Министерство науки и высшего образования Российской Федерации

Федеральное государственное автономное образовательное учреждение

высшего образования

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

Электротехнический факультет

Кафедра «Информационные технологии и автоматизированные системы»

направление подготовки: 09.03.01- «Информатика и вычислительная техника»

**Лабораторная работа по классам № 1**

**по дисциплине**

**«Основы алгоритмизации и программирования»**

**«Классы и объекты. Инкапсуляция.»**

Выполнил студент гр. ИВТ-21-1Б

Галавтдинов Станислав Сергеевич

Проверил:

Яруллин Денис Владимирович

(оценка) (подпись)

(дата)

г. Пермь, 2021

**Постановка задачи**:

1. Реализовать определение нового класса. Для демонстрации работы с объектами написать главную функцию. Продемонстрировать разные способы создания объектов и массивов объектов.
2. Поле first – дробное число x, координата точки, поле second – дробное число y, координата точки. Реализовать метод distance() – вычисление расстояния от точки с координатами (first,second) до начала координат.

**Анализ задачи:** 1) Класс является абстрактным типом данных, определяемым пользователем, и представляет собой модель реального объекта в виде данных и функций для работы с ними.

2) Данные класса называются полями (по аналогии с полями структуры) или атрибутами, а функции класса — методами. Поля и методы называются элементами класса.

3) Спецификаторы доступа private и public управляют видимостью элементов класса. Элементы, описанные после служебного слова private, видимы только внутри класса. Этот вид доступа принят в классе по умолчанию. Интерфейс класса описывается после спецификатора public. Действие любого спецификатора распространяется до следующего спецификатора или до конца класса. Можно задавать несколько секций private и public, порядок их следования значения не имеет.

4) Получить информацию о содержимом полей, описанных после спецификатора private можно только с помощью специальных методов, которые называются селекторами, а изменить – с помощью методов, которые называются модификаторами.

5) Видимостью элементов класса можно также управлять с помощью ключевых слов struct и class. Если при описании класса используется слово struct, то все поля и методы по умолчанию будут общедоступными public. Если при описании класса используется слово class, то по умолчанию все методы и поля класса будут скрытыми private.

**Код задачи:** #include <iostream>

#include <cmath>

using namespace std;

class A {

private:

double d;

public:

void distance(double x, double y)

{

d = sqrt(pow(x, 2) + pow(y, 2));

}

void Print()

{

cout << "Расстояние: " << d;

}

};

int main()

{

setlocale(LC\_ALL, "Rus");

A a;

double x, y;

cout << "Координата x: ";

cin >> x;

cout << "Координата y: ";

cin >> y;

a.distance(x, y);

a.Print();

}

**Результат программы (рисунок 2):**



Рисунок 2

**Контрольные вопросы:**

1. Что такое класс?

Класс – абстрактный тип данных, определяемый пользователем.

1. Что такое объект(экземпляр) класса?

Объект – переменная класса.

1. Как называются поля класса?

Поля класса – данные класса, которые содержатся в нем.

1. Как называются функции класса?

Функциями класса называются методы.

1. Для чего используется спецификаторы доступа?

Спецификаторы используются для видимости компонентов класса или наоборот.

1. Для чего используется спецификатор public?

Спецификатор public используется для доступа к компонентам класса извне.

1. Для чего используется спецификатор private?

Спецификатор private используется для ограничения доступа к компонентам класса извне.

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

Private.

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

Public.

10) Какой спецификатор доступа должен использоваться при описании интерфейса класса? Почему?

Должен использоваться спецификатор public для того, чтобы к методу можно было обратиться извне.

11) Каким образом можно изменить значения атрибутов экземпляра класса? Изменить значение можно через методы, описанные в public, или, если поля публичные, обратиться к ним напрямую.

12) Каким образом можно получить значения атрибутов экземпляра класса? Получить значение можно через методы, описанные в public, или, если поля публичные, обратиться к ним напрямую.

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?

S[i].name = “”;// i – номер элемента

14. Struct Student

{

String name;

Int group;

…

};

Student s;

Как можно обратиться к полю name объекта s?

s.name = “”;

15. class Student

{

String name;

Int group;

…

};

Student \*s = new Student;

Можно обратиться только через методы класса.

16.

class Student

{ String name;

Int group;

…

};

Объект класса определен следующим образом

Student s;

Как можно обратиться к полю name объекта s?

Можно обратиться только через методы класса.

17. Класс описан следующим образом:

class Student {

Public:

String name;

Int group;

…

};

Объект класса определен следующим образом:

Student\* s = new Student;

Как можно обратиться к полю name объекта s?

S[i].name = “”; // i – номер элемент