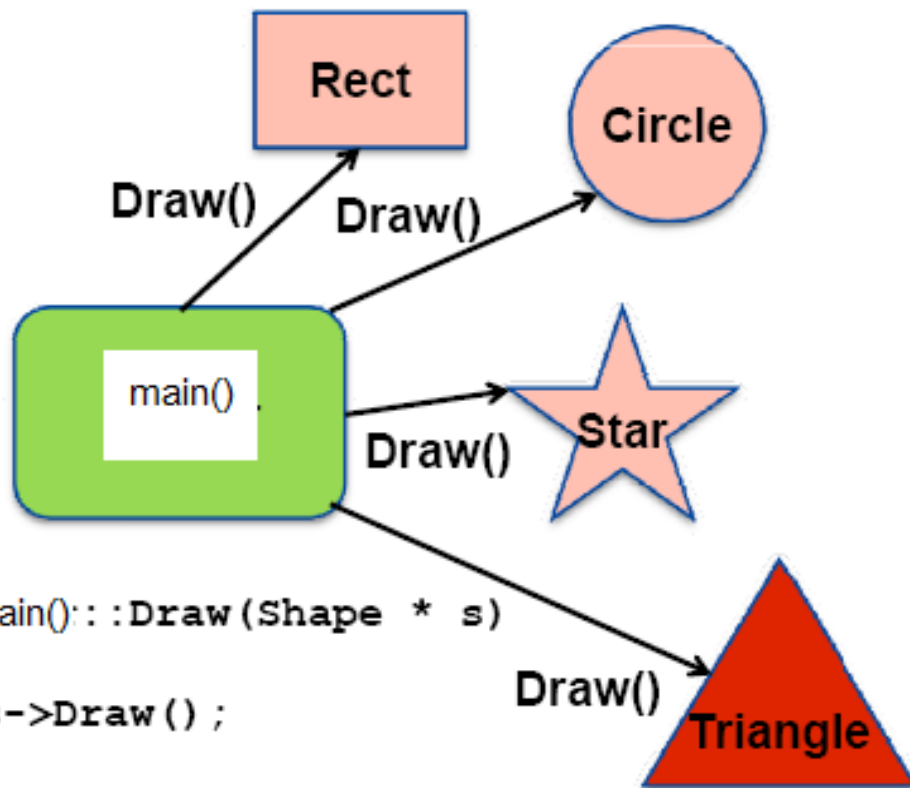


本讲重点分析

- 所谓**多态**，就是指向不同的对象发送**同一消息**时，不同的对象在接收时会产生**不同的行为**
- 函数重载、运算符重载、类型转换属于**静态多态技术**
- 为了把对象的行为留到程序运行时动态确定，需要使用**动态多态技术**，通过**虚函数**实现
- 包含虚函数的基类，其**析构函数**最好也声明为虚函数
- 引入**多态性的好处**：类的设计者与类的使用者分工协作
- 引入**抽象基类**的意义：抽象基类体现了本类族中各类的共性，把本类族中共有的成员函数集中在抽象基类中声明，作为本类族的公共接口。从抽象基类派生出的多个类具有共同的接口

本讲重点分析



多态的真正威力在于：你可以事先设计一些使用其它模块的代码，对模块的实现细节却可以一无所知。这样，一个模块的修改乃至功能的添加都不再会影响到其它模块的代码了。

类的使用者与类的设计者分离。程序员的注意力集中在处理普遍性，而到执行环境中再处理特殊性。多态性把操作的细节留给类的设计者(多为专业人员)去完成，而让程序人员(类的使用者)只需要做一些宏观性的工作，告诉系统做什么，而不必考虑怎么做，极大地简化了应用程序的编码工作。

第8次作业必做题目2道（10周末交）

第1题要求：在第7次作业第2题基础上，对程序进行如下改进从teacher类派生Prof（教授）类，从Prof类派生Female-Prof类（女教授）；对这3类分别定义一个同名函数（PAY（））来计算3类人员的工资；并要求在main()函数中，采用指向Teacher类指针变量来调用这3个类中的PAY（）函数。这3类人员的工资如何发放，权利交给同学；但要求不一样。建议用多态性来实现程序功能。

第2题：声明一个哺乳动物Mammal类，再由此派生出狗Dog类，二者都定义Speak（）成员函数（函数内容自己确定），基类中定义为虚函数。声明一个Dog类的对象，调用Speak（）函数，观察运行结果。

选做题：声明一个Shape抽象类，在此基础上派生出Rectangle和Circle类，二者都有GetArea()函数计算对象的面积，GetPerim()函数计算对象的周长。给定部分程序代码，根据题意要求和基类代码，完成整个程序？

```
#include<iostream.h>

class Sharp
{
public:
    Sharp() {}
    ~Sharp() {}
    virtual float GetArea()=0;
    virtual float GetPerim()=0;
};
```

面向对象程序设计特点总结

- 抽象性：类是对象的抽象，对象是类的具体实例；类模板是类的抽象，类是类模板的具体实例；基类是派生类的抽象，派生类是基类的具体实例等；
- 封装性：类成员的3种访问属性；公有接口与私有实现的分离；类申明与成员函数定义的分离；构造函数；
- 继承性：3种继承方式，派生类成员的4种访问属性；派生类的构造函数；多重继承中函数同名问题与虚基类；
- 多态性：函数重载、运算符重载、多层派生的函数同名问题与虚函数；纯虚函数与抽象类。

面向对象程序设计关键技术总结

- 对象成员的3种访问方式；6种构造函数；对象指针（this）；数据保护的6种常类型；对象动态建立与释放；对象的复制（浅拷贝与深拷贝问题）；静态成员，友元函数；
- 运算重载规则；成员函数与友元函数重载；类型转换函数与转换构造函数；
- 派生类的构造函数的实现方法；基类与派生类的兼容性问题；
- 虚函数的引用方法；虚析构函数；
- Cin和cout；文件操作与文件流；
- 异常处理。