



“十三五”高校计算机应用技术系列规划教材

丛书主编 谭浩强

C++面向对象程序设计 习题解答与上机指导 (第三版)

C++ 面向对象程序设计习题解答与上机指导 (第三版)

林小茶 陈维兴 编著



中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE

“十三五”高校计算机应用技术系列规划教材

丛书主编 谭浩强

C++面向对象程序设计

习题解答与上机指导

(第三版)

林小茶 陈维兴 编著

中国铁道出版社
CHINA RAILWAY PUBLISHING HOUSE

内 容 简 介

本书是中国铁道出版社出版的教材《C++面向对象程序设计（第三版）》（陈维兴、林小茶编著）的配套教材。

书中内容分为3篇，第一篇为“《C++面向对象程序设计（第三版）》习题和参考答案”，这部分内容对主教材中的每道习题都给出了详细的解答，这些习题是编者多年以来在教学中积累、收集并经过验证的习题，全部上机调试通过；第二篇为“C++语言上机实验环境介绍”，简单介绍了在 Visual C++ 2010 环境下调试与运行程序的方法，以方便读者快速熟悉上机环境；第三篇为“上机实验”，包括9套实验，每套实验都给出了实验目的、实验要求、实验步骤以及参考源程序。

本书适合作为高等院校各专业学生学习 C++ 基础课程的配套教材，也适合单独作为学习 C++ 语言的学习辅导书。

图书在版编目（CIP）数据

C++面向对象程序设计习题解答与上机指导 / 林小茶，
陈维兴编著. —3版. —北京：中国铁道出版社，
2017.1

“十三五”高校计算机应用技术系列规划教材
ISBN 978-7-113-10135-0

I. ①C… II. ①林… ②陈… III. ①C语言—程序设计—高等学校—教材 IV. ①TP312

中国版本图书馆 CIP 数据核字（2016）第 265011 号

书 名：C++面向对象程序设计习题解答与上机指导（第三版）
作 者：林小茶 陈维兴 编著

策 划：周海燕
责任编辑：周海燕 徐盼欣
封面设计：付 巍
封面制作：白 雪
责任校对：汤淑梅
责任印制：郭向伟

读者热线：(010) 63550836

出版发行：中国铁道出版社（100054，北京市西城区右安门西街8号）

网 址：<http://www.51eds.com>

印 刷：北京海淀五色花印刷厂

版 次：2004年5月第1版 2009年12月第2版 2017年1月第3版 2017年1月第1次印刷

开 本：787mm×1092mm 1/16 印张：10.5 字数：248千

书 号：ISBN 978-7-113-10135-0

定 价：29.00元

版权所有 侵权必究

凡购买铁道版图书，如有印制质量问题，请与本社教材图书营销部联系调换。电话：(010) 63550836

打击盗版举报电话：(010) 51873659

“十三五”高校计算机应用技术系列规划教材

主任：谭浩强

副主任：陈维兴 严晓舟

委员：（按姓氏音序排列）

安淑芝	安志远	陈志泊	金莹	李宁
李雁翎	李振银	林小茶	刘宇君	宁玲
秦绪好	曲建民	尚晓航	邵丽萍	沈洪
史秀璋	宋红	宋金珂	魏善沛	徐昊
姚怡	张玲	赵乃真		

信息技术的迅猛发展和对人类的深远影响使许多人目瞪口呆。在当今社会，每个人都在享受信息技术的成果，都在直接或间接地应用着信息技术。信息技术改变了世界面貌，改变了人类的生活方式，也改变了人们的思维方式。

早在 30 多年前，我国高等学校已经开始在全体大学生中开展计算机教育，计算机课程成为所有学生的必修课程，掌握计算机基本知识和应用能力成为对大学生的基本要求和毕业后求职的必要条件。大学中的计算机课程的设置和内容随着信息技术的发展而与时俱进，全社会对计算机基础教育的认识和支持率大大提高了，真是今非昔比。

高等学校中的计算机教育是由两部分组成的：一是计算机专业的教育，二是面向 95% 以上大学生的非计算机专业的计算机教育（称为高校计算机基础教育）。两者的培养目标、教学内容和教学方法是不同的。前者主要培养计算机专门人才，后者主要培养各行各业中广大的计算机应用人才。

过去，面向非计算机专业大学生的课程体系和教材往往是根据计算机专业的知识体系和教材来构建的，强调学科的系统性和完整性，强调理论，有的甚至是计算机专业课程的浓缩。事实证明这是不切合实际的，难以取得好的效果。

大学生为什么要学习计算机？答案是不言而喻的：首先是因为计算机有用。如果没有用何必学习它呢？现代社会离开计算机寸步难行，使用计算机将是现代人的一项基本技能。现在有些老年人（包括一些老年知识分子）由于不会使用计算机而感到处处不便，他们的意识、习惯和工作明显落后于时代，影响了他们对社会的贡献，这是很可惜的。

有人轻视应用，以为应用就是操作，因此认为“理论高级，应用低级”，这是一种误解。应用是分层次的，应用有初级、中级和高级之分。搞理论的人只是少数，绝大多数人将来是搞应用的。大到两弹一星，小到网上购物，在各个领域，都可以看到计算机应用无所不在，所有的人都可以尽其所能，大显身手。

计算机基础教育在本质上是计算机应用的教育，应当以应用为目的，以应用为出发点，应该以计算机应用为主线来构建课程体系，明确分析和提出应用能力的要求，编写出体现应用特点的教材。

计算机基础教育要综合考虑三方面因素：信息技术的发展、面向应用的需要以及科学思维的培养。在计算机基础教学中应当做到：讲知识、讲应用、讲方法，并且把三者紧密结合起来。首先要讲知识，因为知识是基础，应用和方法都需要知识支撑；同时要讲应用，因为计算机基础教育不是纯理论的学习，要面向应用，提高应用能力；还要注意向学生传授方法，使学生掌握规律，学会思考，培养科学的思维方法。

对多数人来说，学习计算机的目的是利用这个现代化工具处理面临的各种问题，使自己能够跟上时代前进的步伐，同时在学习过程中努力培养自己的信息素养，使自己具有信息时代所要求的科学素质，站在信息技术发展和应用的前列，推动我国信息技术的发展。

学习计算机课程有两种不同的方法：一是从理论入手；二是从实际应用入手。不同的人有不同的学习内容和学习方法。大学生中的多数人将来是各行各业中的计算机应用人才。对他们来说，不仅需要“知道什么”，更重要的是“会做什么”。因此，在学习过程中要以应用为目的，注重培养应用能力，大力加强实践环节训练，激励创新意识。

由于全国各地、各高等院校的情况不同，需要有不同特点的教材来满足不同学校、不同专业教学的需要。因此，在教材建设上应当提倡百花齐放，推陈出新。应当提供不同内容、不同风格的教材，供各校选用。

根据培养应用型人才的需要，我们组织编写了这套“‘十三五’高校计算机应用技术系列规划教材”。这套教材的特点是突出应用技术，面向实际应用，强调培养应用能力，学以致用。在选材上，根据实际应用的需要决定内容的取舍，重视实践环节，不涉及过多的理论，坚决舍弃那些现在用不到、将来也用不到的内容。在叙述方法上，采取“提出问题—解决问题—归纳分析”的三部曲，这种从实际到理论、从具体到抽象、从个别到一般的方法，符合人们的认知规律，且在实践过程中已取得了很好的效果。

本丛书可以作为应用型大学的计算机应用技术课程的教材，程度较高的高职高专学校也可从中选择适用的教材，也可作为广大计算机爱好者的自学教材。

本丛书由浩强创作室与中国铁道出版社共同策划，由有丰富教学经验的高校老师编写而成。中国铁道出版社以很高的热情和效率组织了这套教材的出版工作。在组织编写及出版推广过程中，得到各高等院校老师的大力支持，对此谨表衷心的感谢。

本丛书如有不足之处，请各位专家、老师和广大读者不吝指正。希望通过本丛书的出版，能为我国计算机教育事业的发展和人才培养做出贡献。

全国高等院校计算机基础教育研究会荣誉会长
丛书主编

谭浩强

第一版前言

FOREWORD

学过程序设计的人，都有一个体会，看别人编写的程序，好象挺明白的，但是一旦要自己编写一个程序，就感觉无从下手。这是因为程序设计是一门对实践环节要求很高的课程，初学者要想真正学会 C++ 语言程序设计，最重要的要抓住两个关键环节：一个是多做程序设计的习题，多编程，另一个就是多上机。写在纸上的程序是否正确，最好的办法就是上机验证一下。为此，我们编写了这本习题解答与实验指导。本书在对教材中的习题进行解答的同时，也对一些基本的程序算法和规则进行了详细的分析，希望能帮助学习者尽快掌握 C++ 语言程序设计的基本规则与编程规律，并能够熟练运用这些规则与技巧，编制出具有良好风格的应用程序，最终能够顺利地通过上机调试。

本书的主要内容分为三篇：第一篇“《C++面向对象程序设计》习题和参考答案”是对教材中的习题的详细解答；第二篇“C++语言上机实验环境介绍”介绍了 C++ 程序设计调试环境 Visual C++ 6.0 和 Turbo C++；第三篇“上机实验题与参考答案”安排了 10 套精心设计的实验，每个实验都给出了详细的实验目的、实验要求和实验步骤，帮助学习者掌握 C++ 程序设计的方法，并进一步加深对课程相关内容的理解与掌握。考虑到学生在上机操作中，经常遇到一些错误信息，但看不懂，不知如何修改，因此我们把一些常用的错误信息注释放在附录中，以供学生上机实验时使用。

提供习题解答和实验参考源程序的主要目的是给学习者一个参考和借鉴，作者在这里要强调一点，程序设计是一个创作的过程，解决一个实际问题的程序肯定不是惟一的，因此，在阅读本书的参考答案之前，希望读者已经独立思考过教材中的习题以及实验题目，这样才有助于程序设计水平的提高，并且不要把本书的参考源程序作为唯一的答案。本书中所有程序都经作者在 Visual C++ 6.0 或 Turbo C++ 3.0 上调试通过。

本书第一篇的内容由陈维兴教授和林小茶副教授共同编写，附录由陈维兴编写，第二篇和第三篇由林小茶编写。

在本书编写和出版过程中，全国计算机基础教育研究会会长谭浩强教授给予了指导和把关，在此表示衷心的感谢。

由于编者水平有限，书中难免还存在一些缺点和错误，殷切希望广大读者批评指正。

编者

2004 年 4 月

第二版前言

FOREWORD

本书是为了配合中国铁道出版社出版的教材《C++面向对象程序设计(第二版)》(陈维兴、林小茶)而编写的辅助教材。

由于第二版主教材对第一版主教材有一些改进,因此辅助教材也做了一些相应的改动。

(1)为了适应教学,我们更换了一些习题,我们认为,这些习题能更好地锻炼学生的程序设计能力。

(2)主教材中使用了标准C++的头文件改写了所有的源程序,本辅助教材也同样将习题解答中所有的源程序进行了修改,即:系统头文件不带后缀“.h”,使用系统库时使用命名空间std。

有一点提请读者特别注意:本教材保留了对Turbo C++ 3.0环境的描述,但是,该环境并不支持命名空间std,因此,如果在Turbo C++ 3.0环境下对程序进行调试的话,应该将程序做一定的修改。

```
#include <iostream>
using namespace std;
```

上面两行是使用命名空间的语句,要使程序能在Turbo C++ 3.0环境下执行,应将其修改为:

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

以笔者多年的教学经验,面向对象程序设计思想的建立不是一件容易的事情,许多学生都反映,听得懂,看得懂,就是自己一编程序,仍然感觉无从下手。本辅助教材就是想帮助学生解决编写程序的难题。因此,在选择例题时,特别注意配合主教材的内容循序渐进地对面向对象程序设计的能力进行训练。

请读者正确地使用本教材,不要在没有经过任何思考的情况下,直接阅读答案,这样做一点好处也没有,并且违背了我们编写本教材的初衷。

本书第一部分的内容由陈维兴和林小茶共同编写,附录由陈维兴编写,第二部分和第三部分由林小茶编写。

本书适合作为高等院校各专业学生学习C++基础的配套教材,也适合单独作为学习C++语言的学习辅导书。

由于编者水平有限,书中难免还存在一些缺点和错误,殷切希望广大读者批评指正。

编者

2009年10月

第三版前言

本书是为了配合中国铁道出版社出版的教材《C++面向对象程序设计（第三版）》（陈维兴、林小茶编著）而编写的辅助教材。

由于第三版主教材相对第二版主教材有一些改动，因此辅助教材也做了一些相应的变化。

（1）根据第二版教材出版后的使用情况，我们更换了一些习题。我们认为，这些习题能更好地体现面向对象程序设计的特性。

（2）因为主教材采用了新的调试环境：Visual C++ 2010，因此，习题部分的所有程序也同样在 Visual C++ 2010 环境进行了重新调试。

笔者在此还是要强调，如果想提高自己的编程能力，必须上机操作。

请读者正确地使用本书，不要在没有经过任何思考的情况下，直接阅读答案，这样做不仅没有益处，并且违背了我们编写本书的初衷。

以笔者多年的教学经验，面向对象程序设计思想的建立不是一件容易的事情，许多学生都反映，听得懂，看得懂，但编写程序时，仍然感觉无从下手。本书就是想帮助学生解决编写程序的难题。因此，在选择例题时，特别注意配合主教材的内容循序渐进地对面向对象程序设计的能力进行训练。

本书由林小茶、陈维兴编著。其中，第一篇的内容由陈维兴和林小茶共同编写，第二篇和第三篇由林小茶编写。

本书适合作为高等院校各专业学生学习 C++ 基础的配套教材，也适合单独作为学习 C++ 语言的学习辅导书。

由于编者水平有限，书中难免还存在一些疏漏和不足，殷切希望广大读者批评指正。

编者

2016年8月

目录

CONTENTS

第一篇 《C++面向对象程序设计（第三版）》习题和参考答案

第 1 章	面向对象程序设计	2
第 2 章	C++基础	7
第 3 章	类和对象（一）	16
第 4 章	类和对象（二）	25
第 5 章	继承与派生	45
第 6 章	多态性与虚函数	55
第 7 章	运算符重载	63
第 8 章	函数模板与类模板	73
第 9 章	C++的输入和输出	80
第 10 章	异常处理和命名空间	88
第 11 章	STL 标准模板库	91
第 12 章	面向对象程序设计方法与实例	95

第二篇 C++语言上机实验环境介绍

第 13 章	在 Visual C++ 2010 环境下调试与运行程序	102
--------	------------------------------------	-----

第三篇 上机实验

实验 1	C++基础练习	108
实验 2	类与对象（一）	111
实验 3	类与对象（二）	116
实验 4	继承与派生	122
实验 5	多态性与虚函数	128
实验 6	函数模板与类模板	137
实验 7	输入/输出的格式控制	141
实验 8	文件的输入与输出	146
实验 9	综合练习	150

第一篇

《C++面向对象程序设计 (第三版)》习题和参考答案

【1.1】什么是面向对象程序设计？

【解】

面向对象程序设计是一种新型的程序设计范型。这种范型的主要特征是：

程序 = 对象 + 消息

面向对象程序的基本元素是对象，面向对象程序的主要结构特点：第一，程序一般由类的定义和类的使用两部分组成，在主程序中定义各对象并规定它们之间传递消息的规律。第二，程序中的一切操作都是通过向对象发送消息来实现的，对象接收到消息后，启动有关方法完成相应的操作。

面向对象程序设计方法模拟人类习惯的解题方法，代表了计算机程序设计新颖的思维方式。这种方法的提出是对软件开发方法的一场革命，是目前解决软件开发面临困难的最有希望、最有前途的方法之一。

【1.2】在面向对象程序设计中，什么是对象？什么是类？对象与类的关系是什么？

【解】

在面向对象程序设计中，对象是描述其属性的数据以及对这些数据施加的一组操作封装在一起构成的统一体。对象可以认为是：数据 + 操作。对象所能完成的操作表示它的动态行为，通常也把操作称为方法。

在面向对象程序设计中，“类”就是具有相同的数据和相同的操作的一组对象的集合，也就是说，类是对具有相同数据结构和相同操作的一类对象的描述。

类和对象之间的关系是抽象和具体的关系。类是多个对象进行综合抽象的结果，一个对象是类的一个实例。

【1.3】现实世界中的对象有哪些特征？

【解】

现实世界中的对象，具有以下特性：

- (1) 每一个对象必须有一个名字以区别于其他对象。
- (2) 用属性来描述它的某些特征。
- (3) 有一组操作，每个操作决定对象的一种行为。
- (4) 对象的操作可以分为两类：一类是自身所承受的操作，一类是施加于其他对象的操作。

【1.4】什么是消息？消息具有什么性质？

【解】

在面向对象程序设计中，一个对象向另一个对象发出的请求称为“消息”。当对象接收到发向它的消息时，就调用有关的方法，执行相应的操作。消息是一个对象要求另一个对象执行某个操作的规格的说明，通过消息传递才能完成对象之间的相互请求或相互协作。

消息具有以下三个性质：

(1) 同一个对象可以接收不同形式的多个消息，做出不同的响应。

(2) 相同形式的消息可以传递给不同的对象，所做出的响应可以是不同的。

(3) 对消息的响应并不是必需的，对象可以响应消息，也可以不响应。

【1.5】什么是方法？在 C++ 中它是通过什么来实现的？

【解】

在面向对象程序设计中的消息传递实际是对现实世界中的信息传递的直接模拟。调用对象中的函数就是向该对象传送一个消息，要求该对象实现某一行（功能、操作）。对象所能实现的行为（操作），在面向对象程序设计中称为方法，它们是通过调用相应的函数来实现的，在 C++ 中方法是通过成员函数来实现的。

【1.6】什么是抽象和封装？

【解】

抽象是人类认识问题的最基本的手段之一。抽象是将有关事物的共性归纳、集中的过程。抽象是通过特定的实例（对象）抽取共同性质后形成概念的过程。面向对象程序设计中的抽象包括两个方面：数据抽象和代码抽象（或称行为抽象）。前者描述某类对象的属性或状态，也就是此类对象区别于彼类对象的特征物理量；后者描述某类对象的共同行为特征或具有的共同功能。

在现实世界中，所谓封装就是把某个事物包围起来，使外界不知道该事物的具体内容。在面向对象程序设计中，封装是指把数据和实现操作的代码集中起来放在对象内部，并尽可能隐蔽对象的内部细节。

封装是面向对象程序设计方法的一个重要特性，所谓封装具有两方面的含义：一是将有关的数据和操作代码封装在一个对象中，各个对象相对独立、互不干扰。二是将对象中某些数据与操作代码对外隐蔽，即隐蔽其内部细节，只留下少量接口，以便与外界联系，接收外界的消息。这种对外界隐蔽的做法称为信息隐蔽。信息隐蔽有利于数据安全，防止无关人员访问和修改数据。

【1.7】什么是继承？请举例说明。

【解】

继承所表达的是对象类之间的相关关系，这种关系使得某类对象可以继承另外一类对象的特征和能力。现实生活中，继承是很普遍和容易理解的。例如，我们继承了父母的一些特征，如种族、血型、眼睛的颜色等，父母是我们所具有的属性的基础。

图 1-1 所示是一个继承的典型例子：汽车继承的层次。

以面向对象程序设计的观点，继承所表达的是对象类之间相关的关系。这种关系使得某一类可以继承另外一个类的特征和能力。

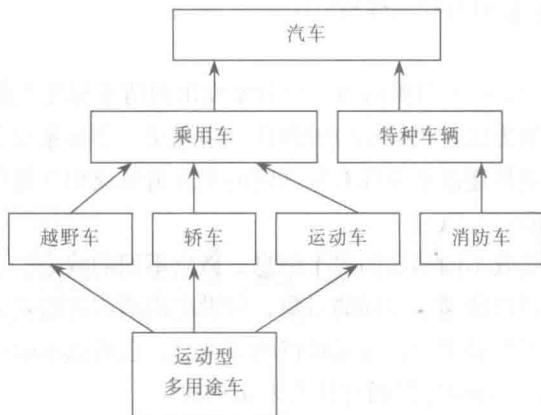


图 1-1 汽车继承的层次图

【1.8】若类之间具有继承关系，则它们之间具有什么特征？

【解】

若类之间具有继承关系，则它们之间具有下列几个特性：

- (1) 类间具有共享特征（包括数据和操作代码的共享）。
- (2) 类间具有差别或新增部分（包括非共享的数据和操作代码）。
- (3) 类间具有层次结构。

假设有两个类 A 和 B，若类 B 继承类 A，则类 B 包含了类 A 的特征（包括数据和操作），同时也可以加入自己所特有的新特性。这时，称被继承类 A 为基类或父类；称继承类 B 为 A 的派生类或子类。同时，我们还可以说，类 B 是从类 A 中派生出来的。

【1.9】什么是单继承、多继承？请举例说明。

【解】

从继承源上分，继承分为单继承和多继承。单继承是指每个派生类只直接继承了一个基类的特征。图 1-2 表示了一种单继承关系。它表示 Windows 操作系统的窗口之间的继承关系。

多继承是指多个基类派生出一个派生类的继承关系。多继承的派生类直接继承了不止一个基类的特征。例如，小孩喜欢的玩具车即继承了车的一些特性，还继承了玩具的一些特征，如图 1-3 所示。

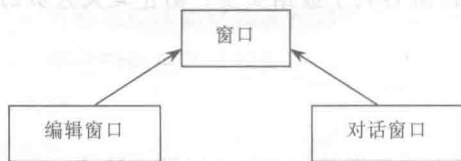


图 1-2 单继承关系

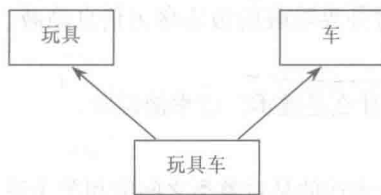


图 1-3 多继承关系

【1.10】什么是多态性？请举例说明。

【解】

多态性也是面向对象程序的重要特征。它是指不同的对象收到相同的消息时产生不同的行为方式。例如，同样双击 Windows 系统桌面上的图标时，有的是打开多媒体播放器，有的是打开资

源管理器。

利用多态性,用户只需发送一般形式的消息,而将所有的实现留给接收消息的对象。对象根据所收到的消息而做出相应的动作。

【1.11】传统程序设计方法的局限性主要有哪些?

【解】

传统程序设计方法的局限性主要有三个方面:

(1) 传统程序设计开发软件的生产效率低下。

传统程序设计的生产方式仍是采用较原始的方式进行,程序设计基本上还是从语句一级开始。软件的生产中缺乏大粒度、可重用的构件,软件的重用问题没有得到很好的解决,从而导致软件生产的工程化和自动化屡屡受阻。传统程序设计的特点是数据与其操作分离,而且对同一数据的操作往往分散在程序的不同的地方。这样,如果一个或多个数据的结构发生了变化,那么这种变化将波及程序的很多部分甚至遍及整个程序,致使许多函数和过程必须重写,严重时会导致整个软件结构的崩溃。传统程序设计是面向过程的,其数据和操作相分离的结构,使得维护数据和处理数据的操作过程要花费大量的精力和时间,严重地影响了软件的生产效率。

总之,要提高软件生产的效率,就必须很好地解决软件的重用性、复杂性和可维护性问题。但是传统的程序设计是难以解决这些问题的。

(2) 传统程序设计难以应付日益庞大的信息量和多样的信息类型。

当代计算机的应用领域已从数值计算扩展到了人类社会的各个方面,所处理的数据已从简单数字和字符发展为具有多种格式的多媒体数据(如文本、图形、图像、影像、声音等),描述的问题从单纯的计算问题到仿真复杂的自然现象和社会现象。随着计算机处理的信息量与信息类型的迅速增加,程序的规模日益庞大,复杂度不断增加,传统程序设计方法是无法应付的。

(3) 传统的程序设计难以适应各种新环境。

当前,并行处理、分布式、网络和多机系统等已经或将是程序运行的主流方式和主流环境。这些环境的一个共同的特点是都具有一些有独立处理能力的结点,结点之间有通信机制,即以消息传递进行联络。显然传统的程序设计技术很难适应这些新环境。

【1.12】面向对象程序设计的优点主要有哪些?

【解】

面向对象程序设计的优点主要包括以下几个方面:

(1) 可提高程序的重用性。

重用是提高软件开发效率的最主要的方法,面向对象程序设计能比较好地解决软件重用的问题。对象所固有的封装性和信息隐藏等机理,使得对象内部的实现与外界隔离,具有较强的独立性,它可以作为一个大粒度的程序构件,供同类程序直接使用。

(2) 可控制程序的复杂性。

面向对象程序设计采用了数据抽象和信息隐藏技术,把数据及对数据的操作放在一个个类中,作为相互依存、不可分割的整体来处理。这样,在程序中任何要访问这些数据的地方都只需简单地通过传递信息和调用方法来进行,这就有效地控制了程序的复杂性。

(3) 可改善程序的可维护性。

在面向对象程序设计中,对对象的操作只能通过消息传递来实现,所以只要消息模式即对应

的方法界面不变,方法体的任何修改不会导致发送消息的程序修改,这显然对程序的维护带来了方便。另外,类的封装和信息隐藏机制使得外界对其中的数据 and 程序代码的非法操作成为不可能,这也就大大地减少了程序的错误率。

(4) 能够更好地支持大型程序设计。

面向对象技术在数据抽象和抽象数据类型之上又引入了动态链接和继承性等机制,进一步发展了基于数据抽象的模块化设计,使其更好地支持大型程序设计。

(5) 增强了计算机处理信息的范围。

面向对象程序设计方法模拟人类习惯的解题方法,代表了计算机程序设计的新颖的思维方法。用类来直接描述现实世界中的类型,可使计算机系统的描述和处理对象从数据扩展到现实世界和思维世界的各种事物,这实际上大大扩展了计算机系统处理的信息量和信息类型。

(6) 很好地适应新的硬件环境。

面向对象程序设计中的对象、消息传递等思想和机制,与分布式、并行处理、多机系统及网络等硬件环境也恰好相吻合。面向对象的思想也影响到计算机硬件的体系结构,现在已在研究直接支持对象概念的面向对象计算机。这样的计算机将会更适合于面向对象程序设计,更充分地发挥面向对象技术的威力。

第 2 章 | C++ 基础

【2.1】简述 C++ 的主要特点。

【解】

C++ 最主要的特点有以下几点：

(1) C++ 保持与 C 兼容，这就使许多 C 代码不经修改就可以为 C++ 所用，用 C 语言编写的众多的库函数和实用软件可以用于 C++ 中。

(2) C++ 保持了 C 语言的简洁、高效和接近汇编语言等特点，并对 C 语言的功能做了不少扩充。用 C++ 编写的程序比 C 语言编写的程序更安全，可读性更好，代码结构更为合理。

(3) 用 C++ 编写的程序质量高，从开发时间、费用到形成的软件的可重用性、可扩充性、可维护性和可靠性等方面有了很大的提高，使得大中型的程序开发变得更加容易。

(4) 增加了面向对象的机制，C++ 几乎支持所有的面向对象程序设计特征，体现了近几十年来在程序设计和软件开发领域出现的新思想和新技术。

C++ 最有意义的方面是支持面向对象的特征，然而，由于 C++ 与 C 语言保持兼容，使得 C++ 不是一个纯正的面向对象的语言，C++ 既可用于面向过程的结构化程序设计，也可用于面向对象的程序设计。

【2.2】下面是一个 C 程序，改写它，使它采用 C++ 风格的 I/O 语句。

```
#include <stdio.h>
int main()
{
    int a,b,d,min;
    printf("Enter two numbers: ");
    scanf("%d%d",&a,&b);
    min=a>b?b:a;
    for(d=2;d<min;d++)
        if(((a%d)==0)&&((b%d)==0)) break;
    if(d==min)
    {
        printf("No common denominators\n");
        return 0;
    }
    printf("The lowest common denominator is %d\n",d);
    return 0;
}
```

【解】