

国家精品课程主讲教材

高等学校计算机程序设计课程系列教材

C++程序设计教程(第2版)

罗建军 朱丹军 顾刚 刘路放 编著
冯博琴 审



高等教育出版社
Higher Education Press

国家精品课程主讲教材
高等学校计算机程序设计课程系列教材

C++ 程序设计教程

(第2版)

罗建军 朱丹军 顾刚 刘路放 编著
冯博琴 审

高等教育出版社

内容提要

本书以“精讲多练”的方式介绍面向对象 C++ 程序设计,主要包括 C++ 编程基础(基本数据类型和表达式、控制结构、函数、指针等)和面向对象的程序设计(类和对象、继承与派生、多态性、模板和异常等)。在编写格式上,按以下 8 个主题精心组织内容:本章目标、授课内容、自学内容、程序设计举例、编程提示、实例编程、小结和习题,从而做到条理清楚,逻辑分明。

本书除了突出编程方法的讲解力度以外,还提供了大量有代表性的例题、实例和习题,在阐述面向对象理论的同时,突出了实践环节,切实使读者的独立编程能力得到提高。

本书可作为高等学校理工类各专业程序设计的教材或参考书,也可供应用程序开发人员学习参考。

图书在版编目(CIP)数据

C++ 程序设计教程/罗建军等编著. —2版. —北京:高等教育出版社,2007.8

ISBN 978-7-04-022080-3

I. C… II. 罗… III. C 语言-程序设计-教材 IV. TP312

中国版本图书馆 CIP 数据核字(2007)第 109259 号

策划编辑 张 龙 责任编辑 彭立辉 封面设计 于文燕 责任绘图 尹 莉
版式设计 陆瑞红 责任校对 王效珍 责任印制 朱学忠

出版发行 高等教育出版社

社 址 北京市西城区德外大街 4 号

邮政编码 100011

总 机 010-58581000

经 销 蓝色畅想图书发行有限公司

印 刷 北京鑫海金澳胶印有限公司

购书热线 010-58581118

免费咨询 800-810-0598

网 址 <http://www.hep.edu.cn>

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

网上订购 <http://www.landrace.com>

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

畅想教育 <http://www.widedu.com>

开 本 787×1092 1/16

印 张 25.5

字 数 620 000

版 次 2004 年 8 月第 1 版

2007 年 8 月第 2 版

印 次 2007 年 8 月第 1 次印刷

定 价 31.70 元

本书如有缺页、倒页、脱页等质量问题,请到所购图书销售部门联系调换。

版权所有 侵权必究

物料号 22080-00

第2版前言

自2004年第1版《大学C++程序设计教程》问世以来,经过本校十余位教师、万余名学生的教学与实践,收到了大量的修改建议;同时,出版社和兄弟院校也给出了很多有益的反馈意见。通过三年的积累,我们深感需要对有关内容进行修订,才能真正使本书做到深入浅出、易学易教,从而更好地满足教学一线需要。

在修订本教材时,作者对每章的内容都重新进行了仔细研究论证,并综合考虑了C++知识体系结构、计算机教学规律和学生学习习惯等诸多因素。因此,除了修正原书的错误外,新版在第1版的基础上做了较大的改动。主要的修改和调整如下:

(1) 为了使教师能够更容易地组织和实施教学过程,本次修订调整了部分章节的讲解次序,解决了以往教学实践中所反映出来的一些问题。

➤ 合并原书中的第3章(基本数据类型)和第5章(表达式)的部分内容,改编成为第2章(基本数据类型与表达式),将原来的第2章(控制结构)改编成第3章,从内容上做到循序渐进,更便于学生理解。

➤ 类与对象的基本概念和方法往往是学生开始学习面向对象程序设计的一个难点,新版教材将原来的第9章的内容扩展为两章(第9、10章)。

➤ 对原书的第14章(线性表)重新进行了组织、改写并调整为第15章,删去了原书中学生反映生涩难懂的第15章(查找和排序)和第16章(数值计算)。

(2) 为了保证学生能够在较短的时间内迅速掌握C++的基本知识,具备相应的程序设计能力,本次修订进一步突出了“精讲多练”的特色,除完善知识点的讲解外,还加强了技能点的训练。

➤ 配备了一些在教学实践中效果比较好的例题和习题,特别是对于某些重点章节(如第3章控制结构),更是大幅度地增加了例题和习题的类型和数量。

➤ 突出了编程方法学的讲解力度,增加了结构化设计方法(第8章)和面向对象软件开发方法(第16章)方面的内容。

(3) 为了与C++标准(ISO/IEC 14882)相一致,使用标准C++语言的头文件改写了所有源程序,替换了原来的C语言库函数的头文件形式。

本次修订工作是在冯博琴教授组织领导下,由罗建军(第1~8章、第13章、附录)、朱丹军(第9~12章、第14章、第16章)、顾刚(第15章)等负责完成,全书由罗建军统稿,冯博琴教授主审。

本书的修订得到了西安交通大学“十一五”规划本科生系列教材的经费支持,在此表示感谢。另外,还要感谢作者所在的西安交通大学计算机教学实验中心从事程序设计教学的十多位同仁,多年来他们不仅在使用本书的教学实践中提出问题和建议,使本书能够很快纠正错误和弥补不足,而且还积极参与“计算机程序设计”国家精品课程网

站的建设工作,使本书最终成为配套完善的立体化多媒体教材。感谢兄弟院校的授课教师和使用本教材的广大学生,他们反馈来的大量问题和建议也使本书能够更好地满足各类读者的需求。最后,还要特别感谢西安交通大学 2005 级的张亮、吴丽娜等同学,作为本书修订版清样的第一批学生读者,他们也提出了很多修改意见和建议。

国家精品课程“计算机程序设计”的课程网站为 <http://programming.xjtu.edu.cn>。该网站提供与本教材同步的全交互性网上辅导答疑讨论区、最新教学辅助课件、全程教学录像、上机实验指导及示例程序源码、优秀学生作业、课程大纲、教学计划等资源,所有内容均免费开放,并由专人负责不断更新,可供教师教学和学生自学使用。

“一切为了教学,一切为了读者”是我们的心愿和目标,但由于水平所限,难免存在不足之处,恳请广大教师和同学指正。

编著者

2007年4月于西安交通大学

第1版前言

在面向对象程序设计语言中,C++是最流行的语言之一。C++从C语言演变而来,语法上也更严格、安全。而它提供的面向对象功能,使采用C++编写的程序更容易理解和维护,可以更高程度地进行复用,从而更快、更正确、更经济地开发软件。随着国际标准化组织(ISO)和美国国家标准化协会(ANSI)的ISO/ANSI C++语言标准(ISO/IEC FDIS 14882)的推出,C++迅速成为使用最广泛的编程语言。

本教程是C++入门的教科书,适用于高等学校理工类各专业学生程序设计能力的培养。为了使基础不高的初学者也能很快地掌握程序设计方法,我们在确定教学目标、设计教学模式、编写教程内容等方面进行了一系列革新探索。本教材具有3个特点:

(1) 全书是按照教育部最新的“关于进一步加强高校计算机基础教学的几点意见”文件要求组织编写,并以现代教育理论为指导,适合于“精讲多练”的教学模式。

(2) 作者在多年教学实践中发现初学者经常易犯一些小错误,或者不注意培养好的编程习惯。本书将这些带有普遍性的问题以编程提示的形式列举出来,以便初学者少犯错误,少走弯路,尽快掌握C++语言程序设计方法。

(3) 为了便于初学者进一步理解和巩固C++基本知识,提高解决实际问题的能力,本书增加了基本数据结构和数值计算两方面的内容。

本教材的目标是使学生掌握使用C++设计应用程序的基本技能,了解面向对象和结构化程序设计的方法,能够编写、调试和运行实用、规范、可读性好的C++程序。本书“从零开始”,在内容组织上循序渐进,不要求学生学过程序设计方面的先修课程。

本教程由16章组成,又可分为三部分:

第一部分为第1~8章,介绍C++编程的基本内容,包括控制结构、基本数据类型、表达式、函数、指针和引用。

第二部分为第9~13章,介绍类与对象、继承和多态性等面向对象程序设计的基础理论以及C++的标准库。

第三部分为第14~16章,介绍基本数据结构和数值计算方法。

为了便于教学,每章均按以下主题进行组织:

本章目标 本书的特点是“精讲多练”,因此为教师和学生规定明确的教学和学习目标是非常重要的。

授课内容 是建议教师课堂讲授的内容。一般来说,授课内容是本章所有教学内容的“纲”,起着联系本章所有项目的作用。授课内容部分的分量按两学时组织。第1章的授课内容分量略轻,这是因为在第1章的授课时间中还应划分出部分时间用于介绍上机实验的编辑、调试和运行应用程序项目的基本步骤(这部分内容可参考与本书配套的学习指导书)。

自学内容 “自学内容”和“授课内容”部分一起组成了每一章的基本教学内容。这部分内容通常都是“授课内容”的延伸和继续,由学生在课外时间自学。必须强调的是自学部分并非不重要,建议不要省略。一般来说,教师应在授课时间中抽出5~10分钟对自学内容略做导引,以便利于学生自学。

程序设计举例 为了补充授课内容和自学内容部分的例题,设置了程序设计举例部分,提供了大量例题。所有例题均与本章的授课或自学等部分的内容相关,是学生学习和复习本章内容的重要参考资料。同时,本书的所有源代码都是采用标准C++编写的,程序可以使用各种C++编译器进行编译,通常不需要修改或稍做修改即可在所有编译器中运行。

实例编程 提供一个比较复杂完善的程序例子,一般是综合多个章节的知识点,解决某个实际问题,它对提高学生编程兴趣,培养学生分析问题和解决问题的能力很有帮助。这部分一般可以作为习题课或辅导课选讲的内容,也可留给有余力的学生自学。

编程提示 介绍一些编程中常见的问题和技巧,帮助学生养成良好的编程习惯,从而编写出正确、清晰易懂,易于测试维护的代码。实践证明,这部分内容对学生提高编程质量是很有帮助的。

小结 是对本章重要知识点的总结,可帮助学生复习本章的关键知识。

习题 每章均配有若干习题,供学生上机练习。这些练习题目均为程序设计题目,传统的做法是先编程,再上机。由于C++的特点,也可以在写出较详细的伪代码之后直接上机。“精讲多练”式教学方法的基本特点是上机时数较多,所以这部分的习题量较大,因此在上机时数不足的情况下可以酌情选做若干题目。

为了保证教学效果,在条件许可的情况下最好采用直接在计算机房进行的联机电化教学。在这种情况下,每个教学单元(即每章)可使用连续的4学时,先由教师讲解授课部分,并对自学部分等内容进行简短的指导(共2学时),然后学生即可在教师指导下上机练习(2学时)。除此之外,如果能够提供一定数量的课外机时(如20~30小时)则更好。

近年来,西安交通大学计算机教学实验中心在计算机基础教育的理论和实践等方面进行了一系列探索和改革,其成果(“精讲多练”的教学模式是其中之一)荣获了1997年国家级教学成果一等奖。这些成果都是在冯博琴教授(博士生导师、国家级教学名师、教育部非计算机专业计算机课程教学指导分委员会主任委员)的领导下完成的,本课程的建设也不例外。本教材的构思和编写得到了冯博琴教授的多方指导,并由他审核了书稿,在此向冯老师表示深深的谢意。同时,西安交通大学对本书的编写也非常重视,将该书列入了西安交通大学“十五”规划本科生系列教材予以重点支持。参加编写工作的队伍由西安交通大学计算机教学实验中心长期从事C++程序设计教学和科研的一线教师组成,主要人员有:罗建军(第1~8章、第12章、附录),朱丹军(第9~11章、第13章),顾刚(第14~16章)。现远在加拿大的刘路放教授也对本书的编写非常关心,并起到了非常关键的作用。在本书编写过程中,作者亦曾与杨琦、崔舒宁、刘志强、程向前、李波、杨振平和卫颜俊等诸多同事进行了多次交流,受益匪浅。以上同事还提供了一些有用的材料,在此一并表示感谢。由于作者学识浅陋,编写时间仓促,书中错误在所难免,希望读者不吝指教。本书编者的电子邮件地址分别是 jjluo@ctec.xjtu.edu.cn、djzhu@ctec.xjtu.edu.cn、ggu@ctec.xjtu.edu.cn。

编著者

2004年3月于西安交通大学

给教师教学的建议

➤ 本课程建议采取“精讲多练”的教学模式来组织教学。

➤ 要引导学生正确使用本教材,着重培养学生实际上机编程的能力,不要在具体语法细节上纠缠过多。

➤ 要引导学生使用最简单、最基本的语句编写出正确的程序,而不要沉湎于编程技巧。

➤ 本书中配备了大量在教学实践中效果比较好的例题和习题,所涉及的编程方法也很广泛,部分章节的例题和习题可能会有一定难度,教师可根据教学需要自行进行增删或调整到相应的习题课中讨论。

➤ 本书比较全面地阐述了 C++ 程序设计的基本原理和方法,每章都是一个可以讲授学习的独立模块。考虑到不同学校、不同专业的情况可能有所不同,教师可根据学生程度和课程学时的不同对章节内容酌情进行筛选。例如,对于学时比较少的课程(48 学时以下),可以重点讲授 C++ 编程的基本内容(即 1~8 章)和类与对象的基本概念(9、10 章)。

➤ 本书所有程序基本都采用标准 C++ 程序,通常不用修改或只需稍加调整即可在目前常用的各种编译器上使用。有关上机实验指导部分内容的材料,可参考本书精品课程网站上的多媒体课件和配套辅导教材《C++ 程序设计教程学习指导(第 2 版)》。

给学生学习的建议

➤ 学习程序设计的关键是“多练”，纸上谈兵学不到程序设计本领，只有大量编程、上机调试才能掌握编程技巧。勤于上机练习是学好程序设计语言的唯一途径，编程能力与在计算机上投入的时间成正比。

➤ 应该指出的是，本书不是自学教材，必须在教师教授和指导下学习，如果学生仅靠自己阅读本书来学习 C++，其效率会很低。但本书又鼓励学生自学某些内容，主要是要求学生在听完教师讲授的知识点后，扩大知识面，提高编程能力。因此，本书的自学部分是指在教师指导下的自学。

➤ 编程序最重要的是思想，仔细揣摩别人的程序（如本书中的大量例题）将获益匪浅。

➤ 即使可以很容易地从网上得到本书的所有例题源代码，也建议最好将书中的所有程序都亲手输入到计算机上编译调试，并尝试对其进行修改扩充。

➤ 编程序要肯花精力才能学好，要多看优秀程序，多做上机练习。

➤ 对已经编写好的程序，可采用某种有效的方式组织和保存好，它们也许会成为你今后工作学习中很有用的资源，同时，经常回顾以前编写的程序，并尝试用新学到的知识重写它们，会给你带来新的认识和体会。

➤ 初学者应尽量使用简单、明了的 C++ 语句完成程序的编写，而不要刻意去追求编程技巧和代码的简练。写出正确的、容易读懂的程序是最重要的。

➤ 良好的编程风格有助于程序的调试、修改、扩充以及日后的阅读。

➤ 除了传统的学习方法以外，还应该充分利用网络平台，提高获取信息、处理信息和交流信息的能力。本书附录 B 提供了一些有用的网络资源，可以参考使用。

目 录

第 1 章 C++ 语言简介	1	2.3.1 变量的定义	23
本章目标	1	2.3.2 变量的初始化	24
授课内容	1	2.4 运算符与表达式	24
1.1 C++ 程序的基本结构	1	2.4.1 算术运算符和算术 表达式	25
1.2 算法与程序	2	2.4.2 关系运算符和关系 表达式	25
1.3 C++ 程序的基本要素	4	2.4.3 逻辑运算符和逻辑 表达式	25
1.3.1 标识符、关键字和 标点符号	4	2.4.4 赋值运算符和赋值 表达式	26
1.3.2 注释	5	2.4.5 自增运算符和自减 运算符	27
1.3.3 输入与输出	5	2.5 表达式中各运算符的 运算顺序	28
1.4 分析问题、编码、运行	6	2.6 类型不同的数据之间的 混合算术运算	29
自学内容	8	自学内容	31
1.5 程序设计语言的发展	8	2.7 类型修饰符	31
1.6 C++ 语言背景介绍	10	2.8 typedef 语句	31
程序设计举例	11	2.9 常量修饰符	32
编程提示	15	2.10 八进制和十六进制 常量	32
小结	16	2.11 位运算表达式	34
习题	16	2.12 其他具有副作用的 运算符	37
第 2 章 基本数据类型与 表达式	18	程序设计举例	37
本章目标	18	编程提示	42
授课内容	18	小结	42
2.1 数据类型	18	习题	43
2.2 常量	20	第 3 章 控制结构	45
2.2.1 整型常量	20		
2.2.2 实型常量	20		
2.2.3 字符常量	21		
2.2.4 字符串常量	22		
2.2.5 布尔型常量	22		
2.3 变量	23		

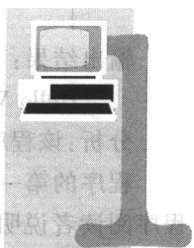
本章目标	45	习题	90
授课内容	45		
3.1 程序的基本控制结构	45	第5章 函数	91
3.2 C++的控制结构	47	本章目标	91
3.2.1 顺序结构	47	授课内容	91
3.2.2 选择结构	48	5.1 函数概述	91
3.2.3 循环结构	52	5.2 函数的定义	91
自学内容	55	5.3 函数的调用	93
3.3 C++的其他控制转移		5.4 函数间的参数传递	95
语句	55	5.4.1 值调用	95
3.4 表达式语句	57	5.4.2 引用调用	96
程序设计举例	59	5.5 函数与数组	98
实例编程——模拟仿真	63	5.6 局部变量和全局变量	99
编程提示	67	自学内容	101
小结	67	5.7 函数声明	101
习题	68	5.8 带有默认参数的函数	102
第4章 数组与字符串	69	5.9 C++的库函数	103
本章目标	69	5.10 变量的存储类别	103
授课内容	69	5.10.1 自动变量	103
4.1 数组	69	5.10.2 静态变量	104
4.1.1 一维数组	70	5.10.3 寄存器变量	105
4.1.2 二维数组	72	5.10.4 外部变量	105
4.1.3 多维数组	74	5.11 多个源程序文件程序中的	
4.2 字符型数组和字符串		全局变量说明	105
处理库函数	75	5.12 变量使用	106
4.2.1 字符型数组的定义和		程序设计举例	107
初始化	75	实例编程——万年历	110
4.2.2 字符串的输入与		编程提示	114
输出	75	小结	114
4.2.3 字符串处理库函数	76	习题	115
自学内容	78	第6章 指针	117
4.3 C++的string类	78	本章目标	117
程序设计举例	81	授课内容	117
实例编程——Josephus问题	87	6.1 地址与指针	117
编程提示	89	6.1.1 地址	117
小结	89	6.1.2 指针	118

6.2 指针运算.....	119	习题	162
6.2.1 “*”和“&”运算符	119		
6.2.2 指针变量算术运算	121		
6.2.3 指针变量比较运算	122		
6.2.4 指针变量下标运算	122		
6.3 指针与数组.....	122		
6.4 动态存储分配.....	125		
自学内容	127		
6.5 指针数组.....	127		
6.6 指向指针的指针.....	130		
6.7 指针的初始化.....	131		
程序设计举例	132		
实例编程——棋类游戏	135		
编程提示	139		
小结	139		
习题	140		
第7章 函数与指针	141		
本章目标	141		
授课内容	141		
7.1 递归函数.....	141		
7.2 函数重载.....	144		
7.3 指针和函数的关系.....	146		
7.3.1 指针作为函数的 参数	146		
7.3.2 返回指针的函数	146		
7.3.3 指向函数的指针	147		
7.4 带参数的 main()函数	149		
自学内容	150		
7.5 内联函数.....	150		
7.6 无参函数.....	151		
7.7 void 和 const 类型的 指针.....	151		
程序设计举例	152		
实例编程——最大子串和	157		
编程提示	161		
小结	161		
		第8章 结构化设计方法	163
		本章目标	163
		授课内容	163
		8.1 结构体类型.....	163
		8.1.1 结构体类型的定义	164
		8.1.2 结构体变量的初始化 和使用	165
		8.2 枚举类型.....	166
		8.3 自顶向下、逐步求精的程序 设计方法.....	170
		自学内容	172
		8.4 共用体.....	172
		8.5 数组和结构体.....	174
		8.5.1 结构体中的数组	174
		8.5.2 数组中的结构体	174
		8.6 结构体中的结构体.....	175
		8.7 结构体与指针.....	176
		8.8 编译预处理.....	177
		8.8.1 文件包含.....	177
		8.8.2 宏定义	178
		8.8.3 条件编译.....	179
		8.9 结构化程序设计方法的 发展历史.....	180
		程序设计举例	181
		实例编程——“验证”哥德巴赫 猜想	187
		编程提示	190
		小结	190
		习题	190
		第9章 类和对象(一)	192
		本章目标	192
		授课内容	192
		9.1 面向对象的基本概念.....	192
		9.1.1 对象和面向对象	192

9.1.2 面向对象的基本特征	193	程序设计举例	228
9.2 类与对象的声明和定义	195	实例编程——职工档案管理 系统	232
9.2.1 类的声明	195	小结	236
9.2.2 对象的定义	197	习题	236
9.2.3 类和对象的关系	197	第 11 章 继承	237
9.3 成员函数	197	本章目标	237
9.3.1 成员函数的定义	198	授课内容	237
9.3.2 内联成员函数	198	11.1 有关继承的基本概念	237
9.4 对象的访问	198	11.2 派生类	239
9.5 对象的存储	203	11.2.1 派生类的声明	239
自学内容	204	11.2.2 派生类的构成	240
9.6 类的组合	204	11.3 派生类的继承方式和 访问属性	242
9.7 接口与实现方法的 分离	205	11.3.1 公有继承	242
程序设计举例	207	11.3.2 私有继承	245
小结	210	11.3.3 保护继承	246
习题	210	11.4 派生类的构造函数和析构 函数	249
第 10 章 类和对象(二)	211	11.4.1 派生类的构造 函数	250
本章目标	211	11.4.2 派生类的析构 函数	250
授课内容	211	自学内容	252
10.1 构造函数	211	11.5 显式访问基类成员	252
10.1.1 对象初始化和构造 函数	211	程序设计举例	254
10.1.2 构造函数的重载	214	实例编程——象棋类	261
10.1.3 数据成员的初始化 方法	217	小结	263
10.2 析构函数	220	习题	264
10.3 对象与指针	222	第 12 章 多态性	265
10.3.1 指向对象的指针	222	本章目标	265
10.3.2 指向对象成员的 指针	223	授课内容	265
10.3.3 this 指针	225	12.1 多态性概述	265
自学内容	226	12.2 派生类对象替换基类 对象	267
10.4 string 类的成员函数	226		

12.3 虚函数	269	14.3.1 iostream 类库的头文件	316
12.3.1 虚函数定义	269	14.3.2 输入/输出流类和对象	316
12.3.2 虚函数的使用限制	271	14.3.3 输入/输出流的成员函数	317
12.4 抽象类	272	14.4 格式化 I/O	319
12.5 运算符重载	275	14.4.1 流格式状态标志和格式化函数	319
自学内容	277	14.4.2 流操纵符	322
12.6 const 修饰符	277	14.5 文件处理	325
12.7 静态成员	278	14.5.1 文件和流	327
程序设计举例	279	14.5.2 打开和关闭文件	327
实例编程——较完整的日期类 ..	284	14.5.3 文件读/写	329
小结	288	14.5.4 二进制文件	330
习题	289	14.5.5 随机访问文件	333
第 13 章 模板与异常处理	290	自学内容	335
本章目标	290	14.6 对象的输入/输出	335
授课内容	290	14.7 名字空间	336
13.1 模板	290	程序设计举例	338
13.1.1 函数模板	290	小结	339
13.1.2 类模板	292	习题	340
13.2 异常处理机制	294	第 15 章 链表	341
自学内容	297	本章目标	341
13.3 友元	297	授课内容	341
13.3.1 友元函数	297	15.1 链表概述	341
13.3.2 友元类	298	15.2 链表类	342
程序设计举例	299	自学内容	350
实例编程——矩阵类	305	15.3 顺序表	350
小结	312	程序设计举例	354
习题	313	小结	362
第 14 章 标准库和输入/输出流	314	习题	362
本章目标	314	第 16 章 程序设计与软件开发技术	363
授课内容	314	本章目标	363
14.1 标准库概述	314		
14.2 流	315		
14.3 输入/输出流	316		

授课内容	363	16.4.2 迭代器	385
16.1 面向对象的程序设计		16.4.3 算法	385
方法	363	小结	386
16.2 案例分析——电话簿		习题	386
管理程序	365		
16.2.1 问题描述	365	附录 A ASCII 码表	387
16.2.2 解决方案	366		
自学内容	382	附录 B C++ 的技术支持	389
16.3 软件开发技术基础——			
软件生命周期	382	参考文献	391
16.4 标准模板库简介	383		
16.4.1 容器	384		



本章目标

了解 C++ 程序的基本特点。

授课内容

1.1 C++ 程序的基本结构

计算机系统是由硬件和软件两大部分组成的,功能强大的硬件系统是计算机工作的基础,但如果缺乏必要的软件支持,其作用也将十分有限。

为计算机编写软件需要使用程序设计语言。目前,可用的计算机语言有数百种之多,它们各有所长,有些适用于开发数据库应用程序,有些适用于开发科学计算程序,有的简便易学,有的功能全面。本书将介绍其中使用最为广泛的 C++ 语言。

首先看一个简单例子,以便对用 C++ 语言编写程序有一个初步的认识。

【例 1-1】 第一个 C++ 程序,在计算机屏幕上显示“Hello World!”。

程序代码如下:

```
// Example 1-1: 屏幕上显示“Hello World!”  
#include <iostream> //包含基本输入/输出库文件  
using namespace std; //使用名字空间 std  
int main( ) //主函数名  
{  
    cout << "Hello World!" << endl; //屏幕显示语句  
    return 0; //表示程序顺利结束
```



```

}

```

输出结果:

```

Hello World!

```

分析:该程序非常简单,仅由一个主函数 `main()` 构成,在主函数中也只有两条语句。

程序的第一行是注释,注释以“//”开头,用于说明或解释程序段的功能、变量的作用以及向程序阅读者说明的其他任何内容。在该程序中还有其他的一些注释,所有的注释行在编译器将 C++ 程序编译成目标代码时都会被忽略掉,因此即使添加了很多注释也不会影响目标代码的效率。恰当地使用注释可以使程序清晰易懂、易于调试,便于程序员之间进行交流与协作,因此在编写程序时,精心撰写注释是一个良好的编程习惯。

第二行是编译预处理命令,用来指示编译器在对程序进行预处理时,将输入/输出标准库文件 `iostream` 的内容包含到程序中。`iostream` 文件的作用是为程序提供与输入/输出操作有关的信息,当一个程序包含有 `iostream` 文件时,就会自动定义与键盘输入相关联的输入流对象 `cin` 以及与屏幕输出显示相关联的输出流对象 `cout` 在内的多个标准流。

第三行说明使用名字空间 `std`,可保证对 C++ 标准库操作的每个特性都是唯一的,不至于发生命名冲突。

从第四行到最后一行,是主函数。主函数是该程序的主体部分,由其说明部分

```

int main( )

```

和用一对花括号 `{}` 括起来的函数体构成。

通常,一个 C++ 程序包含一个或多个函数,但其中只能有一个主函数 `main()`,C++ 程序的执行就是从 `main()` 函数开始的。`main()` 函数左边的关键字 `int` 表示 `main()` 函数会返回一个整数值,与程序的倒数第二行的 `return 0` 相对应。有关函数以及上面提到的编译预处理命令和名字空间的进一步内容,将在后续章节陆续介绍,在此只要记住每个程序中都要包含这样的语句即可。

在函数体内,有两条以分号结束的语句:第一条是输出语句,用于将字符串显示在计算机屏幕上;第二条是 `return` 语句,它放在函数的末尾,提供了终止函数执行的一种方法,该语句中的数值 `0` 表明 `main()` 函数成功运行完毕。

1.2 算法与程序

一般而言,在工程实践中所遇到的问题要比上面的例子复杂得多。要编写用于解决应用问题的程序,首先必须彻底了解问题并认真计划解决问题的方案,即算法。例如,给定两个正整数 p 和 q ,如何求出它们的最大公因数,古希腊数学家欧几里得(Euclid)给出了一个著名的算法:

- (1) 如果 $p < q$,交换 p 和 q ;
- (2) 求 p/q 的余数 r 。
- (3) 如果 $r = 0$,则 q 就是所求的结果,否则反复做如下工作:

令 $p = q, q = r$,重新计算 p 和 q 的余数 r ,直到 $r = 0$ 为止,则 q 就是原来两个正整数的最大公因数。