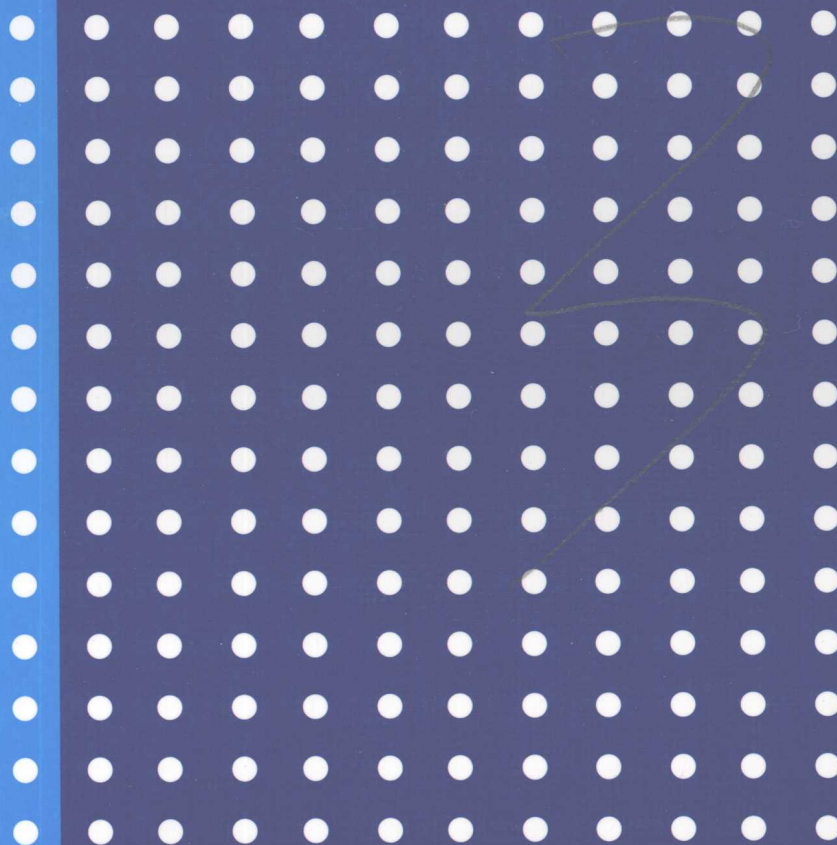


重点大学计算机专业系列教材

C++语言程序设计教程

孟宪福 王 旭 编著



清华大学出版社



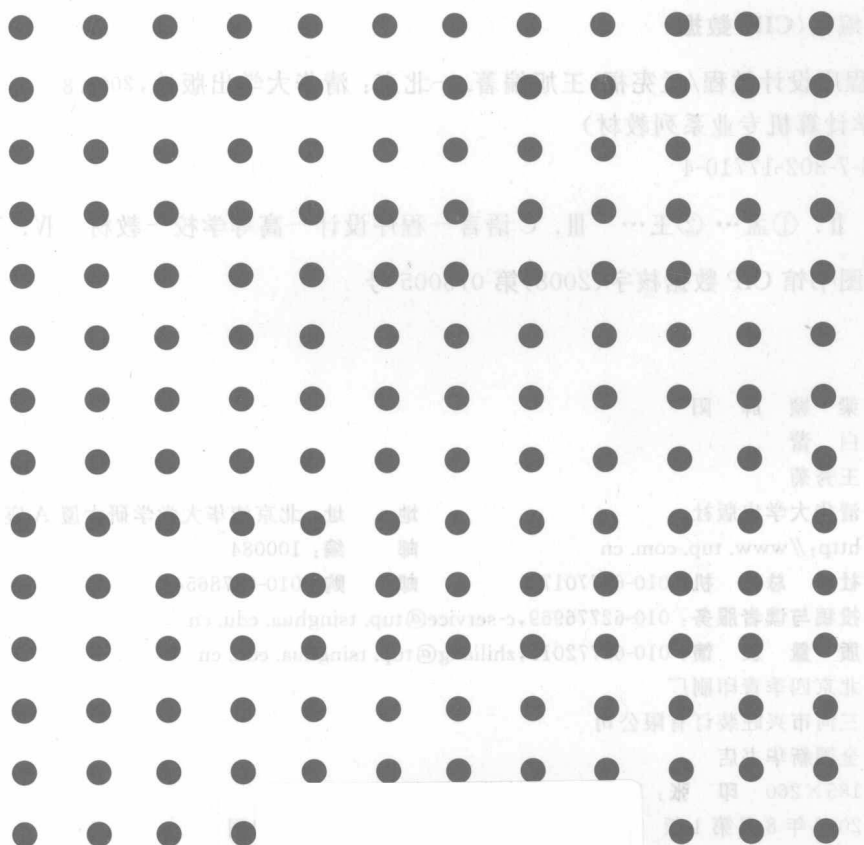
TP312/2958

2008

重点大学计算机专业系列教材

C++语言程序设计教程

孟宪福 王旭 编著



清华大学出版社

北京

内 容 简 介

C++语言是在C语言的基础上发展起来的面向对象程序设计语言,使用它不仅可以编写应用软件,而且特别适合于编写系统软件。本书共由10章组成,按照循序渐进的原则,逐步而系统地介绍C++语言的基本概念和语法规则,特别是花大量的篇幅来详细讲解面向对象程序设计的基本概念,包括类、继承、多态和重载等,并利用单独的一章来专门介绍类的设计,使读者在学完本书后,能尽快应用C++语言来解决实际问题。本书是作者根据多年的C++语言教学经验和实践体会编写而成的,在内容编排上尽量体现易学的特点,在文字叙述上力求条理清晰、简洁,便于读者阅读。

本书可以作为大专院校计算机专业或非计算机专业的教材及教学参考书,也可作为自学用书。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

C++语言程序设计教程/孟宪福,王旭编著. —北京:清华大学出版社,2008.8

(重点大学计算机专业系列教材)

ISBN 978-7-302-17710-4

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

中国版本图书馆CIP数据核字(2008)第075005号

责任编辑:梁颖 薛阳

责任校对:白蕾

责任印制:王秀菊

出版发行:清华大学出版社

地 址:北京清华大学学研大厦A座

<http://www.tup.com.cn>

邮 编:100084

社 总 机:010-62770175

邮 购:010-62786544

投稿与读者服务:010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

印 刷 者:北京四季青印刷厂

装 订 者:三河市兴旺装订有限公司

经 销:全国新华书店

开 本:185×260 印 张:17.25 字 数:413千字

版 次:2008年8月第1版 印 次:2008年8月第1次印刷

印 数:1~4000

定 价:26.00元

本书如存在文字不清、漏印、缺页、倒页、脱页等印装质量问题,请与清华大学出版社出版部联系调换。联系电话:(010)62770177 转 3103 产品编号:029738-01

出版说明

随着国家信息化步伐的加快和高等教育规模的扩大,社会对计算机专业人才的需求不仅体现在数量的增加上,而且体现在质量要求的提高上,培养具有研究和实践能力的高层次的计算机专业人才已成为许多重点大学计算机专业教育的主要目标。目前,我国共有 16 个国家重点学科、20 个博士点一级学科、28 个博士点二级学科集中在教育部部属重点大学,这些高校在计算机教学和科研方面具有一定优势,并且大多以国际著名大学计算机教育为参照系,具有系统完善的教学课程体系、教学实验体系、教学质量保证体系和人才培养评估体系等综合体系,形成了培养一流人才的教学和科研环境。

重点大学计算机学科的教学与科研氛围是培养一流计算机人才的基础,其中专业教材的使用和建设则是这种氛围的重要组成部分,一批具有学科方向特色优势的计算机专业教材作为各重点大学的重点建设项目成果得到肯定。为了展示和发扬各重点大学在计算机专业教育上的优势,特别是专业教材建设上的优势,同时配合各重点大学的计算机学科建设和专业课程教学需要,在教育部相关教学指导委员会专家的建议和各重点大学的大力支持下,清华大学出版社规划并出版本系列教材。本系列教材的建设旨在“汇聚学科精英、引领学科建设、培育专业英才”,同时以教材示范各重点大学的优秀教学理念、教学方法、教学手段和教学内容等。

本系列教材在规划过程中体现了如下一些基本组织原则和特点。

1. 面向学科发展的前沿,适应当前社会对计算机专业高级人才的培养需求。教材内容以基本理论为基础,反映基本理论和原理的综合应用,重视实践和应用环节。

2. 反映教学需要,促进教学发展。教材要能适应多样化的教学需要,正确把握教学内容和课程体系的改革方向。在选择教材内容和编写体系时注意体现素质教育、创新能力与实践能力的培养,为学生知识、能力、素质协调发展创造条件。

3. 实施精品战略,突出重点,保证质量。规划教材建设的重点依然是专业基础课和专业主干课;特别注意选择并安排了一部分原来基础比较好的优秀教材或讲义修订再版,逐步形成精品教材;提倡并鼓励编写体现重点大学

计算机专业教学内容和课程体系改革成果的教材。

4. 主张一纲多本,合理配套。专业基础课和专业主干课教材要配套,同一门课程可以有多本具有不同内容特点的教材。处理好教材统一性与多样化的关系;基本教材与辅助教材以及教学参考书的关系;文字教材与软件教材的关系,实现教材系列资源配套。

5. 依靠专家,择优落实。在制订教材规划时要依靠各课程专家在调查研究本课程教材建设现状的基础上提出规划选题。在落实主编人选时,要引入竞争机制,通过申报、评审确定主编。书稿完成后要认真实行审稿程序,确保出书质量。

繁荣教材出版事业,提高教材质量的关键是教师。建立一支高水平的以老带新的教材编写队伍才能保证教材的编写质量,希望有志于教材建设的教师能够加入到我们的编写队伍中来。

教材编委会

前言

随着计算机技术的不断发展以及软件程序的高度复杂化,面向对象程序设计的重要性也越来越突现出来,而 C++ 语言则是面向对象程序设计的最重要的代表性语言之一。

C++ 语言是在被广泛应用的 C 语言的基础上发展起来的。C++ 语言在 C 语言已有的功能的基础上,强化了 C 语言的基本功能,特别是增加了对类的处理能力,即: $C++ = C + \text{基本功能的扩充} + \text{类的功能}$ 。从这一公式不难看出, C++ 语言几乎完全继承了 C 语言的所有功能。从表面上看来,由于 C++ 语言继承了 C 语言的所有功能,因此,只要学会了 C 语言,就应该很容易学会 C++ 语言,其实不然。就类本身来讲,就包含了很多复杂的概念,而对于这些概念的正确理解则是学好面向对象程序设计语言的关键;同时, C 语言是面向函数的程序设计语言,而 C++ 语言则是面向对象的程序设计语言。这样,在程序设计方法上就有很大的差别。

本书是在作者多年讲授 C++ 语言教案的基础上,广泛听取读者和同行的建议,并参考最新资料经系统整理而成的。尽管大部分读者可能都学过 C 语言,但考虑到 C++ 语言作为一门独立的课程应具有其系统性,因此本书系统地介绍了 C++ 语言的各种语法成分和程序设计特点。在内容编排上,按照循序渐进的原则,逐步介绍 C++ 语言中的基本概念和理论,在章节内容的安排上,尽可能考虑读者的接受能力,使整个学习过程按照从简单到复杂的顺序进行。为了使读者能够尽快利用 C++ 语言来解决实际问题,在本书的每一章中都给出了大量的例题,这些例题对于理解 C++ 语言的语法现象、完整掌握 C++ 语言的特点是非常有益的。同时,考虑到面向对象程序设计语言的特点,在本书的第 8 章中专门利用一章的篇幅来详细介绍类的设计,所给出的几个例子都是具有代表性的并具有实用价值的。通过对这些实例的学习,能够使读者进一步掌握面向对象程序设计的要点,并能达到举一反三的目的。

本书共由 10 章组成,按照循序渐进的原则,逐步地介绍 C++ 语言的基本概念和语法规则,特别是花大量的篇幅来详细讲解面向对象程序设计的基本概念,包括类、继承、多态和重载等。书中的所有例题都在 Visual C++ 环境下测试完成。每章的最后也都附有一定量的习题,这些习题对于读者巩固已学

的内容是大有益处的。

作者认为,要学好 C++ 语言,除了掌握 C++ 语言的基本理论之外,还必须要加强实践环节,读者可以边学习边上机。刚开始时可以调试本书中的例题,待学习一段时间之后,就可以调试自己编写的程序了。只有这样,才能加快学习进度,提高学习效率。

由于作者水平有限,经验不足,书中难免存在不妥之处,敬请有关老师、计算机工作者和广大读者批评指正。

作 者

2008 年 4 月

读者意见反馈

亲爱的读者：

感谢您一直以来对清华版计算机教材的支持和爱护。为了今后为您提供更优秀的教材，请您抽出宝贵的时间来填写下面的意见反馈表，以便我们更好地对本教材做进一步改进。同时如果您在使用本教材的过程中遇到了什么问题，或者有什么好的建议，也请您来信告诉我们。

地址：北京市海淀区双清路学研大厦 A 座 602 室 计算机与信息分社营销室 收
邮编：100084 电子邮箱：jsjc@tup.tsinghua.edu.cn
电话：010-62770175-4608/4409 邮购电话：010-62786544

教材名称：C++ 语言程序设计教程

ISBN 978-7-302-17710-4

个人资料

姓名：_____ 年龄：_____ 所在院校/专业：_____

文化程度：_____ 通信地址：_____

联系电话：_____ 电子信箱：_____

您使用本书是作为：☐指定教材 ☐选用教材 ☐辅导教材 ☐自学教材

您对本书封面设计的满意度：

☐很满意 ☐满意 ☐一般 ☐不满意 改进建议_____

您对本书印刷质量的满意度：

☐很满意 ☐满意 ☐一般 ☐不满意 改进建议_____

您对本书的总体满意度：

从语言质量角度看 ☐很满意 ☐满意 ☐一般 ☐不满意

从科技含量角度看 ☐很满意 ☐满意 ☐一般 ☐不满意

本书最令您满意的是：

☐指导明确 ☐内容充实 ☐讲解详尽 ☐实例丰富

您认为本书在哪些地方应进行修改？（可附页）

您希望本书在哪些方面进行改进？（可附页）

电子教案支持

敬爱的教师：

为了配合本课程的教学需要，本教材配有配套的电子教案（素材），有需求的教师可以与我们联系，我们将向使用本教材进行教学的教师免费赠送电子教案（素材），希望有助于教学活动的开展。相关信息请拨打电话 010-62776969 或发送电子邮件至jsjc@tup.tsinghua.edu.cn 咨询，也可以到清华大学出版社主页（<http://www.tup.com.cn> 或 <http://www.tup.tsinghua.edu.cn>）上查询。

重点大学计算机专业系列教材书目

本系列教材	作 者	书 号
C 语言程序设计	李春葆等	9787302144779
C 语言程序设计辅导	李春葆等	9787302144465
C 语言高级程序设计	张俐等	9787302132875
Java 与 UML 面向对象程序设计教程	刘晓冬等	9787302156451
Java 语言程序设计	郎波等	9787302106357
Linux 实践及应用	罗文村等	9787302130130
SoC 技术原理与应用	郭兵等	9787302125525
UML 与软件建模	徐宝文等	9787302118466
Web 开发技术及其应用	王成良	9787302162292
Windows 汇编语言程序设计实验指导	谭毓安、张雪兰等	9787302171942
电子商务导论	黄晓涛等	9787302111122
多媒体技术与网页设计	陈新龙等	9787302134633
多媒体计算机原理与应用	鲁宏伟等	9787302119708
汇编语言程序设计——从 DOS 到 Windows	张雪兰等	9787302124368
计算机病毒与反病毒技术	张仁斌等	9787302127277
计算机网络工程实践教程——基于 Cisco 路由器和交换机	陆魁军	9787302141938
计算机网络工程实践教程——基于华为路由器和交换机	陆魁军	9787302122159
计算机网络基础实践教程	陆魁军	9787302116653
计算机语言与程序设计	谌卫军	9787302154341
计算机组成原理	张功萱等	9787302113607
嵌入式系统开发原理与实践	陈文智等	9787302116004
实用数值计算方法	甄西丰	9787302118534
数据结构(C++ 描述)	金远平等	9787302107989
数据结构教程(第二版)	李春葆等	9787302142294
数据库系统基础教程	叶小平、汤庸等	9787302142638
数据库系统实验指导教程	汤娜等	9787302125600
数据通信基础	国林等	9787302130659
数字图像处理与分析	刘直芳等	9787302134824
数据挖掘原理与算法(第二版)	毛国君等	9787302158769
网络测试与故障诊断实验教程	曹庆华等	9787302134008
网站设计与建设	刘运臣等	9787302168539
网络协议与网络安全	凌力	9787302157564
微机系统和接口应用技术	朱世鸿	9787302124276
微型计算机技术	田艾平等	9787302105480
微型计算机原理与接口实践	宁飞等	9787302127284
信息安全技术基础和安全策略	薛质、李建华等	9787302140870
信息安全数学基础	陈恭亮	9787302084471
信息与网络安全实验教程	王常吉等	9787302156062
应用系统开发导论	韩伟力、臧斌宇等	9787302163695
中文文本信息处理的原理与应用	苗夺谦等	9787302154983

CONTENTS

目录

第 1 章 绪言	1
1.1 面向对象程序设计的特点	1
1.2 C++ 语言程序的开发过程	2
1.3 C++ 语言程序的结构	4
习题	6
第 2 章 数据类型、运算符和基本语句	8
2.1 基本概念	8
2.1.1 标识符	8
2.1.2 常量	9
2.1.3 变量	9
2.1.4 关键字	9
2.2 基本数据类型	9
2.2.1 整型变量及其常量	9
2.2.2 浮点型变量及其常量	10
2.2.3 字符型变量及其常量	10
2.2.4 void 型数据	11
2.2.5 bool 型变量及其常量	11
2.3 long、short、signed、unsigned 关键字	12
2.3.1 long 和 short 关键字	12
2.3.2 signed 和 unsigned 关键字	12
2.4 枚举	13
2.5 const 关键字	14
2.6 volatile 关键字	15
2.7 typedef 关键字	16
2.8 不同类型数据之间的转换	16
2.8.1 自动类型转换	16
2.8.2 强制类型转换	17

2.9 运算符	18
2.9.1 算术运算符	18
2.9.2 增 1、减 1 运算符	18
2.9.3 关系运算符	19
2.9.4 逻辑运算符	20
2.9.5 位运算符	20
2.9.6 赋值运算符	24
2.9.7 条件运算符	24
2.9.8 逗号运算符	25
2.9.9 sizeof 运算符	26
2.9.10 指针运算符	26
2.9.11 成员访问运算符	26
2.10 基本语句	26
2.10.1 语句、复合语句和空语句	27
2.10.2 if 语句	27
2.10.3 switch 语句	28
2.10.4 while 语句	31
2.10.5 for 语句	32
2.10.6 do-while 语句	33
2.10.7 break 语句	34
2.10.8 continue 语句	35
2.10.9 goto 语句	36
2.10.10 return 语句	36
习题	37
第 3 章 数据的输入和输出	40
3.1 标准输入和输出	40
3.1.1 基于运算符<<和>>的输入输出	41
3.1.2 字符的输入 get()和输出 put()	45
3.1.3 字符串的输入 get()和 getline()	46
3.2 文件	47
3.2.1 文件的打开和关闭	47
3.2.2 文件的输入和输出	49
3.2.3 错误处理	49
习题	51
第 4 章 数组、指针和引用	54
4.1 数组	54
4.1.1 一维数组	54

4.1.2	二维数组	57
4.1.3	字符数组和字符串	61
4.2	指针	63
4.2.1	指针的基本概念	63
4.2.2	void 型指针	65
4.2.3	二级指针	65
4.2.4	指针和数组	67
4.3	引用	75
4.4	内存的申请与释放	77
	习题	78
第 5 章	函数	80
5.1	函数的定义和调用	80
5.2	函数的返回值及其类型	81
5.2.1	函数返回值	81
5.2.2	函数返回指针	82
5.2.3	函数返回引用	85
5.3	函数原型	86
5.4	函数的参数及其传递方式	88
5.4.1	将值传递给函数	88
5.4.2	将常量传递给函数	88
5.4.3	将指针传递给函数	89
5.4.4	将引用传递给函数	90
5.4.5	将数组传递给函数	90
5.5	函数的递归调用	92
5.6	变量的作用域和存储类	94
5.6.1	变量的作用域	95
5.6.2	变量的存储类	96
5.7	无参函数和默认参数	99
5.8	函数的重载	102
5.9	inline 函数	103
5.10	外部函数和静态函数	105
	习题	106
第 6 章	类	109
6.1	类的定义	109
6.2	公共、私有和保护	110
6.3	数据成员和成员函数	111
6.3.1	成员函数的使用	111

6.3.2	成员函数的内部定义和外部定义	111
6.3.3	数据成员的保护	113
6.4	构造函数和析构函数	114
6.4.1	构造函数	114
6.4.2	析构函数	121
6.5	复制构造函数	122
6.5.1	复制构造函数的说明和定义	122
6.5.2	默认复制构造函数	124
6.5.3	复制构造函数的进一步说明	128
6.6	变换构造函数和变换函数	131
6.6.1	变换构造函数	131
6.6.2	变换函数	132
6.7	静态数据成员和静态成员函数	134
6.8	this 指针	137
6.9	友元	139
6.9.1	友元函数	139
6.9.2	友元类	141
6.10	运算符的重载	142
6.10.1	operator 函数的功能	142
6.10.2	operator 函数的重载	145
6.10.3	类的友元是 operator 函数	147
6.10.4	几个典型的例子	149
6.11	const 对象	153
6.12	类的嵌套定义	156
6.13	类的数据成员是类对象或常量	160
6.13.1	类的数据成员是类对象	160
6.13.2	类的数据成员是常量	163
6.14	对象数组	166
6.15	指向类的成员的指针	169
6.15.1	指向类的非静态成员的指针	170
6.15.2	指向类的静态成员的指针	171
6.16	结构	174
6.17	联合	176
6.18	位段	177
	习题	178
第 7 章	继承	180
7.1	基类和派生类	180
7.2	虚函数和多态	184
7.2.1	静态结合和动态结合	185

7.2.2	虚函数.....	187
7.3	纯虚函数和抽象类	190
7.3.1	纯虚函数.....	190
7.3.2	抽象类.....	193
7.4	虚析构造函数	193
7.5	继承的种类	195
7.5.1	多重继承.....	195
7.5.2	直接继承和间接继承.....	199
7.6	多重基类和虚拟基类	200
7.7	继承方式下的构造与析构	202
	习题.....	204
第 8 章	类的设计	206
8.1	计数器类的设计	206
8.2	字符串类的设计	213
8.3	链表类的设计	216
8.4	堆栈类的设计	220
8.5	数组类的设计	222
8.6	用于实现多态性的例子	224
	习题.....	226
第 9 章	模板和异常处理	228
9.1	模板	228
9.1.1	函数模板.....	229
9.1.2	类模板.....	230
9.1.3	STL 简介	233
9.2	异常处理	237
9.2.1	try 关键字的使用	238
9.2.2	throw 关键字的使用	238
9.2.3	catch 关键字的使用	239
9.2.4	异常处理对象.....	241
9.2.5	异常处理中的构造与析构.....	244
	习题.....	246
第 10 章	编译预处理	247
10.1	宏定义	247
10.2	文件包括	250
10.3	条件编译	252
10.4	其他	255
	习题	257

绪 言

第 1 章

随着计算机技术的不断发展以及软件设计规模的不断扩大,计算机软件的开发面临着两大难题:一是如何越过程序设计的复杂性障碍问题;另一个是如何利用软件来自然地表示客观世界,也就是对象模型问题。面向对象的程序设计技术很好地解决了上述问题,而 C++ 语言正是面向对象程序设计技术的具体实现和典型代表。

C++ 语言是在 C 语言的基础上发展起来的,它既融合了面向对象的程序设计技术,又保留了 C 语言的特征。C++ 语言在提供了面向对象的设计能力的同时,又保持了与 C 语言的高度兼容性,使 C 程序设计人员能够比较容易地转向 C++ 语言。

1.1 面向对象程序设计的特点

世界上的任何事物都可以被看作为对象,对象是对现实世界的抽象。一本书可以是一个对象,而一个图书馆也可以是一个对象。从软件设计的角度来讲,一个对象就是一个高度抽象的模块,该模块中既包含了相应的数据结构,又提供了对数据结构进行操作的方法。正是由于这种高度抽象的结果,使得面向对象程序设计具有许多面向过程程序设计所无法比拟的特点。

归纳起来,面向对象程序设计有如下一些主要特点。

1. 模块化

模块化又被称为抽象。采用面向对象技术设计出来的程序都是由一个对象组成的,在每一个对象中,既定义了相应的数据结构,同时又定义了操作这些数据结构的方法,这样,每一个对象都是一个完整的功能模块,都是对事物的高度抽象。因此,由面向对象程序设计语言所设计出来的程序,其模块化程度高,易于扩充和维护。

2. 数据隐藏

数据隐藏又被称为封装。在面向过程的程序设计中,一般注重的是程序的代码,而把数据放在次要位置上。实际上,只有将数据和操作这些数据的方法结合起来,才能完整地描述一个程序。数据隐藏的目的是将对象的设计和对象的使用分开,使用者不必了解对象实现的细节,只要利用设计者所提供的接口来访问该对象即可。同时,由于为对象中的成员加上了访问权限控制信息,使得外部对对象中成员的访问是有限制的:对于那些私有成员来讲,外部就无法进行访问,从而起到了数据隐藏的作用。

面向对象程序设计通过定义类把数据和方法集成在一个对象中,外部程序只能通过类所规定的接口和类进行通信。

3. 继承

在面向对象程序设计中,对象是程序的基本单位,复杂的对象可由简单的对象来组成,而继承操作正是用于完成这一工作的。它允许从已经存在的类中继承相应的成员(数据和方法),只要告诉编译器你的新类是由另一个类继承而来的,它就会把另一个类中除私有成员之外的所有成员都赋给你的新类,就如同你重新定义的一样。显然,继承操作能够减轻程序设计的工作量,提高程序的可靠性。

4. 多态性

所谓多态,是指一个名字可具有多种不同的语义,或者说多个函数具有相同的名字但具有不同的作用。在继承操作当中,如果父类和子类中的某个函数具有相同的名字,且被定义为虚函数,则利用指向父类的指针,根据所赋给的对象不同(父类对象或子类对象),就可以调用不同的函数。

面向对象程序设计中的多态性,给程序设计带来了很大的灵活性,使我们既能够重用已有的类,又能够满足新类的需要。

5. 重载

在面向过程的程序设计语言中,每个函数必须有一个唯一的名字,也就是说函数不能重名。而在像 C++ 这样的面向对象程序设计语言中,多个函数可以共用一个名字,只要它们的参数个数不同或参数类型不同即可,这就是函数的重载。在 C++ 语言中,不但函数可以重载,运算符也可以重载,可以根据需要为已有的运算符赋予不同的意义。显然,增加了重载功能以后,将提高程序的可理解性,使程序功能实现起来更自然、更流畅。

1.2 C++ 语言程序的开发过程

开发一个 C++ 语言程序的基本过程如图 1.1 所示。

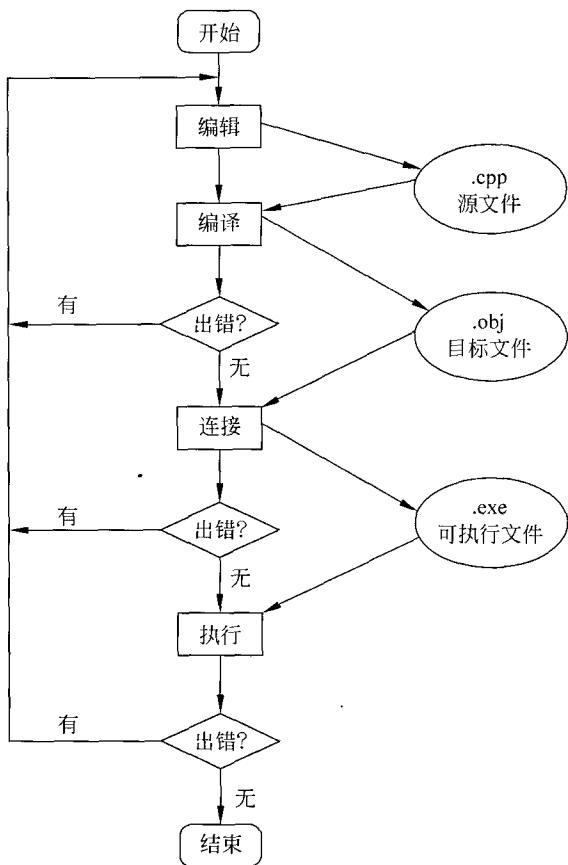


图 1.1 C++ 语言程序的开发过程

1. 编辑

选择适当的编辑程序,将 C++ 语言源程序通过键盘输入到计算机中,并以文件的形式存入到磁盘中。经过编辑后得到的源程序文件是以 .cpp 为其文件扩展名的。

2. 编译

通过编辑程序将源程序输入到计算机后,需要经过 C++ 语言编译器将其编译成目标程序。在对源程序的编译过程中,可能会发现程序中的一些词法或语法错误,这时就需要重新利用编辑程序来修改源程序,然后再重新编译。经过编译后得到的目标文件是以 .obj 为其文件扩展名的。

3. 连接

经过编译后生成的目标文件是不能直接执行的,它需要经过连接之后才能生成可执行的代码。连接后所得到的可执行文件是以 .exe 为其文件扩展名的。

4. 执行

经过编译、连接之后,源程序文件就可以生成可执行的文件,这时就可以执行了。在操