面向对象与流

陈云帆

数据结构与算法 补充内容——C++特性简介

http://www.jpk.pku.edu.cn/pkujpk/course/sjjg

补充内容 面向对象与流

- 类与对象
  - 类的概念及基本语法
  - 默认函数——构造、析构、复制构造、赋值与取址
  - 特殊成员——this指针
  - 函数模板与类模板
- 流
  - 标准输入输出流
  - 流操纵算子
  - 文件输入输出流

函数模板

实际问题中的需要：
对不同类型数据可用的排序函数 sort

```
template<class T>
return-type sort(...T...)
```

函数模板

一个实际的输出函数：

```
template<class T>
void print( const T array[], int size){
    int i;
    for ( i =0; i<size; i++) cout<<array[i];
    return;
}
```

int a[10]; print(a,10);

函数模板

一个实际的输出函数：

```
template<class T1, class T2>
void print(T1 arg1, T2 arg2 , string s, int k)
    { cout<<arg1<<s<<arg2<<k<<endl;  return; }
```

类模板

- 为了多快好省地定义出一批相似的类,可以定义类模板,然后由类模板生成不同的类

- 数组是一种常见的数据类型，元素可以是：
  - 整数
  - 字符串
  - ……

类模板：在定义类的时候给它一个/多个参数,这个/些参数表示不同的数据类型。在调用类模板时，指定参数，由编译系统根据参数提供的数据类型自动产生相应的模板类。

类模板的定义

```
template <class T>//类模板的首部，声明类模板的参数
class Carray{
    T *ptrElement;
    int size;
public:
    Carray(int length);
    ~ Carray();
    int len();
    void setElement(T arg, int index);
    T getElement(int index);
};
```

使用类模板声明对象

```
Carray<int> arrayInt(50), *ptrArrayInt;
    //创建一个元素类型为int的Carray模板类，并声明该模板类
      的一个对象、以及一个指针。
```

不同模板参数产生的模板类，不是同一个类

参考文献

- 北京大学 郭炜、刘家瑛《程序设计实习》
  https://www.coursera.org/course/pkupop
- Prata, S. (2011). C++ primer plus. Addison-Wesley Professional.

数据结构与算法

谢谢聆听

国家精品课“数据结构与算法”
http://www.jpk.pku.edu.cn/pkujpk/course/sjjg/

张铭，王腾蛟，赵海燕
高等教育出版社，2008. 6。“十一五”国家级规划教材