Министерство науки и высшего образования Российской Федерации

Федеральное государственное автономное образовательное учреждение

высшего образования

**«Пермский национальный** **исследовательский политехнический университет»**

Электротехнический факультет

Кафедра «Информационные технологии и автоматизированные системы»

направление подготовки: 09.03.04– «Программная инженерия»

**Лабораторная работа №1.**

**Классы и объекты. Инкапсуляция.**

Выполнил студент гр. РИС-23-3Б

Мельников Антон Дмитриевич

Проверил:

ст. преп. каф. ИТАС

Полякова Ольга Андреевна

(оценка) (подпись)

(дата)

г. Пермь, 2024

**1.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.

**2.Описание класса**

1. Класс Pair представляет пару значений: оклад и количество отработанных дней.
2. Методы класса позволяют инициализировать, вводить с клавиатуры, выводить на экран и вычислять начисленную сумму на основе этих значений.
3. Внешняя функция make\_Pair создаёт объект Pair с переданными значениями.
4. Функция main демонстрирует использование класса и внешней функции.

**3.Код программы:**

#include <iostream>

using namespace std;

class Pair {

private:

double first; // оклад

int second; // количество отработанных дней

public:

// Метод инициализации

void Init(double f, int s)

{

first = f;

second = s;

}

// Метод ввода с клавиатуры

void Read()

{

double f;

int s;

cout << "Введите оклад: ";

cin >> f;

cout << "Введите количество отработанных дней: ";

cin >> s;

Init(f, s);

}

// Метод вывода на экран

void Show()

{

cout << "Оклад: " << first << ", Количество отработанных дней: " << second << endl;

}

// Метод вычисления суммы

double summa()

{

const int daysInMonth = 30; // Предполагаем, что в месяце 30 дней

return (first / daysInMonth) \* second;

}

};

// Внешняя функция создания объекта Pair

Pair make\_Pair(double f, int s) {

if (f <= 0 || s <= 0) {

cerr << "Ошибка: некорректные параметры." << endl;

exit(1);

}

Pair p;

p.Init(f, s);

return p;

}

int main()

{

setlocale(LC\_ALL, "Rus");

// Создание объекта класса Pair

Pair p1;

p1.Read();

p1.Show();

cout << "Начисленная сумма: " << p1.summa() << endl;

// Создание объекта класса Pair с помощью внешней функции

Pair p2 = make\_Pair(500.0, 20);

p2.Show();

cout << "Начисленная сумма: " << p2.summa() << endl;

// Массив объектов Pair

Pair arr[2];

for (int i = 0; i < 2; i++) {

arr[i].Read();

arr[i].Show();

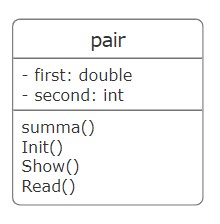
cout << "Начисленная сумма: " << arr[i].summa() << endl;

}

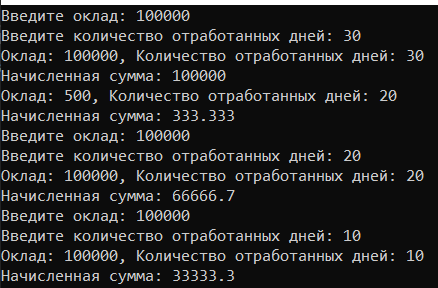
return 0;

}

**4.UML диаграмма класса**



**5.Скриншоты работы программы.**



**6.Вывод.**

Поставленная задача выполнена, программа работает

**7.GitHub.**

