C++简明教程

杜建成　潘金贵　编著

上海科学普及出版社

C++简明教程

杜建成　潘金贵　编著

上海科学普及出版社

(沪)新登字第 305 号

责任编辑　胡名正　郭子安

封面设计　毛增南

C++简明教程

杜建成　潘金贵　编著

上海科学普及出版社出版

(上海曹杨路 500 号　邮政编码 200063)

新华书店上海发行所发行　常熟高专印刷厂印刷

开本 787×1092　1/32　印张 7.75　字数 168000

1994 年 10 月第 1 版　1994 年 10 月第 1 次印刷

ISBN 7-5427-0912-7/TP·217　定价：8.70 元

内容提要

本书重点介绍C++语言与面向对象的程序设计方法有关的功能以及用C++语言进行面向对象程序设计的基本技巧。通过简洁明了的叙述和丰富的示例引导读者分析许多程序，从而培养读者解决实际问题的编程能力。每一章节后附有小结、习题程序和思考题，供读者自我测试和巩固知识。书中的示例程序和习题程序均经实际运行。本书适于用作计算机专业以及自学考试、函授教育的教材和参考书，以及广大工程技术人员的培训和自学教材。

前　言

C++是在C的基础上扩充而成的一种计算机程序设计语言，学习用C++语言进行程序设计的读者，一般应对C语言有足够的知识。本书将引导读者学习C++语言与面向对象的程序设计方法有关的高级功能以及用C++语言进行面向对象程序设计的技巧。

任何一种语言都是一种工具。人们常说，工欲善其事，必先利其器。反过来说，工之所以要利其器，全为了善其事。C++作为一种支持面向对象程序设计方法的计算机语言工具不可谓之不利，然而，要用它来开发可重用性、可移植性、可维护性均好的软件系统，主要还要靠对它的正确、熟练的运用。我们在编写本书时，改变传统教材的写法，并不过分注重语言知识和语言形式的严格描述，而是通过简洁的叙述和丰富的示例直接引导读者编制实际的程序，注重读者能力的培养。在写法上，我们遵循统一的风格，即每一小节在简洁叙述有关概念之后均给出有助于理解所述概念的例题、思考题和至少一个填空习题。在编排时，我们把习题集中到每一章的结尾，习题的编号基本上与章节对应。书中所给的例子程序以及填充后的习题程序都是完整的、可运行的。习题的参考解答作为附录给出，读者在做这些习题或思考题时，建议先不看给出的参考解答，而是自己动手做，然后与参考解答比较。读者只要弄清一些重要的基本概念，模仿所给的例子程序独立地完成所给的习题，就一定能大大提高自己的编程能力。

随着面向对象的程序设计方法和C++一类面向对象

的语言的流行，面向对象的软件系统设计、开发将成为今后的一大潮流。读者正确且熟练地掌握了面向对象的程序设计工具C++语言的功能和编程技巧后，必将在这一潮流中成为一个成功的弄潮儿。本书愿助你一臂之力。

编著者

1994年8月于南京

内容提要

本书重点介绍C++语言与面向对象的程序设计方法有关的功能以及用C++语言进行面向对象程序设计的基本技巧。通过简洁明了的叙述和丰富的示例引导读者分析许多程序，从而培养读者解决实际问题的编程能力。每一章节后附有小结、习题程序和思考题，供读者自我测试和巩固知识。书中的示例程序和习题程序均经实际运行。本书适于用作计算机专业以及自学考试、函授教育的教材和参考书，以及广大工程技术人员的培训和自学教材。

目　录

第一章　C++ 语言概述

本章我们将回顾 C++ 语言的发展过程，介绍什么是面向对象的程序设计方法(OOP)，并以复数型(complex)的程序为例来说明类、运算符的重载等几个最重要的 OOP 概念。

§1.1　C++ 是一种什么样的语言

本节我们先回顾一下 C++ 语言的发展过程，然后介绍什么是面向对象的程序设计方法(OOP)。

1.1.1　C++ 语言的发展过程

C++ 是美国贝尔实验室的 Bjarne Stroustrup 于 1980 年开发出来的一种面向对象的程序设计语言。开始的时候，叫作"带有类的 C(C with classes)"，1983 年，Rick Mascitti 将它命名为 C++。自从 1985 年由 AT&T(美国电话电报公司，贝尔实验室是它的子公司)正式公布以来，至今已有越来越多的程序员转向用 C++ 语言编程，它很有可能成为今后最为流行的一种计算机程序设计语言。

C++ 语言是在迄今最为普及的面向过程的 C 语言的基础上引入面向对象的程序设计概念后扩充而成的，在它的设计过程中，从 Simula 67 语言中吸取了类的概念，还从 Algol 68 语言中吸取了运算符重载的概念。

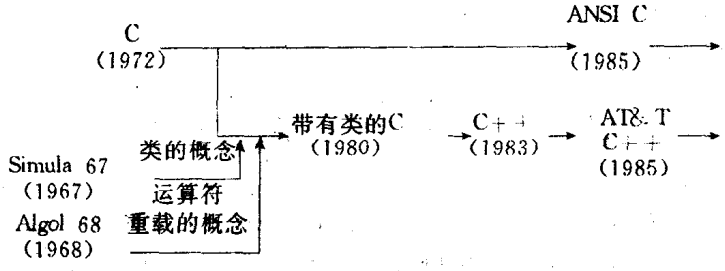

图1.1 C++语言的发展过程

1.1.2 什么是面向对象的程序设计方法(OOP)

迄今所流行的程序设计语言大多是所谓的"过程式语言",或称"面向过程的程序设计语言",如人们所熟知的BASIC、FORTRAN、COBOL、PASCAL、C等等高级语言均属此类。

用这类语言编制程序的一个特点是:解决某一个问题的数据和对数据进行操作的过程(或函数)是独立设计的,以对数据进行操作的过程流(即所谓处理流程)作为程序的主体。

与此不同,面向对象的程序设计方法(Object - Oriented Programming,本书以下简称OOP)却是把数据和对数据进行的操作作为一个整体定义成抽象数据类型(Abstract Data Type,简称ADT),程序的主体是这些ADT所说明的变量(这种变量我们称之为对象)。

面向对象一词是1981年由D. Robson首先使用的,面向对象的程序设计语言中,最早且最著名的是1972年由Xerox公司开发出来的Smalltalk语言。但是,由于该语言不是在现有语言的基础上发展起来的,而是一种全新的语言,至今尚未

真正普及开来。

C＋＋语言继承了C语言的语言规格，是在它的基础上引入面向对象的概念后扩充而成的，因此，极易为用惯了C语言的程序员所接受。此外，C＋＋语言还有Smalltalk语言所不具有的其他一些优良特色，大有取而代之之势。

§1.2 OOP概要

本节我们将以复数型(complex)的程序为例来说明类、运算符的重载等几个最重要的OOP概念。

1.2.1 一个简单的OOP程序例子

用C＋＋语言和面向对象的设计方法编制的复数型(complex)的程序清单如下：

```
/**************************/
/*复数型程序清单            */
/**************************/
#include <stdio.h>

class complex {                //复数类的定义
    double x,y;                //成员数据
    public:
    complex (double r=0,double i=0)        //构造函数
        {x=r;y=i;}
    void print()                           //成员函数
    {
```

```
        printf("(%5.2f,%5.2f)\n",x,y);
    }
    friend complex operator+(complex a, complex b)
                                        //运算符形式函数
    {
        return (complex(a.x+b.x, a.y+b.y));
    }
    friend complex operator-(complex a, complex b)
    {
        return (complex(a.x-b.x, a.y-b.y));
    }
};
void main(void)
{
    complex a(2,3), b(5,2), c,d;      //对象说明
    c=a+b;
    d=a-b;
    c.print();
    d.print();
}
```

试执行以上程序，看看其结果是否为：

(7.00, 5.00)

(-3.00, 1.00)

这个程序主要做了下列三件事：

① 定义类

② 说明对象

③ 调用成员函数对成员数据进行处理

1.2.2 抽象数据类型(类)和封装性

前面定义了一个对复数进行处理的抽象数据类型 complex，对于 complex 型的变量 a，b(由抽象数据类型所说明的变量习惯上称作对象，有时，在 OOP 术语中，类和由类所说明的变量都称为对象，后者又称为前者的实例)，是用

```
c=a+b;
```

这个语句进行复数型加法运算的，显示复数 c 的内容用的则是

```
c.print();
```

这个语句。

在 C ++ 语言中，抽象数据类型是用 class(类)如下定义的：

```
class  类名{
       成员数据
       成员函数
};
```

因此，我们可以说，C ++ 语言中的类是一种抽象数据类型，它是由成员数据和对成员数据进行操作的成员函数构成的。

类的成员数据，只能像 c.print() 这样通过成员函数进行存取，而不能在类的外部随意进行存取。这样一来，成员数据和成员函数二者就有机地结合在一起了，成员数据对类的外部是屏蔽的，这就叫作封装性(encapsulation)。

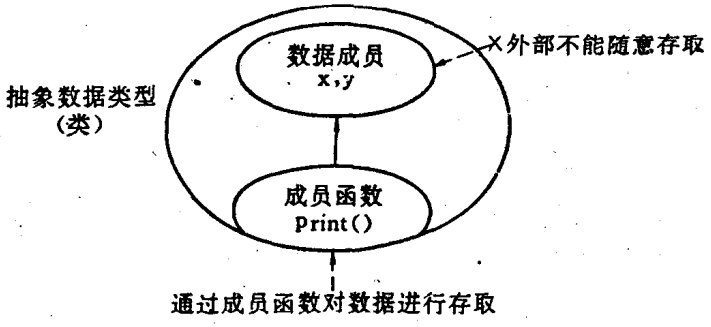

图 1.2　类和封装性

1.2.3　构造函数

当我们像

```
complex a(2,3);
```

这样进行 complex 型对象说明时，对象 a 的成员数据的初值设定是由构造函数完成的。

定义在类的内部，与类名同名的特殊成员函数叫作构造函数，它是用传递给它的形式参数的数据对对象的成员数据进行初值设定的。在对象说明和类型转换的时候，构造函数都将发挥它的作用。

1.2.4　运算符的重载

为了用像 a＋b，a－b 这样的形式进行加减运算，必须对运算符＋和－进行重新定义，这就叫作运算符的重载。所谓重载，就是使得一个运算符(或函数)身兼若干种不同的功能。在本例中，运算符＋和－是用 operator＋和 operator－这样的函数进行重载的，这样的函数叫作运算符形式函数。