать: a. метод инициализации Init (метод должен контролировать значения аргументов на корректность); b. ввод с клавиатуры Read; c. вывод на экран Show.
3. Реализовать внешнюю функцию make\_тип(), где тип – тип реализуемой структуры. Функция должна получать значения для полей структуры как параметры функции и возвращать структуру как результат. При передаче ошибочных параметров следует выводить сообщение и заканчивать работу.



**Код программы**

**ооп 2.cpp**

#include <iostream>

#include "Header.h"

using namespace std;

int main()

{

double a, b, x;

cin >> a >> b >> x;

cunt func(a, b);

if (x != 0)

cout << func.root(x) << endl;

else

cout << "Err. X cannot be 0" << endl;

cin >> a >> b >> x;

cunt func1;

func1.setfirst(a);

func1.setsecond(b);

if (x != 0)

cout << func.root(x) << endl;

else

cout << "Err. X cannot be 0" << endl;

return 0;

}

**Header.h**

**#include <iostream>**

using namespace std;

class cunt

{

double first = 0;

double second = 0;

public:

cunt();

cunt(double a, double b);

~cunt();

void setfirst(double x);

void setsecond(double x);

double root(double x);

};

**Header.cpp**

**#include <iostream>**

#include "Header.h"

using namespace std;

cunt::cunt()

{

first = 0;

second = 0;

}

cunt::cunt(double a, double b)

{

this->first = a;

this->second = b;

}

cunt::~cunt()

{

this->first = 0;

this->first = 0;

}

void cunt::setfirst(double x)

{

this->first = x;

}

void cunt::setsecond(double x)

{

this->second = x;

}

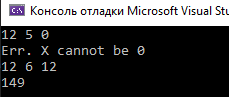
double cunt::root(double x)

{

return first \* x + second;

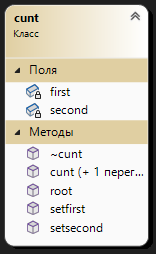
**}**

**Вывод программы**

****

***Рисунок 1 – вывод программы***

**UML диаграмма**

****

***Рисунок 2 – UML диаграмма***

**Вывод**

Программа выполняет свою задачу.

**Заключение**

Для решения задачи потребовались знания языка программирования, а конкретнее C++. Программа выполняет те условия, что были указаны в постановке задачи и работает без проблем.