

高 职 高 专 计 算 机 系 列 规 划 教 材

中国计算机学会高职高专教育学组推荐出版



C++ 程序设计

(第2版)

周志德 侯正昌 编著



电子工业出版社

PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

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

高职高专计算机系列规划教材

C++程序设计

(第2版)

周志德 侯正昌 编 著

电子工业出版社

Publishing House of Electronics Industry

北京 • BEIJING

内 容 简 介

本书共 12 章, 前 2 章介绍 C++ 的概述、上机过程与 C++ 数据类型和表达式。第 3、4 章叙述三种基本程序结构、流程控制语句和数组。第 5、6 章讨论函数的定义和调用方法、函数的嵌套调用和递归调用、变量的存储类型、内联函数、重载函数、编译预处理中的宏定义、“文件包含”处理与条件编译。第 7 章讲解指针变量、指针数组、指向一维数组的指针变量、返回指针值的函数、函数指针变量、new 和 delete 运算符、引用。第 8 章介绍枚举型、结构体与链表。第 9、10 章讲述类和对象、构造函数与析构函数、继承与派生、冲突、支配规则和赋值兼容性与静态成员。第 11、12 章叙述友元与运算符重载、多态性与虚函数、流类体系与文件操作。

本书可以作为高职高专学院计算机、电子等专业的教材, 本书起点低, 不要求学过其他程序设计语言, 可作为程序设计的入门语言来学习, 也可作为从事计算机应用工作的工程技术人员培训和自学的参考书。

未经许可, 不得以任何方式复制或抄袭本书之部分或全部内容。
版权所有, 侵权必究。

图书在版编目(CIP)数据

C++ 程序设计 / 周志德, 侯正昌编著. —2 版. —北京: 电子工业出版社, 2006.2

(高职高专计算机系列规划教材)

ISBN 7-121-02232-X

I. C… II. ①周… ②侯… III. C 语言—程序设计—高等学校: 技术学校—教材 IV. TP312

中国版本图书馆 CIP 数据核字 (2006) 第 005264 号

责任编辑: 贺志洪

印 刷: 北京市李史山胶印厂

出版发行: 电子工业出版社

北京市海淀区万寿路 173 信箱 邮编 100036

经 销: 各地新华书店

开 本: 787×1092 1/16 印张: 22.75 字数: 582.4 千字

印 次: 2006 年 2 月第 1 次印刷

印 数: 5000 册 定价: 31.00 元

凡购买电子工业出版社的图书, 如有缺损问题, 请向购买书店调换。若书店售缺, 请与本社发行部联系。
联系电话: (010) 88254043。质量投诉请发邮件至 zltz@phei.com.cn, 盗版侵权举报请发邮件至 dbqq@phei.com.cn。

出版说明

高职高专的计算机专业面临着两方面的巨大变化,一方面是计算机技术的飞速发展,另一方面是高职高专教育本身的改革和重组。

当前,计算机技术正经历着高速度、多媒体及网络化的发展。计算机教育,特别是计算机专业的教材建设必须适应这种日新月异的形势,才能培养出不同层次的、合格的计算机技术专业人才。

自20世纪70年代末高等专科学校计算机专业相继成立以来,高等专科学校积极探索具有自己特色的教学计划和配套教材。1985年,在原电子工业部的支持下,由全国数十所高等专科学校参加成立了“中国计算机学会教育委员会大专教育学组”,之后又成立了“大专计算机教材编委会”。从1986年到1999年,在各校老师的共同努力下,相继完成了3轮高等专科学校计算机教材的规划与出版工作,出版了78种必修课、选修课、实验课教材,较好地解决了高層次计算机专业的教材需求。

为了适应计算机技术的飞速发展以及高职高专计算机教育发展的需要,“中国计算机学会教育委员会高职高专教育学组”和“高职高专计算机教材编委会”从2000年7月开始,又组织了本科高校、高等专科学校、高等职业技术学院和成人教育高等院校的有教学经验的老师,学习、研究、参考了“全国高校计算机专业教学指导委员会”和“中国计算机学会教育委员会”制定的高等院校《计算机学科教学计划2000》,制定了《高职高专计算机教育2002》,规划了高专、高职、成人高等教育三教统筹的第4轮教材。

第4轮教材的编写工作以招标的方式征求每门课程的编写大纲和主编,要求投标老师详细说明课程改革的思路、本课程和相关课程的联系、重点和难点的处理等。在第4轮教材的编写过程中,编委会强调加强实践环节、强调三教统筹、强调理论够用为度的原则,特别要求教学内容要适应高职高专教育发展的新形势。经过编委会、编者和出版社的共同努力,第4轮教材比前3轮教材得到了更广泛的使用,已经出版60多种。

在第4轮教材的出版过程中,得到了教育部高教司高职高专处的支持、指导和帮助,经过专家的评审,已有8种被列为“国家十五规划教材”,14种被列为“教育部规划教材”。

第4轮教材具有以下特点:

1. 在编写上突出高等职业教育的特点,强调淡化理论,加强实训,突出职业技能训练。
2. 内容反映新知识、新技术和新方法,使学生能更快地适应就业岗位的需要。
3. 对实践性较强的课程,本系列设计了主教程、上机指导教程(初级实践指导与练习)和实训教程(高级实践指导与练习)。

4. 为了满足课堂教学和教师备课的需要,教材配有电子教案或电子课件。

5. 为了配合计算机等级考试和认证考试,部分教材的习题中安排了相应的题型。

本系列教材已于2004年7月至9月陆续推出32个新品种,使得第4轮教材达到近100种,基本覆盖了高职高专计算机专业的主要课程。

“中国计算机学会教育委员会高职高专教育学组”和“高职高专计算机教材编委会”恳切希望学生、教师和专家对本套教材提出宝贵的批评和建议。

中国计算机学会教育委员会高职高专教育学组

2005年11月24日

部分学组成员单位名单

安徽淮南联合大学	河北工业职业技术学院
安徽职业技术学院	河北师范大学
保定职业技术学院	河南大学
北方工业大学	河南机电高等专科学校
北京船舶工业管理干部学院	河南新乡平原大学
北京电子信息职业技术学院	河南职业技术学院
北京科技大学职业技术学院	黑龙江大学职业技术学院
北京师范大学信息科学学院	湖北沙市大学
北京市机械局职工大学	湖南财经高等专科学校
北京信息工程学院	湖南城市学院
常州工学院	湖南大学
成都电子机械高等专科学校	湖南环境生物职业技术学院
成都航空职业技术学院	湖南计算机高等专科学校
成都师范高等专科学校	湖南民政职业技术学院
成都信息工程学院	湖南税务高等专科学校
承德石油高等专科学校	湖南铁道职业技术学院
重庆电子职业技术学院	湖州职业技术学院
重庆工业职业技术学院	淮安信息职业技术学院
佛山科技学院	淮海工学院
福建信息职业技术学院	黄石高等专科学校
福州大学职业技术学院	吉林大学
广东女子职业技术学院	吉林交通职业技术学院
广东轻工职业技术学院	吉林职业师范学院工程学院
广西水利电力职业技术学院	济源职业技术学院
广西职业技术学院	江汉大学
广州大学科技贸易技术学院	江苏常州机电职业技术学院
广州航海高等专科学校	金陵职业大学
广州市财贸管理干部学院	军械工程学院
桂林电子工业学院	空军后勤学院
哈尔滨师范大学	兰州师范专科学校
哈尔滨学院	兰州石化职业技术学院
海淀走读大学信息学院	连云港化工高等专科学校
海口经济职业技术学院	辽东学院
海南职业技术学院	辽宁交通高等专科学校
杭州经贸职业技术学院	辽阳高等职业技术学院
河北沧州职业技术学院	柳州职业技术学院
河北大学	洛阳大学

漯河职业技术学院
南京工程学院
南京建筑工程学院
南京农业专科学校
南京师范大学
南京钟山学院
南宁职业技术学院
宁波高等专科学校
青岛化工学院
青岛科技大学
青岛职业技术学院
山西大同职业技术学院
山西工业职业技术学院
山西师范大学
山西水利职业技术学院
陕西工业职业技术学院
上海第二工业大学
上海电机技术高等专科学校
上海交通大学应用技术学院
上海理工大学
上海旅游高等专科学校
上海商业职业技术学院
上海托普信息技术学院
上海托普职业技术学院
上海应用技术学院
韶关大学
邵阳高等专科学校
深圳职业技术学院
沈阳电力高等专科学校
四川师范学院
四川托普信息职业技术学院

苏州市职工大学
苏州铁路机械学校
苏州职业大学
台州职业技术学院
泰州职业技术学院
天津滨海职业学院
天津渤海职业技术学院
天津大学高职学院
天津电子信息职业技术学院
天津轻工业学院
天津师范大学计算机与信息工程学院
潍坊高等专科学校
温州大学
无锡职业技术学院
武汉职业技术学院
西安电子科技大学
新疆农业职业技术学院
兖州矿区职业大学
云南财贸学院
云南民族大学职业技术学院
浙江大学
浙江工贸职业技术学院
浙江工商大学
浙江育英学院
郑州工业高等专科学校
郑州经济管理干部学院
郑州经济管理学院
中国保险管理干部学院
中国地质大学
中国人民大学成人教育学院
中州大学

前 言

《C++程序设计》介绍一种目前流行的面向对象程序设计语言,是学习数据结构等计算机课程的基础,是学习可视化程序设计软件的语言基础,是计算机专业及许多工科专业学生应掌握的一门计算机语言课程。原教材是根据《中国计算机学会高职高专教育学会》2001年审定的编写大纲编写的,主要介绍C++中的数据类型、程序结构与控制语句、数组、函数、结构体、指针、类、对象、继承、重载、多态性技术、文件操作的基本概念与方法。自2003年教材出版以来,作者在使用本教材进行教学的过程中,又积累了许多教学经验,总结归纳出一些新教学方法,并根据高职高专要求对教学内容做了适当删减。为此,有必要对本教材进行修订,将这些教学经验与教学方法供读者共享。修订后的教材具有以下特点。

1. 本教材将《C语言程序设计》与《C++程序设计》综合为一本教材,用C++的语言来描述原先用C语言描述的内容,然后再加上面向对象的程序设计内容。这样做的好处是:

(1) 学生可以从数据类型、程序结构等基础内容开始由浅入深地进行学习,所以本教材起点低,可作为程序设计的入门教材来学习。

(2) 由于将两门课综合为一门课,所以可减少总的教学时间。可在一学期内完成本门课的学习。

(3) 可以使学生直接学习面向对象的程序设计方法。

2. 针对高职高专类学生的特点,本书尽可能使用通俗易懂的语言来叙述各章节内容,并尽可能使用典型例题来说明各章节知识点的概念与使用方法,力求将各章节的重点难点解释清楚,以求多数学生在教师讲课后能看懂教材,学会知识的应用。

3. 由于描述C++类与对象的程序段一般都较长,因此本书中尽量用同一类型的例题介绍某个系列概念,例如,用描述学生成绩类讲解类与对象的概念、定义及使用方法。用描述矩形的类介绍构造函数、拷贝构造函数、默认构造函数、析构函数等一系列的概念、定义及使用方法,以减少教师在黑板上的书写工作量,提高课堂的讲课效率。

4. 对学生比较难理解的内容,采用先通过例题分析,然后引出基本概念,给出定义格式、结论等正式内容。

5. 高职高专学生在学习C++程序设计时遇到的困难之一是:一方面要理解C++中许多比较难的概念,另一方面又要理解复杂的算法。为了解决该问题,本教材在程序的算法上重点抓住常用的一些典型算法,如:累加和、累乘积、最大值、最小值、平均值、排序,并将这些典型算法作为介绍各章节基本概念例题,这样可减轻学生在理解算法上的负担,提高课堂的教学效果。一些有较复杂算法的例题只会出现在每章后的程序设计应用举例中。

6. 每章前都有学习本章目的、要求,每章后有本章小结,并配有一定量的习题,便于教师教学和学生自学。各章内容充实,安排合理,衔接自然。

7. 每章后增加实验部分,其包含实验目的、实验要求、实验内容、实验提示等。将实验指导书综合在教材内方便教师教学与学生做实验。

8. 在部分章节中增加了习题课内容,习题课采用边小结内容边举例的方式进行。

9. 对于某些算法(如三种排序算法)尽可能用较简练的语句描述出算法核心含义,使读

者易理解、易记忆。

10. 从第 5 章开始以后的内容, 凡是用到函数概念的知识点, 如指针变量作为函数参数、返回指针值的函数、结构体数组作为函数参数、运算符重载函数, 均以函数的定义、调用、参数传送三步进行描述。这样做的好处是: 一方面, 通过多次重复使学生能更好地掌握函数的定义、调用、参数传送的概念, 另一方面, 多次使用函数的定义、调用、参数传送三步方式更容易引入新知识点的概念。

在本书编写的过程中, 参考了目前国内比较优秀的有关 C++ 语言程序方面的书籍资料, 在此谨向有关作者表示感谢。对电子工业出版社老师在修订工作中给予的大力支持表示衷心的感谢。

本书第 1~6 章由侯正昌编写, 第 7~12 章由周志德编写, 全书由无锡职业技术学院周志德副教授统编, 南京邮电学院唐瑞庭副教授审阅。在本书编写的过程中, 刘德强、许敏、颜惠琴、王得燕老师对书中内容提出了非常宝贵的意见, 在此, 仅对上述老师的工作表示感谢。

本书若有错误及不足之处, 恳请读者给予指正。

编 者

2005 年 11 月 24 日

目 录

第1章 C++概述	(1)
1.1 C++的起源	(1)
1.2 C++的特点	(1)
1.3 C++程序的基本结构	(2)
1.4 C++上机操作	(5)
1.4.1 C++程序的开发步骤	(5)
1.4.2 C++程序上机操作方法	(6)
本章小结	(9)
习题1	(10)
第2章 数据类型和表达式	(11)
2.1 数据类型	(11)
2.2 常量和变量	(13)
2.2.1 常量	(13)
2.2.2 变量	(16)
2.3 运算符和表达式	(17)
2.3.1 算术运算符和算术表达式	(17)
2.3.2 赋值运算符和赋值表达式	(19)
2.3.3 自增、自减运算符	(20)
2.3.4 关系运算符和关系表达式	(21)
2.3.5 逻辑运算符和逻辑表达式	(22)
2.3.6 逗号运算符和逗号表达式	(24)
2.3.7 复合赋值运算符	(25)
2.3.8 数据类型长度运算符 (sizeof 运算符)	(25)
2.4 简单输入和输出	(26)
2.4.1 数据输出 cout	(26)
2.4.2 数据输入 cin	(27)
2.4.3 简单输入/输出格式控制	(28)
本章小结	(29)
习题2	(30)
实验二	(32)
第3章 程序结构和流程控制语句	(33)
3.1 程序的三种基本结构和语句	(33)
3.1.1 程序的三种基本结构	(33)
3.1.2 C++程序的组成	(34)
3.1.3 C++程序的语句	(35)
3.2 分支语句	(36)

3.2.1	if 语句	(36)
3.2.2	条件运算符和条件表达式	(41)
3.2.3	switch 语句	(42)
3.3	循环语句	(45)
3.3.1	while 语句	(45)
3.3.2	do...while 语句	(47)
3.3.3	for 语句	(49)
3.3.4	三种循环语句的比较	(52)
3.3.5	循环语句的嵌套	(52)
3.4	控制执行顺序的语句	(53)
3.4.1	break 语句	(53)
3.4.2	continue 语句	(54)
3.4.3	语句标号和 goto 语句	(55)
3.4.4	exit() 和 abort() 函数	(56)
3.5	程序设计举例(习题课)	(57)
3.5.1	分支语句应用举例	(57)
3.5.2	循环语句应用举例	(61)
本章小结		(65)
习题 3		(67)
实验三(1)		(70)
实验三(2)		(71)
第 4 章 数组		(73)
4.1	数组的定义和应用	(73)
4.1.1	一维数组的定义和使用	(73)
4.1.2	二维数组的定义和使用	(80)
4.2	字符数组的定义和使用	(84)
4.2.1	字符数组和字符串	(84)
4.2.2	字符串处理函数	(87)
4.3	数组应用举例(习题课)	(90)
4.3.1	一维数组应用举例	(90)
4.3.2	二维数组应用举例	(92)
4.3.3	字符数组应用举例	(95)
本章小结		(96)
习题 4		(97)
实验四(1)		(100)
实验四(2)		(101)
第 5 章 函数		(103)
5.1	函数的定义和调用	(103)
5.1.1	函数的概念	(103)
5.1.2	函数的定义	(104)

5.1.3	函数的调用	(105)
5.1.4	实参与形参的数据传送	(108)
5.2	函数的嵌套调用和递归调用	(109)
5.2.1	函数的嵌套调用	(109)
5.2.2	函数的递归调用	(110)
5.3	数组作函数参数(习题课)	(113)
5.3.1	数组元素作函数实参	(113)
5.3.2	数组名作函数参数	(114)
5.4	变量的存储类型	(118)
5.4.1	作用域	(118)
5.4.2	局部变量与全局变量	(120)
5.4.3	动态与静态变量	(121)
5.4.4	变量的存储类型	(121)
5.5	内联函数	(127)
5.6	具有默认参数值的函数	(128)
5.7	函数的重载	(129)
	本章小结	(130)
	习题5	(133)
	实验五	(137)
第6章	编译预处理	(139)
6.1	文件包含处理	(139)
6.2	宏定义	(142)
6.2.1	不带参数的宏定义	(142)
6.2.2	带参数的宏定义	(144)
6.3	条件编译	(146)
	本章小结	(149)
	习题6	(150)
第7章	指针	(152)
7.1	指针与指针变量	(152)
7.1.1	指针的概念	(152)
7.1.2	指针变量的定义与引用	(152)
7.1.3	指针变量的运算	(154)
7.2	指针与数组	(159)
7.2.1	一维数组与指针	(159)
7.2.2	二维数组与指针	(160)
7.2.3	字符串与指针	(164)
7.3	指针变量与数组作为函数参数(习题课)	(166)
7.3.1	指针变量作为函数参数	(166)
7.3.2	数组与指针作为函数参数	(168)
7.4	指针数组和指向一维数组的指针变量	(171)

7.4.1	指针数组	(171)
7.4.2	指向一维数组的指针变量	(173)
7.5	返回指针值的函数与函数指针变量	(175)
7.5.1	返回指针值的函数	(175)
7.5.2	函数指针变量	(178)
7.6	new 和 delete 运算符	(181)
7.6.1	new 运算符	(181)
7.6.2	delete 运算符	(182)
7.6.3	使用 new 和 delete 运算符应注意的事项	(183)
7.7	引用类型变量和 const 类型的指针	(184)
7.7.1	引用类型变量的定义及使用	(184)
7.7.2	const 类型变量	(185)
	本章小结	(187)
	习题 7	(190)
	实验七 (1)	(193)
	实验七 (2)	(194)
第 8 章	枚举型和结构体	(195)
8.1	枚举类型的定义及应用	(195)
8.1.1	枚举类型的定义	(195)
8.1.2	枚举类型变量的定义	(196)
8.1.3	枚举类型变量的引用	(197)
8.2	结构体的定义及应用	(200)
8.2.1	结构体的概念	(200)
8.2.2	结构体类型的定义	(200)
8.2.3	结构体变量的定义	(201)
8.2.4	结构体变量的引用	(202)
8.2.5	结构体变量与数组作为函数参数	(204)
8.3	链表	(208)
8.3.1	链表的概念	(208)
8.3.2	链表的基本操作	(209)
	本章小结	(218)
	习题 8	(219)
	实验八 (1)	(220)
	实验八 (2)	(222)
第 9 章	类和对象	(223)
9.1	概述	(223)
9.2	类与对象	(225)
9.2.1	类	(225)
9.2.2	对象	(229)
9.3	构造函数	(232)

9.3.1	构造函数的定义	(232)
9.3.2	用构造函数初始化对象的过程	(234)
9.3.3	默认构造函数	(235)
9.3.4	拷贝的构造函数	(236)
9.3.5	用 new 运算符动态定义对象	(237)
9.4	析构函数	(239)
9.4.1	定义析构函数	(239)
9.4.2	析构函数的调用	(239)
9.4.3	默认的析构函数	(243)
9.5	构造函数和对象成员	(243)
9.6	this 指针	(246)
本章小结		(247)
习题 9		(248)
实验九		(251)
第 10 章	继承和派生类	(253)
10.1	继承与派生	(253)
10.1.1	继承与派生的基本概念	(253)
10.1.2	派生类的定义	(255)
10.1.3	派生类的构造函数与基类成员的初始化	(257)
10.2	冲突、支配规则和赋值兼容性	(262)
10.2.1	冲突	(262)
10.2.2	支配规则	(264)
10.2.3	赋值兼容规则	(266)
10.2.4	基类和对象成员的几点说明	(266)
10.3	虚基类	(266)
10.3.1	多重派生的基类拷贝	(266)
10.3.2	虚基类	(268)
10.4	静态数据成员	(269)
本章小结		(272)
习题 10		(274)
实验十		(276)
第 11 章	友元与运算符重载	(278)
11.1	友元函数	(278)
11.1.1	定义普通函数为友元函数	(278)
11.1.2	友元注意事项	(279)
11.2	运算符重载	(280)
11.2.1	运算符重载的概念	(280)
11.2.2	二元运算符重载	(280)
11.2.3	一元运算符重载	(285)
11.2.4	字符串类运算符重载	(293)

11.3 多态性与虚函数	(297)
11.3.1 多态性技术	(297)
11.3.2 虚函数	(297)
11.3.3 纯虚函数	(300)
11.4 类与对象的特性	(301)
本章小结	(302)
习题 11	(304)
实验十一	(306)
第 12 章 流类体系与文件操作	(308)
12.1 流类体系	(308)
12.1.1 流 (Stream)	(308)
12.1.2 基本流类体系	(309)
12.1.3 标准输入/输出流	(310)
12.1.4 流的格式控制	(311)
12.1.5 数据输入/输出成员函数	(318)
12.2 文件操作	(320)
12.2.1 C++文件概述	(320)
12.2.2 C++的文件流类体系	(320)
12.2.3 文件的使用方法	(321)
12.2.4 文本文件的使用	(325)
12.2.5 二进制文件的使用	(330)
本章小结	(339)
习题 12	(342)
实验十二	(343)
附录 A C++中的关键字	(345)
附录 B 常用库函数	(347)
参考文献	(350)

第1章 C++概述

通过本章学习，应了解 C++ 的起源及其特点。掌握 C++ 程序的基本结构，会编写极简单的 C++ 程序。了解 C++ 程序的开发步骤，熟悉 Visual C++ 集成环境，初步掌握 C++ 程序的上机操作方法。

1.1 C++的起源

C++ 是在 C 语言的基础上逐步发展和完善起来的，因此介绍 C++ 不妨首先回顾一下 C 语言的发展。

1967 年，Martin Richards 为编写操作系统软件和编译程序开发了 BCPL 语言（Basic Combined Programming Language）；1970 年，Ken Thompson 在继承 BCPL 语言的许多优点的基础上开发了实用的 B 语言；1972 年，贝尔实验室的 Dennis Ritchie 在 B 语言的基础上，做了进一步的充实和完善，开发出了 C 语言。当时，设计 C 语言是为了编写 UNIX 操作系统，以后，C 语言经过多次改进，逐渐开始流行。目前常用的 C 语言版本基本上都是以 ANSI C 为基础的。

C 语言具有许多优点，比如语言简洁灵活，运算符和数据结构丰富，具有结构化控制语句，程序执行效率高，同时具有高级语言和汇编语言的优点等。与其他高级语言相比，C 语言具有可以直接访问物理地址的优点，与汇编语言相比又具有良好的可读性和可移植性。因此，C 语言得到了极为广泛的应用。

随着 C 语言应用的推广，C 语言存在的一些缺陷或不足也开始暴露出来，并受到大家的关注。比如 C 语言对数据类型检查的机制比较弱，缺少支持代码重用的结构；随着软件规模的扩大，难以适应开发特大型程序。同时 C 语言毕竟是一种面向过程的编程语言，已经不能满足运用面向对象的方法开发软件的需要。C++ 便在 C 语言基础上，为克服 C 语言本身存在的缺点，同时为支持面向对象的程序设计而研制出来的一种通用的程序设计语言，它是在 1980 年由贝尔实验室的 Bjarne Stroustrup 创建的。

研制 C++ 的一个重要目标是使 C++ 是一个更好的 C，所以 C++ 根除了 C 中存在的问题；另一个重要目标就是面向对象的程序设计，因此在 C++ 中引入了类的机制。最初的 C++ 被称为“带类的 C”，1983 年正式命名为 C++（C Plus Plus）。以后经过不断完善，形成了目前的 C++。

当前运用得较为广泛的 C++ 有 Microsoft 公司的 Visual C++（简称 VC++）和 Borland 公司的 Borland C++（简称 BC++）。本书以 Microsoft Visual C++ 6.0 集成环境为例介绍 C++ 语言。

1.2 C++的特点

C++ 的主要特点表现在两个方面：一是全面兼容 C 语言，二是支持面向对象的程序设计

方法。

(1) C++是一个更好的 C，它保持了 C 语言的优点，大多数的 C 程序代码略作修改或不作修改就可在 C++的集成环境下调试和运行。这对于继承和开发当前已在广泛使用的软件是非常重要的，可以节省大量的人力和物力。

(2) C++是一种面向对象的程序设计语言。它使得程序中各个模块的独立性更强，程序的可读性和可移植性更强，程序代码的结构更加合理，程序的扩充性更强。这对于设计、编制和调试一些大型的软件尤为重要。

(3) C++集成环境不仅支持 C++程序的编译和调试，而且也支持 C 程序的编译和调试。通常，C++集成环境约定：当源程序文件的扩展名为.c 时，则为 C 程序；而当源程序文件的扩展名为.cpp 时，则为 C++程序。本书中，所有例题程序的文件扩展名均为.cpp。

(4) C++的语句非常简练，对语法限制比较宽松，因此 C++语法非常灵活。其优点是给用户编程带来书写上的方便。其缺点是由于编译时对语法限制比较宽松，许多逻辑上的错误不容易发现，给用户编程增加了难度。

1.3 C++程序的基本结构

为了说明 C++程序的基本结构先举三个例题，然后通过三个例题引出 C++程序的基本结构。

【例 1.1】 文本的原样输出。文件名为 example1_1.cpp。

```
//文本原样输出程序
#include <iostream.h>
void main(void)
{   cout<<"Welcome to C++!\n";
}
```

该程序经编译和连接后，运行可执行程序时，在显示器上显示：

```
Welcome to C++!
```

该程序中，main()表示主函数，每个 C++程序必须有且只能有一个主函数，C++程序总是从主函数开始执行的。main()函数之前的 void 表示 main()函数没有返回值，main()函数后括号内的 void 表示 main()函数没有形式参数。在花括号内的部分是函数体，函数体由语句组成，每个语句由分号结束。cout 是 C++程序中的一个输出流，与符号“<<”结合使用可以输出常量、变量的值及原样输出双引号中的字符串。“\n”是换行符，即输出上述信息后换行。

程序中的#include 是 C++编译预处理中的文件包含命令，“iostream.h”是头文件，为了能使用输出流 cout 和输入流 cin，程序开头必须用#include 命令将文件 iostream.h 中的内容包含到本文件中来。

程序中以“//”开头的是注释，注释是对程序的说明，用来提高程序的可读性，可以放在程序的任何位置，对程序的编译和运行不起作用。

【例 1.2】 求两个整数的和。

```
/*求两个整数的和程序*/
```

```

#include <iostream.h>

void main(void)
{
    int a,b,sum;           //说明变量 a,b,sum 为整型数
    cout<<"Input a,b:";    //显示提示信息
    cin>>a>>b;             //从键盘上输入变量 a、b 的值
    sum=a+b;               //求和
    cout<<"Sum="<<sum<<endl; //输出结果
}

```

该程序经编译和连接后，运行可执行程序时，在显示器上显示：

```

Input a,b: 3 5
Sum =8

```

该程序中的语句：int a,b,sum; 用来说明变量 a、b、sum 为 int（整型）变量。程序中的语句：sum=a+b; 是一个赋值语句，表示将 a 和 b 的值相加，其结果送给变量 sum。在“/*”和“*/”之间的部分也表示注释。“endl”是换行符。

【例 1.3】 输入两个整数 a 和 b，用自定义函数 add()求两数和。

```

#include <iostream.h>

int add(int x,int y)
{
    int z;
    z=x+y;
    return z;
}

void main(void)
{
    int a,b,sum;
    cout<<"Input a,b:";
    cin>>a>>b;
    sum=add(a,b);
    cout<<"Sum="<<sum<<endl;
}

```

该程序经编译和连接后，运行可执行程序时，在显示器上显示：

```

Input a,b: 3 5
Sum=8

```

该程序由两个函数组成：主函数 main()和被调用函数 add()。函数 add()的作用是求 x 和 y 的和，并赋给 z，最后通过 return z 语句返回给主函数。主函数用来输入两个变量 a 和 b 的值，调用 add()函数将变量 a、b 的值传送给形参 x、y，再求两数和，并返回给 sum 输出结果。

通过【例 1.3】，可以归纳出 C++程序的基本结构如下：

(1) C++程序由包括 main()在内的一个或多个函数组成，函数是构成 C++程序的基本单位。其中名为 main()的函数称为主函数，可以将它放在程序的任何位置。但是，不论主函数放在程序的什么位置，一个 C++程序总是从主函数开始执行，由主函数来调用其他函数。所以，任