

C++程序设计(II)

授课：毛国红

电话：13857144844（644844）

作业邮箱：1419470@QQ.com



浙江工业大学计算机学院



第7讲 类与对象

1

简单讨论

2

类/类的定义和使用

3

对象/对象的声明和使用

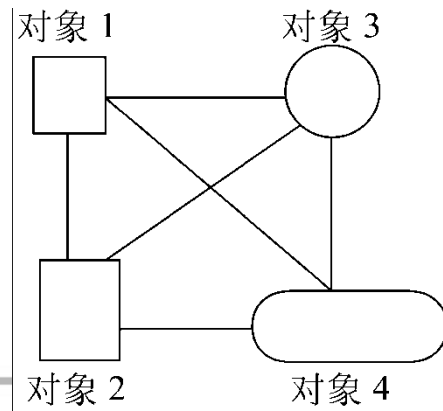
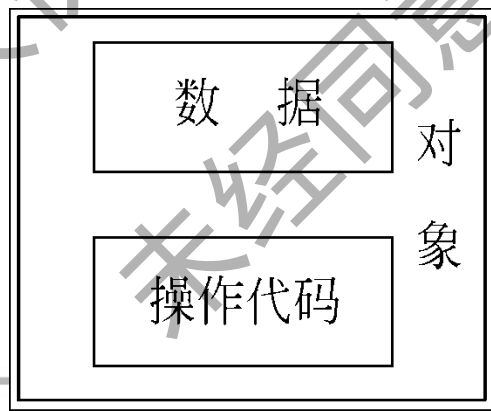
4

用类和对象构造程序的实例

第7讲 类与对象-对象/对象的声明和使用

❖ 对象的概念

- 任何事物都可看成对象(object)。对象是构成系统的基本单位。
- 任何一个对象都应当具有属性(attribute)和行为(behavior)这两个要素，它能根据外界给的信息进行相应的操作。
- 在一个系统中的多个对象之间通过一定的渠道相互联系。要使某一个对象实现某一种行为(即操作)，应当向它传递相应的消息。对象之间就是这样通过发送和接收消息互相联系的。

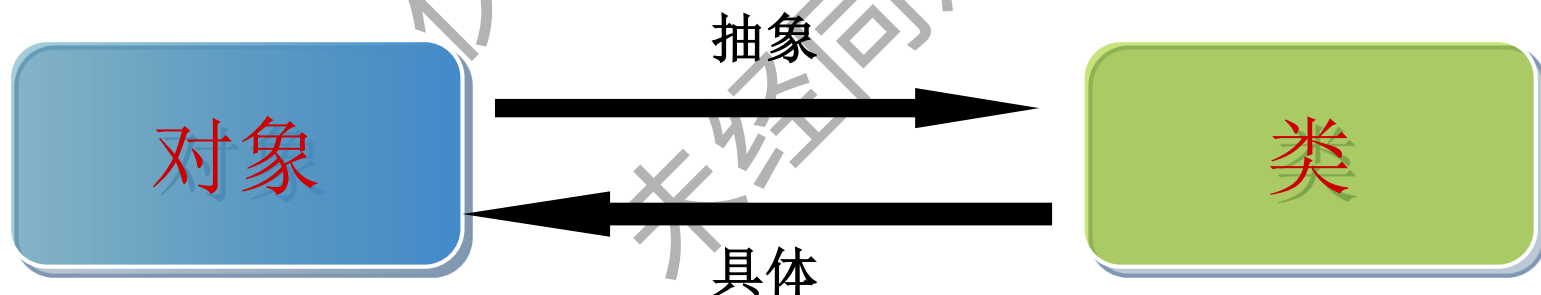




第7讲 类与对象-对象/对象的声明和使用

❖ 类与对象的关系

- 每一个实体都是对象。有一些对象是具有相同的结构和特性的。每个对象都属于一个特定的类型。类是关于一组结构与行为相似的对象共同特性的描述。
- 在C++中对象的类型称为类(class)。类代表了某一批对象的共性和特征。即：类是对象的抽象，而对象是类的具体实例(instance)。
- 在C++程序中，类是面向对象程序静态文本的概念，而对象是面向对象程序动态运行时的概念。





第7讲 类与对象-对象/对象的声明和使用

❖ 对象的声明

- 与普通变量声明相似
类类型 对象名;
- `pool apool;` 或者 `class_x obj1,obj2;`

❖ 对象的使用(在类外使用公有成员):

- 1、通过对象名和成员运算符访问对象中的成员

`apool.build();`

- 2、通过指向对象的指针访问对象中的成员

`pool *ptr=&apool;`

`ptr->build();`

`(*ptr).build();`

★类内的隐含对象指针**this**:

所有在类内使用类成员的行为都是通过**this**进行。

this可以省略。



第7讲 类与对象-对象/对象的声明和使用

例: **this**指针的使用

```
double pool::rail_area()  
{  
    return cir_area(radius+c)  
        - cir_area(radius); }  
}
```

```
double pool::rail_area()  
{  
    return this->cir_area((*this).radius+(*this).c)  
        - this->cir_area(this->radius);  
}
```

❖ 对象的使用(在类外使用公有成员):

- 1、通过对象名和成员运算符访问对象中的成员
- 2、通过指向对象的指针访问对象中的成员
- 3、通过对象的引用变量访问对象中的成员

```
pool &bpool=apool;  
bpool.build();
```



第7讲 类与对象-用类和对象构造程序的实例

❖ **首要问题：**如何设计类？

❖ **停车场收费：**有一个停车场，需要替管理员设计一个程序来解决停车收费的问题。

收费的具体标准：

- 1小时内免费；
- 1小时以上：
 - 超过**15**分钟不到**30**分钟按半小时收费；
 - 超过**30**分钟不到**1**小时按**1**小时收费；
 - 超过**1**小时按每小时**4**元收费；
- 暂不考虑超过当天的停车情况。若考虑则自己拓展，如累计每**24**小时可以一次性收费。



第7讲 类与对象-用类和对象构造程序的实例

❖ 从应用的需求出发，面向对象的分析：

- 停车场：位置，大小(可以停几辆车)，收费标准，几个管理员.....
- 管理员：性别，年龄，籍贯，身高，体重，受教育程度，.....
- 汽车：品牌，颜色，轮子个数，.....
- 时钟/时间：品牌，是否正常工作，时，分，秒，.....

❖ 对一个为管理员设计的收费系统，哪个对象是需要关注的？



+

收费标准





第7讲 类与对象-用类和对象构造程序的实例

❖ 如何设计时钟类?

- 要如何使用时钟?

有车进入停车场的时候看一下时钟登记;
有车离开停车场的时候看一下时钟登记.
→ 计算停车时间, 结合收费标准收费。

- 时钟是什么样的? → 模拟数字时钟



属性/数据

有: 时, 分, 秒 (hh:mm:ss)

行为/函数

可以: 显示时间

设置时间 → 调整时间(调整时, 调整分, 调整秒)





第7讲 类与对象-用类和对象构造程序的实例

❖ 时钟类的设计

```
class clock{  
public:  
    void show_time();  
    void set_time();  
private:  
    int hour;  
    int minute;  
    int second;  
};
```



clock.hpp/clock.h



第7讲 类与对象-用类和对象构造程序的实例

❖ 时钟类的使用

```
#include "clock.hpp"
#include <iostream>
using namespace std;
int main()
{ clock arriveTime, leaveTime;
  arriveTime.set_time();
  arriveTime.show_time();
  leaveTime.set_time();
  leaveTime.show_time();
  停车时间=leaveTime - arriveTime;
  收费(停车时间);
  return 0;
}
```

➡ parking.cpp

计算停车时间
计算停车收费



第7讲 类与对象-用类和对象构造程序的实例

❖ 时钟类的使用

- 停车时间= $\text{leaveTime} - \text{arriveTime}$;
 $\text{parkingTime} = \text{leaveTime.diff}(\text{arriveTime});$
→ 类内的成员函数 $\text{double diff}(\text{clock } T);$
 $\text{double diff}(\text{clock } \& T);$
 $\text{double diff}(\text{const clock } \& T);$
→ 能不能使用 ?
 $\text{parkingTime} = \text{diff}(\text{leaveTime}, \text{arriveTime});$
对应函数原型 $\text{double diff}(\text{clock } T1, \text{clock } T2);$
- 收费(停车时间)
 $\text{double parkingFee}(\text{double } t);$ //普通函数



第7讲 类与对象-用类和对象构造程序的实例

❖ 时钟类的设计使用

```
class clock{
public:
    void show_time();
    void set_time();
    //double diff(clock T);
    //double diff(clock& T);
    double diff(const clock& T);
private:
    int hour;
    int minute;
    int second;
};
```

```
#include "clock.hpp"
#include <iostream>
using namespace std;
double parkingFee(double);
int main()
{
    clock arriveTime,leaveTime;
    double parkingTime,Fee;
    arriveTime.set_time();
    arriveTime.show_time();
    leaveTime.set_time();
    leaveTime.show_time();
    parkingTime=leaveTime.diff(arriveTime);
    Fee=parkingFee(parkingTime);
    cout<<"收费" <<Fee<<endl;
    return 0;
}
```



第7讲 类与对象-用类和对象构造程序的实例

❖ 时钟类的实现(1)

```
void clock::show_time()
{ cout<<hour<<":"<<minute<<":"
  <<second;

  cout<<endl;
}

void clock::set_time()
{ cin>>hour>>minute>>second;
}
```

```
class clock{
public:
    void show_time();
    void set_time();
    double diff(const clock& T);
private:
    int hour;
    int minute;
    int second;
};
```

- ★如何实现停车时间的计算? `double diff(const clock& T);`
需要考虑: 1.计时的模式(12小时 or 24小时)→检查输入数据
2.时间的规格化(转为分钟)→增加工具函数



第7讲 类与对象-用类和对象构造程序的实例

❖ 时钟类的实现(2)

```
.....  
void clock::set_time()  
{ do{  
    cin>>hour>>minute>>second;  
}while(hour<0||hour>24||  
    minute<0||minute>59||  
    seconde<0||second>59);  
}
```

```
double clock::diff(const clock& T)  
{ long d=normlize()-T.normlize()-60;  
  if(d<0) return 0;  
  int h=d/60,m=d%60;  
  if(m<15) return h;  
  if(m>=15&&m<30) return h+0.5;  
  if(m>=30&&m<60) return h+1;  
}
```

```
class clock{  
public:  
    void show_time();  
    void set_time();  
    double diff(const clock& T);  
private:  
    long normalize(); //需调整  
    int hour;  
    int minute;  
    int second;  
};
```

```
long clock::normalize()  
{ return hour*60+minute;  
}
```

C++程序设计 (II)

Thank You !



浙江工业大学计算机学院