Министерство высшего образования и науки Российской Федерации

Федеральное государственное автономное образовательное учреждение высшего образования

**«Пермский национальный исследовательский политехнический университет» (ПНИПУ)**

Электротехнический факультет

Кафедра «Информационные технологии и автоматизированные системы»

ОТЧЁТ

Лабораторная работа №1

Классы и объекты. Инкапсуляция.

Выполнил

Студент группы РИС-22-2б

Вековшинин Д. А.

Проверила

Доцент кафедры ИТАС

Полякова О. А.

Пермь 2023

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

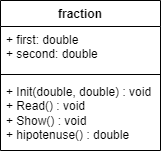
структуры. Функция должна получать значения для полей структуры как

параметры функции и возвращать структуру как результат. При передаче

ошибочных параметров следует выводить сообщение и заканчивать работу.

Вариант 10: Поле first – дробное положительное число, катетa прямоугольного треугольника, поле second – дробное положительное число, катет b прямоугольного треугольника. Реализовать метод hipotenuse() –вычисление гипотенузы.

UML-диаграмма



Определение компонентных функций

//реализация метода для инициализации полей структуры

void fraction::Init(double 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::hipotenuse()

{

return pow(pow(first, 2)+pow(second, 2), 0.5);

}

Определение функции make\_()

fraction make\_fraction(double F, double S)

{

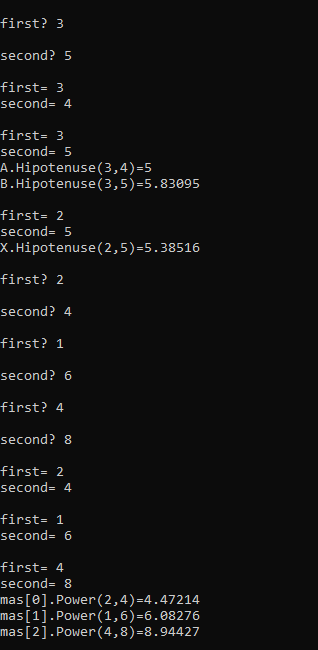
fraction t;//создали временную переменную

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

return t;//вернули значение переменной t

}

Результаты работы программы



Ответы на контрольные вопросы

1. Что такое класс?

Класс является абстрактным типом данных, определяемым пользователем, и представляет собой модель реального объекта в виде данных и функций для работы с ними.

1. Что такое объект (экземпляр) класса?

Поля и методы называются объектами класса.

class fraction

{

double first;

double second;

void Init(double, double);

};

1. Как называются поля класса?

Поля класса называются атрибутами

double first;

double second;

1. Как называются функции класса?

Функции класса называются методами

void Init(double, double);

void Read();

void Show();

double hipotenuse();

1. Для чего используются спецификаторы доступа?

Спецификаторы доступа управляют видимостью элементов класса

private:

double first;

double second;

public:

void Init(double, double);

1. Для чего используется спецификатор public?

Спецификатор public используется для описания интерфейса класса

public:

void Init(double, double);

1. Для чего используется спецификатор private?

Элементы, описанные после служебного слова private, видимы только внутри класса.

private:

double first;

double second;

1. Если описание класса начинается со спецификатора class, то какой спецификатор доступа будет использоваться по умолчанию?

Спецификатор доступа private принят в классе по умолчанию.

class fraction

{

double first; //private

double second; //private

};

1. Если описание класса начинается со спецификатора struct, то какой спецификатор доступа будет использоваться по умолчанию?

Если при описании класса используется слово struct, то все поля и

методы по умолчанию будут общедоступными (public).

struct fraction

{

double first; //public

double second; //public

};

1. Какой спецификатор доступа должен использоваться при описании интерфейса класса? Почему?

public, так как поля интерфейса класса должны быть видимы за пределами этого класса.

public:

void Init(double, double);

1. Каким образом можно изменить значения атрибутов экземпляра класса?

они инициализируются только один раз (с помощью конструктора) и не могут изменяться

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

{

first = F; second = S;

}

1. Каким образом можно получить значения атрибутов экземпляра класса?

Получить информацию о содержимом полей, описанных после спецификатора private можно только с помощью специальных методов, которые называются селекторами.

class fraction

{

double first;

double second;

public:

double getFirst()

{

return first;

};

};

fraction f;

double first = f.getFirst();

13. Класс описан следующим образом

struct Student

{

string name;

int group;

………

};

Объект класса определен следующим образом

Student \*s=new Student;

Как можно обратиться к полю name объекта s?



14. Класс описан следующим образом

struct Student

{

string name;

int group;

……..

};

Объект класса определен следующим образом

Student s;

Как можно обратиться к полю name объекта s?



15. Класс описан следующим образом

class Student

{

string name;

int group;

…….

};

Объект класса определен следующим образом

Student \*s=new Student;

Как можно обратиться к полю name объекта s?

К полю name объекта s нельзя обратиться, так как оно имеет спецификатор доступа private.



16. Класс описан следующим образом

class Student

{

string name;

int group;

public:

….. };

Объект класса определен следующим образом

Student s;

Как можно обратиться к полю name объекта s?

К полю name объекта s нельзя обратиться, так как оно имеет спецификатор доступа private.



17. Класс описан следующим образом

class Student

{

public:

char\* name;

int group; ………

};

Объект класса определен следующим образом

Student \*s=new Student;

Как можно обратиться к полю name объекта s?

