Министерство науки и высшего образования Российской Федерации Федеральное государственное автономное образовательное учреждение высшего образования

**Пермский национальный исследовательский политехнический университет**

Факультет Электротехнический Кафедра ИТАС

Специальность Промышленная Робототехника

**ОТЧЁТ**

**о лабораторной работе №1**

Классы и объекты. Инкапсуляция.

|  |  |
| --- | --- |
|  | Выполнил:  Студент группы ПРТ-21-1Б  Торган Г.А.  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  Проверил:  Доцент кафедры ИТАС Полякова О.А.  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |

**Пермь 2022**

**Постановка задачи:**

Поле first – положительное целое число, целая часть числа, поле second – положительное дробное число, дробная часть числа. Реализовать метод multiply(int k) – умножение на целое число k. Найти сумму ряда с точностью ε=10-4, общий член которого: .

**Текст программы:**

**Lab1.cpp**

#include<iostream>

#include "fraction.h"

using namespace std;

fraction make\_fraction(double F, int S)

{

fraction t;//создали временную переменную

t.Init(F, S);//инициализировали поля переменной t с помощью параметров функции

return t;//вернули значение переменной t

}

int main()

{

extern int k;

//определение переменных А и В

fraction A{};

fraction B{};

A.Init(3.0, 2);//инициализация переменной А

B.Read();//ввод полей переменных В

A.Show();//вывод значений полей переменной А

B.Show();//вывод значений полей переменной В

//вывод значения степени, вычисленного с помощью функции Multiply()

cout << "A.Multiply(" << A.first << "," << A.second << ")=" << A.Multiply(k) << endl;

cout << "B.Multiply(" << B.first << "," << B.second << ")=" << B.Multiply(k) << endl;

//указатели

fraction\* X = new fraction;//выделение памяти под динамическую переменную

X->Init(2.0, 5);//инициализация

X->Show();//вывод значений полей

X->Multiply(k);//вычисление степени

cout << "X.Multiply(" << X->first << "," << X->second << ")=" << X->Multiply(k) << endl;

//массивы

fraction mas[3];//определение массива

for (int i = 0; i < 3; i++)

mas[i].Read();//чтение значений полей

for (int i = 0; i < 3; i++)

mas[i].Show(); //вывод значений полей

for (int i = 0; i < 3; i++)

{

mas[i].Multiply(k); //вычисление степени

cout << "mas[" << i << "].Multiply(" << mas[i].first << "," << mas[i].second << ")=";

cout << mas[i].Multiply(k) << endl;

}

//динамические масивы

fraction\* p\_mas = new fraction[3];//выделение памяти

for (int i = 0; i < 3; i++)

p\_mas[i].Read();//чтение значений полей

for (int i = 0; i < 3; i++)

p\_mas[i].Show();//вывод значений полей

for (int i = 0; i < 3; i++)

{

p\_mas[i].Multiply(k);//вычисление степени

cout << "p\_mas[" << i << "].Multiply(" << p\_mas[i].first << "," << p\_mas[i].second;

cout << ")=" << p\_mas[i].Multiply(k) << endl;

}

//вызов функции make\_fraction()

double y; int z;

cout << "first?"; cin >> y;

cout << "second?"; cin >> z;

//переменная F формируется с помощью функции make\_fraction()

fraction F = make\_fraction(y, z);

F.Show();

return 0;

}

**Fraction.h**

struct fraction

{

int first;

double second;

void Init(int, double);//метод для инициализации полей

void Read();//метод для чтения значений полей

void Show();//метод для вывода значений полей

double Multiply(int k);//вычисление произведения

};

**Fraction.cpp**

#include <iostream>

#include "fraction.h"

using namespace std;

//реализация метода для инициализации полей структуры

void fraction::Init(int F, double S)

{

first = F; second = S;

}

//реализация метода для чтения значений полей структуры

void fraction::Read()

{

cout << "\nfirst?"; cin >> first;

cout << "\nsecond?"; cin >> second;

}

//реализация метода для вывода значений полей структуры

void fraction::Show()

{

cout << "\nfirst=" << first;

cout << "\nsecond=" << second;

cout << "\n";

}

//метод для возведения в степень

double fraction:: Multiply(int k)

{

double cel = first;

cout << "\nK?"; cin >> k;

return (k\*(cel+second));

}