

21世纪大学计算机专业教材

C++ 面向对象 程序设计 习题解答与实验指导

徐宏喆 李文



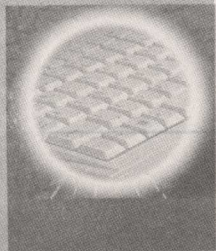
西安交通大学出版社
XI'AN JIAOTONG UNIVERSITY PRESS

21世纪大学计算机专业教材

C++ 面向对象 程序设计

习题解答与实验指导

徐宏喆 李文 彭晓晖 李越



西安交通大学出版社

XI'AN JIAOTONG UNIVERSITY PRESS

· 西安 ·

内容简介

本书是《C++面向对象程序设计》(西安交通大学出版社出版)教材的配套习题集和实验指导,主要为教材各章的课后习题提供参考答案,并为本科生上机实验提供指导。本书旨在通过难易适度的习题,加深学生对C++语言中重要概念的理解,培养学生良好的编程风格,并使能够熟练地应用VC 6.0集成开发环境,掌握常用的调试方法和工具。

本书分为两大部分,第一部分为与《C++面向对象程序设计》教材相配套的习题与解答,参考答案中所有的程序都经过编者上机验证,确保能够编译通过;第二部分为实验指导,一共包括八个实验,指导学生熟悉VC 6.0开发环境,学会建立和调试C++程序,加深对C++编译过程的理解。

图书在版编目(CIP)数据

C++面向对象程序设计习题解答与实验指导/徐宏喆等
编著. —西安:西安交通大学出版社,2007.8
ISBN 978-7-5605-2492-4

I. C… II. 徐… III. C语言-程序设计-高等学校-教学参考资料 IV. TP312

中国版本图书馆CIP数据核字(2007)第101848号

书 名	C++面向对象程序设计习题解答与实验指导
编 著	徐宏喆
出版发行	西安交通大学出版社
地 址	西安市兴庆南路10号(邮编:710049)
电 话	(029)82668357 82667874(发行部) (029)82668315 82669096(总编办)
印 刷	西安市新城区兴庆印刷厂
字 数	200千字
开 本	787mm×1092mm 1/16
印 张	8.5
版 次	2007年8月第1版 2007年8月第1次印刷
书 号	ISBN 978-7-5605-2492-4/TP·496
定 价	12.00元

版权所有 侵权必究

前 言

C++是一种含C语言子集的高效程序设计语言,它既可以进行过程化程序设计,也可以用于面向对象的程序设计。本书是《C++面向对象程序设计》(以下简称教材)的配套用书,包括习题与解答和实验指导两大部分。

第1部分,习题与解答。该部分与教材的章节安排完全对应,作者根据多年的教学经验和实际需要,精选了一些习题并给出了参考解答。这些习题难度适中,内容紧扣教材,以考查学生基础知识为主。对于书中的题目,不仅给出了参考答案,而且阐明了解题思路。本书对每一道编程题目都进行了注释,力求使程序结构清晰易懂。通过本书的学习,能够较全面地掌握C++语言的相关知识,并能较快地提高自己的编程水平。本书提供的答案仅供参考,学生也可按自己的思路另行编程。

第2部分,实验指导。该部分共包括八个实验,每个实验四学时。

实验1 VC++开发环境简介。本实验介绍了VC 6.0集成开发环境。

实验2 简单C++程序的实现。教读者在VC 6.0环境下编写简单的C++风格的程序,以及new和delete运算符的使用。

实验3 复杂数据类型的实现。在引入类的概念前,使读者先熟悉结构体的使用。

实验4 类和对象。指导读者创建类和对象,从这次实验开始真正进入面向对象编程的实践。

实验5 继承与派生。帮助读者加深对继承和派生概念的理解。

实验6 重载与文件I/O。帮助读者加强对重载和虚函数概念的理解,学习文件I/O流函数的使用。

实验7 函数和类模板的实现。指导读者使用模板这一面向对象程序设计中特有的机制。

实验8 Windows编程实验。简单介绍Windows应用程序的编制方法。

以上每个实验都分为实验目的、实验任务、实验内容和步骤以及实验小结四个单元,实验中穿插介绍了C++应用程序的创建过程、程序的调试方法和技巧,还有常用的调试工具等。相信读者认真做完每个实验后,将会使自己动手编制C++程序的能力大大提高。

书中所有程序都经过作者上机调试,可以确保编译通过。

本书可以作为大专院校相关专业学生的参考用书,特别适合于计算机本科和专科的学生、参加国家高等教育自学考试的考生、参加计算机等级考试的考生以及计算机专业人员使用。

由于C++语言知识面广、内容深,加以时间仓促,再加上作者水平有限,书中不足之处在所难免,恳请读者批评指正。

编者

2007年6月

目 录

第 1 部分 习题解答

前言

第 1 章 绪论	(2)
本章要点.....	(2)
本章目标.....	(2)
本章重点.....	(2)
本章难点.....	(2)
习题及解答.....	(3)
第 2 章 面向对象的程序设计	(5)
本章要点.....	(5)
本章目标.....	(5)
本章重点.....	(5)
本章难点.....	(5)
习题及解答.....	(5)
第 3 章 C++ 语言基础	(8)
本章要点.....	(8)
本章目标.....	(8)
本章重点.....	(8)
本章难点.....	(9)
习题及解答.....	(9)
第 4 章 对象和类	(18)
本章要点.....	(18)
本章目标.....	(18)
本章重点.....	(18)
本章难点.....	(18)
习题及解答.....	(18)
第 5 章 静态成员与友元	(43)
本章要点.....	(43)
本章目标.....	(43)

本章重点	(43)
本章难点	(43)
习题及解答	(43)
第 6 章 派生类与继承	(50)
本章要点	(50)
本章目标	(50)
本章重点	(50)
本章难点	(50)
习题及解答	(50)
第 7 章 多态性	(60)
本章要点	(60)
本章目标	(60)
本章重点	(60)
本章难点	(60)
习题及解答	(60)
第 8 章 模板	(77)
本章要点	(77)
本章目标	(77)
本章重点	(77)
本章难点	(77)
习题及解答	(77)
第 9 章 I/O 流	(86)
本章要点	(86)
本章目标	(86)
本章重点	(86)
本章难点	(86)
习题及解答	(86)

第 2 部分 实验指导

实验1 VC++开发环境简介(4 学时)	(90)
一、实验目的	(90)
二、实验任务	(90)
三、实验内容及步骤	(90)
四、实验小结	(97)

实验2 简单 C++ 程序的实现(4 学时)	(98)
一、实验目的	(98)
二、实验任务	(98)
三、实验内容及步骤	(98)
四、实验小结	(101)
实验3 复杂数据类型的实现(4 学时)	(102)
一、实验目的	(102)
二、实验任务	(102)
三、实验内容及步骤	(102)
四、实验小结	(104)
实验4 类和对象(4 学时)	(105)
一、实验目的	(105)
二、实验任务	(105)
三、实验内容及步骤	(105)
四、实验小结	(108)
实验5 继承与派生(4 学时)	(109)
一、实验目的	(109)
二、实验任务	(109)
三、实验内容及步骤	(109)
四、实验小结	(113)
实验6 重载与文件 I/O(4 学时)	(114)
一、实验目的	(114)
二、实验任务	(114)
三、实验内容及步骤	(114)
四、实验小结	(116)
实验7 函数和类模板的实现(4 学时)	(117)
一、实验目的	(117)
二、实验任务	(117)
三、实验内容及步骤	(117)
四、实验小结	(118)
实验8 Windows 编程实验(4 学时)	(120)
一、实验目的	(120)
二、实验任务	(120)
三、实验内容及步骤	(120)
四、实验小结	(128)

第 1 部分 习题解答

- 第 1 章 绪论
- 第 2 章 面向对象的程序设计
- 第 3 章 C++ 语言基础
- 第 4 章 对象和类
- 第 5 章 静态成员与友元
- 第 6 章 派生类与继承
- 第 7 章 多态性
- 第 8 章 模板
- 第 9 章 I/O 流

第 1 章 绪论

【本章要点】

本章从一开始就强调了程序设计和编程语言的关系。程序设计就像施工蓝图,描述了解决方案的步骤和布局,它的表现形式多样,可以是图表也可以是文字,甚至可以只是脑子中的想法;而编程语言就像施工材料,它是编程设计方案的载体,用来将设计方案描述成电脑可以识别的代码。

程序设计经历了从结构化设计方法到面向对象的设计方法过渡。编程语言也随着设计方法的变更而变更。所谓编程语言支持某种设计方法,指的是这种语言的语法规则能够完全地表示设计方法的内容。随着面向对象方法中新概念的不断出现,传统的 C 语言已经不能方便地表达这些内容了,于是 C++ 作为对 C 语言的扩展出现了。

面向对象最基本的概念就是对象。对象是对客观世界中实体的认识,对象之间也是有联系的,具有相同类型属性和操作的对象可以被抽象为类,类是对象的模板。面向对象有三个特性:

- (1)封装性。对象中的属性和操作都被隐藏起来,只把接口暴露给外界。
- (2)继承性。对象间的一种层次关系,如某个对象继承了另一个对象就具有了那个对象所有的属性和方法。
- (3)多态性。同一个操作作用在不同对象时表现出的不同行为。

【本章目标】

1. 掌握本章的基本概念,能在以后的学习过程中熟练运用这些概念;
2. 掌握面向对象的概念;
3. 理解面向对象程序设计的重要意义;
4. 掌握面向对象程序设计的思想原则。

【本章重点】

1. 面向对象程序设计的概念;
2. 采用面向对象程序设计方法的重要意义;
3. 面向对象的程序设计思想。

【本章难点】

1. 面向对象程序设计思想的建立;

2. 由结构化程序设计向面向对象程序的转换。

【习题及解答】

1. 软件开发方法包括哪些？

答：软件开发方法主要包括面向过程开发方法和面向对象开发方法。

2. 试比较面向过程的开发方法和面向对象的开发方法的优缺点。

答：面向过程的结构化方法提出了一组提高软件结构合理性的准则，如分解和抽象、模块的独立性、信息隐蔽等。该方法的主要优点是快速、自然和方便。但是它十分强调前期工作的完美性，如在此阶段有不周到的地方，将给后期的代码编写带来巨大的麻烦，往往可能由于没有弄清用户的需求而使整个开发重来；软件开发过程是个充满回溯的过程，瀑布模型却将其分割为独立的几个阶段，不能从本质上反映软件开发过程本身的规律；此外，过分强调复审并不能完全避免较为频繁的变动。

面向对象开发方法具有以下优点。

(1)与人类的思维习惯类似。面向对象技术对问题空间进行自然分割，以更接近人类思维的方式建立问题域模型，以便对客观实体进行结构模拟和行为模拟，从而使设计出的软件尽可能直接地描述现实世界。

(2)具有良好的稳定性。运用面向对象技术，软件开发时间短，效率高，所开发的程序更稳定。由于面向对象编程的可重用性，可以在应用程序中大量采用成熟的类库，从而缩短了开发时间。

(3)可重用性好。应用程序更易于维护、更新和升级。继承和封装使得应用程序的修改所带来的影响更加局部化。

3. 试列出面向对象的编程方法的基本特征。

答：面向对象的编程方法有四个基本特征。

(1)抽象。抽象就是忽略一个主题中与当前目标无关的方面，以便充分地注意与当前目标有关的方面。

(2)继承。继承是一种联结类的层次模型，并且允许和鼓励类的重用，它提供了一种明确表述共性的方法。这种特征体现了一般和特殊的关系。继承性很好地解决了软件的可重用性问题。

(3)封装。封装是面向对象的特征之一，是对象和类概念的主要特性。封装是把过程和数据包围起来，对数据的访问只能通过已定义的界面。封装保证了模块具有较好的独立性，使得程序维护修改更为容易。

(4)多态性。多态性是指允许不同类的对象对同一消息作出响应。多态性语言具有灵活、抽象、行为共享、代码共享的优势，它很好地解决了应用程序的函数同名问题。

4. 面向对象开发技术有哪些优点？

答：面向对象开发方法具有以下优点。

(1)与人类的思维习惯类似。面向对象技术对问题空间进行自然分割，以更接近人类思维的方式建立问题域模型，以便对客观实体进行结构模拟和行为模拟，从而使设计出的软件尽可能直接地描述现实世界。

(2)具有良好的稳定性。运用面向对象技术，软件开发时间短，效率高，所开发的程序更稳定。由于面向对象编程的可重用性，可以在应用程序中大量采用成熟的类库，从而缩短了开发

时间。

(3)可重用性好。应用程序更易于维护、更新和升级。继承和封装使得应用程序的修改所带来的影响的更加局部化。

5. 什么是 VC++?

答:VC++是目前较为流行的 C++集成开发环境 (IDE),该开发环境是由 Microsoft 公司开发的。该开发环境除了提供标准的 C++语言的库函数以外,还提供了 MFC(微软基础类库),方便用户创建一些高级特性的类,在一定程度上减少了开发人员写任何一个类都要从头开始的重复劳动。

第2章 面向对象的程序设计

【本章要点】

本章介绍了面向对象的几个基本概念。

面向对象最基本的概念就是对象。对象是对客观世界中实体的认识,对象之间也是有联系的,具有相同类型属性和操作的对象可以被抽象为类,类是对象的模板。

面向对象有三个特性:

(1)封装性。对象中的属性和操作都被隐藏起来,只把接口暴露给外界。

(2)继承性。对象间的一种层次关系,如某个对象继承了另一个对象就具有了那个对象所有的属性和方法。

(3)多态性。同一个操作作用在不同对象的时候表现出的不同行为。

【本章目标】

1. 掌握对象与类的概念;
2. 理解对象的三大特性:封装性、继承性和多态性。

【本章重点】

1. 对象与类的概念;
2. 对象的三大特性。

【本章难点】

对象和类的概念和思维模式的建立。

【习题及解答】

1. 什么是程序设计,什么是编程语言,二者之间是什么关系?

答:程序设计是对所求问题的解决方案的描述,主要是对布局和执行步骤的描述。编程语言是设计方案的载体,编程语言按照一定的语法规则来表达设计方案。程序设计和编程语言的关系就像设计图纸和施工材料之间的关系一样,设计图纸描述了房子的布局 and 施工步骤,然后按照图纸将施工材料组织起来,最终完成整个工程。程序设计和编程语言是一对多的关系。

2. 结构化程序设计的思路是什么?

答:结构化的设计在系统布局方面将软件系统划分为若干功能模块,各模块实现独立的功能,再由各模块连接组合构成相应的软件系统。结构化的设计在程序执行顺序方面是严格的

从上到下的逐级调用。

3. 什么是面向对象程序设计?

答:面向对象是一种人理解、分析客观世界的方法,把面向对象应用到设计中就是为了设计更贴近人的思考方式,利用人描述客观问题的方式来生成程序设计对所求问题的表达。人在认识事物的时候总是把事物和对事物能做的操作放在一起来处理,同样的,面向对象设计也是把数据和对数据的操作放在一起设计的,这样就避免了结构化设计中数据和操作分离带来的一系列问题。

4. 请简述类与对象的概念。

答:对象是客观世界中的一个实体,包括名字、属性和操作三个方面。

类可以说是对象的模型,用一个模型便能建立许多类似的对象,它有以下三个特点:

(1) 类是抽象的。

(2) 类的属性其实是一个范围,是不确定的。

(3) 类和对象都有操作,且二者的操作是相同的。

5. 类和对象之间的关系是什么?

答:类与对象的关系就是抽象与具体的关系。类是一个框架,是对具有同类型属性和操作的对象组的抽象;对象则是类的实例化,给类这个框架中的属性填上一组确定的值就可以得到一个对象。

6. 面向对象中,对象的交互机制是什么?

答:对象的交互机制是消息传送机制,但这种传递只是抽象的叫法,并不是真的发送一个完整的消息,一个对别的对象中的方法的调用也是消息的传送。这里的消息发送并不强调发送形式,消息发送只是一种通知,告诉其他对象要使用其中的方法。

7. 面向对象的三大特性是什么?

答:(1)封装性:对象对外隐藏自己的数据和操作,同时将这些数据和操作实现的功能抽象出来,作为其他对象请求服务的接口,这就是封装与抽象。

(2)继承性:对象具有一些相同的属性,但又具有一些本身特有的属性,继承就是对这样的两个对象之间所具有的关系的描述。

(3)多态性:在程序设计中,多态是指同样的消息被不同类型的对象接收时导致完全不同的行为,是对类的特定成员函数的再抽象。

8. 请举例说明什么是对象中数据的抽象和封装。

答:略。

9. 请举例说明现实生活中继承的概念。

答:略。

10. 什么是多继承? 举例说明。

答:略。

11. 继承会破坏数据的封装性吗?

答:(1)基类的 private 数据成员对派生类隐藏,此时不会破坏数据的封装性。

(2)基类的 protected 成员可以被基类的成员函数和友元以及派生类的成员和友元来访问。派生类成员可以通过成员名就可以引用基类的 public 成员和 protected 成员,从而破坏了封装性。

12. 什么是多态? 多态有几种实现方法, 分别是什么?

答: 多态是指同样的消息被不同类型的对象接收时导致完全不同的行为, 是对类的特定成员函数的再抽象。在 C++ 中, 多态主要有两种实现方法: 重载和虚函数。

第3章 C++语言基础

【本章要点】

本章首先介绍了C++程序的结构,C++程序由头文件、主函数、函数变量的定义以及注释组成。在注释方面,C++增加了“//”符号的注释方法,使用很方便,同时C++增加了新的输入输出符号 cout 和 cin,需要在头文件中添加<iostream.h>。

任何语言都有属于自己的数据类型,在C++中,数据类型分为基本数据类型和用户定义数据类型,基本数据类型有六种,前五种在内存中占一定的空间,而空类型仅作为一个语法类型存在。在用户定义数据类型中,着重介绍了联合和枚举两种类型。

表达式由各种数据和操作符构成,表达式的分类是按照操作符的类型分类的,有以下几种:算数表达式、赋值表达式、增量表达式、关系表达式、逻辑表达式、条件表达式、逗号表达式。它们之间具有一定的优先级差别。

表达式或函数加上分号,就可以构成一个语句,语句是组成C++程序的基本单元。语句中赋值语句和控制语句(条件、分支、循环)的用处最广。函数是程序的功能模块,一个函数完成一定的功能,函数在使用前必须声明或定义。

const 是常量修饰符,用 const 修饰的是常量,其值不可以被更改。const 和指针搭配使用分三种情况:指向常量的指针、常指针、指向常量的常指针。

内存分配管理的任务一般由系统自动完成,但在C++中,程序员可以通过运算符 new 申请任意大小的内存,通过运算符 delete 来释放曾经申请的内存空间,这种内存分配方式叫做动态内存分配。

在编程过程中,经常出现变量同名的问题。在C++中,当变量同名时,函数中的操作总是使用函数中的局部变量。如果希望输出全局变量的话,就需要使用作用域运算符::。

引用就是给一个变量起个别名,对引用的操作就是对变量的操作,反之也一样。引用的使用在形式上简单,同时具有指针的功能,C++中提倡使用引用。

【本章目标】

1. 掌握本章的基本概念,能在各节的学习中熟练运用这些概念;
2. 掌握C++编程基础;
3. 熟练使用C++相关语句;
4. 加深对C++语言的理解和掌握。

【本章重点】

1. C++语言基础;

2. 熟练运用 C++ 语言进行程序设计。

【本章难点】

1. C++ 语言的熟练掌握；
2. 进行较为复杂的程序设计。

【习题及解答】

一、填空题

1. 使用“/*”“*/”和“//”符号,为下面的程序加上注释(在你认为需要注释的地方加)。

```

1) #include <iostream. h>
2) void main(){
3)     int a,b,c;
4)     int max;
5)     cout<<"输入 3 个数字"<<endl;
6)     cin>>a>>b>>c;
7)     if(a>b)
8)         max = a;
9)     else max = b;
10)    if(max<c)
11)        max = c;
12)    cout<<"最大的数是 :"<<max<<endl;
13) }
```

参考答案:

/* 输入三个实数,输出其中最大值 */

```

1) #include <iostream. h>
2) void main(){
3)     int a,b,c;                //变量定义
4)     int max;
5)     cout<<"输入 3 个数字"<<endl;    //输入数字
6)     cin>>a>>b>>c;
7)     if(a>b)                    //max=a,b 中较大者
8)         max = a;
9)     else max = b;
10)    if(max<c)                    //max 记录三者最大数
11)        max = c;
12)    cout<<"最大的数是 :"<<max<<endl;    //输出最大数
13) }
```

2. C++ 中提供的新的 I/O 流所需要的头文件是什么? 补充完整下面程序的运行结果。

```
1) #include <iostream. h>
```



```

2) void main(){
3)     int temp;
4)     cout<<"请输入一个数:"<<endl;
5)     cin>>temp;
6)     cout<<"输入的数是:"<<endl;
7)     cout<<temp<<endl;
8) }

```

运行结果:

请输入一个数:

3.1415926

输入的数是:

答案:3.1415926

3. 下面程序的运行结果是什么?

```

1) #include <iostream.h>
2) void main(){
3)     int mark;
4)     int out;
5)     cout<<"输入 mark:"<<endl;
6)     cin>>mark;
7)     switch(mark)
8)     {
9)         case 1 : out =1;
10)        case 2 : out =2;
11)        case 3 : out =3;break;
12)        case 4 : out =4;
13)    }
14)    cout<<out<<endl;
15) }

```

运行结果:

输入 mark:

2 (回车)

分析:输入 2 后,进入语句 10) case 2 : out =2,但由于该语句后没有“break”,所以继续执行后面的语句 11) case 3 : out =3;break;,并跳出 switch 语句,此时 out=3,故结果为 3。

二、判断题:

1、下面的一段关于枚举的程序无法通过编译,哪里出错了?

```

1) #include <iostream.h>
2) void main()

```