东 北 大 学 秦 皇 岛 分 校 计 算 中 心

**实 验 报 告**

|  |  |  |  |
| --- | --- | --- | --- |
| 实验题目：继承与派生 | | | |
| 专业：自动化类 | 班级：2301班 | 学号 ：202313653 | 姓名：焦一晟 |
| 实验日期：2024/4/16 | 机器号： | 实验得分： | 指导教师签字： |

**一、实验目的：**

1. 了解继承在面向对象程序设计中的重要作用

2. 理解继承和派生的概念

3. 掌握继承和派生的使用方法

4. 了解虚基类的作用和用法

**二、实验内容(习题)：**

Prg12.cpp是我举的虚基类应用的实例

**Prg12.cpp**

/\*

在这个示例中，Animal类被声明为虚基类,

这样无论 Dog和 Cat类分别继承 Animal类多少次,

最终在 DogCat类中 Animal类的成员都只有一份实例。

\*/

#include<bits/stdc++.h>

using namespace std;

class Animal{

public:

virtual void eat(){

cout<<"Animal is eating"<<endl;

}

};

class Dog:virtual public Animal{

public:

void bark(){

cout<<"Dog is barking"<<endl;

}

};

class Cat:virtual public Animal{

public:

void meow(){

cout<<"Cat is meowing"<<endl;

}

};

class DogCat:public Dog,public Cat{

public:

void play(){

cout<<"DogCat is playing"<<endl;

}

};

int main(){

DogCat dc;

dc.eat();//调用从虚基类 Animal继承的eat函数

dc.bark();//调用从类 Dog继承的bark函数

dc.meow();//调用从类 Cat继承的meow函数

dc.play();//调用类 DogCat自己的play函数

return 0;

}

**Prg13.cpp**

#include<iostream>

#include"vehicle.h"

//自行车类

class Bicycle:public Vehicle{

private:

int high;//车高度

public:

//构造函数

Bicycle(int h=0,int s=0,int w=0):high(h),Vehicle(s,w){}

//获取车高度

int getHigh()const{return high;}

//设置车高度

void setHigh(int h){high=h;}

};

// 汽车类

class Car:public Vehicle{

private:

int seatNum;//座位数

public:

//构造函数

Car(int seat=0,int s=0,int w=0):seatNum(seat),Vehicle(s,w){}

//获取座位数

int getSeatNum()const{return seatNum;}

//设置座位数

void setSeatNum(int seat){seatNum=seat;}

};

//摩托车类，从自行车和汽车派生

class Motorcycle:public Bicycle,public Car{

public:

//构造函数

Motorcycle(int h=0,int seat=0, int s=0, int w=0):

Bicycle(h,s,w),Car(seat,s,w){}

};

int main() {

//在这里进行测试

Motorcycle moto(100,2,80,500);//车把高度为100,座位数为2,速度80,重量500

std::cout<<"Speed:"<<moto.getSpeed()<<std::endl;

std::cout<<"Weight:"<<moto.getWeight()<<std::endl;

std::cout<<"Handlebar Height:"<<moto.getHigh()<<std::endl;

std::cout<<"Seat Number:"<<moto.getSeatNum()<<std::endl;

return 0;

}

Vehicle.h

// vehicle.h

#ifndef VEHICLE\_H

#define VEHICLE\_H

// 车辆基类

class Vehicle{

private:

int speed; // 速度

int weight; // 重量

public:

// 构造函数

Vehicle(int s = 0, int w = 0) : speed(s), weight(w) {}

// 获取速度

int getSpeed() const { return speed; }

// 获取重量

int getWeight() const { return weight; }

// 设置速度

void setSpeed(int s) { speed = s; }

// 设置重量

void setWeight(int w) { weight = w; }

};

#endif // VEHICLE\_H

**三、总结：实验操作中遇到的问题、心得体会等**

多多调试，掌握过程。