高校计算机教学系列教材

C语言与C++语言程序设计

陈炳和 编著

北京航空航天大学出版社

C语言与C++语言程序设计

陈炳和　编著

北京航空航天大学出版社

内 容 简 介

C语言是目前正被广泛应用的功能强大的计算机程序设计语言，而C++语言是面向对象的程序设计语言的杰出代表。本书将C语言和C++语言贯通起来，其内容由浅入深、循序渐进、通俗易懂；还充分考虑了初学者的特点，重点突出，深入讲解，强调应用。

全书共分14章，第1～9章为基础部分，主要介绍C语言程序设计；第10～13章为提高部分，主要讲解C++语言程序设计；第14章简介Windows应用程序开发。每章都配有习题，书末附有习题的参考答案。全书所有例题都在VC++6.0环境下调试通过，并给出了运行结果，十分便于阅读。本书不再讲解DOS下运行的C语言编译软件，一方面因其落后，另一方面对C和C++使用一种编译软件可以带来不少方便。

本书是以初学计算机高级语言程序设计的高等院校(本、专科)学生为对象而编写的教材。可作为高等院校(本、专科)计算机高级语言程序设计课程的基础教材和相关专业的培训教材，也可作为自学C/C++语言程序设计的教材，并可供有关专业的教师和技术人员参阅。

图书在版编目(CIP)数据

C语言与C++语言程序设计/陈炳和编著. —北京：北京航空航天大学出版社，2004.2

ISBN 7-81077-414-X

Ⅰ.C… Ⅱ.陈… Ⅲ.C语言—程序设计 Ⅳ.TP312

中国版本图书馆CIP数据核字(2004)第007268号

C语言与C++语言程序设计

陈炳和 编著

责任编辑 王 瑛

*

北京航空航天大学出版社出版发行

北京市海淀区学院路37号(100083) 发行部电话：(010)82317024 传真：(010)82328026

http://www.buaapress.com.cn E-mail:bhpress@263.net

河北省涿州市新华印刷厂印装 各地书店经销

*

开本：787×1092 1/16 印张：29.75 字数：762千字

2004年2月第1版 2004年2月第1次印刷 印数：5 000册

ISBN 7-81077-414-X 定价：38.00元

总 前 言

科教兴国，教育先行，在全国上下已形成共识。在教育改革过程中，出现了多渠道、多形式、多层次办学的局面。同时，政府逐年加大教育的投入力度。教育发展了，才能有效地提高全民族的文化、科学素质，使我们中华民族屹立于世界民族之林。

计算机科学与技术的发展日新月异，其应用领域迅速扩展，几乎无处不在。社会发展的需求，促使计算机教育生气蓬勃。从普通高校的系统性教学，到远距离的电视、网上教学；从全面讲述，到不同应用领域的、星罗棋布的培训班；从公办的，到民办的；从纸介教材，到电子教材等等，可以说计算机教学异彩纷呈。要进行教学，就必须有教材。

面对我们这么大的国家和教学形势，在保证国家教学基本要求的前提下，应当提倡教材多样化，才能满足各教学单位的需求，使其形成各自的办学风格和特色。为此，我们组织北京工业大学、北京航空航天大学、北京理工大学、南开大学、天津工业大学等高校的有丰富教学经验的教师编写了计算机教学系列教材，将陆续与师生见面。

系列教材包括以下各项。

（一）基础理论：离散数学。

（二）技术基础：电路基础与模拟电子技术；数字逻辑基础；计算机组成与体系结构；计算机语言（拼盘，选择使用），包括 C＋＋程序设计基础、Visual Basic 程序设计基础、Matlab 程序设计基础、Java 程序设计基础、Delphi 语言基础、汇编语言基础等；数据结构；计算机操作系统基础；计算方法基础；微机与接口技术；数据库技术基础等。

（三）应用基础：计算机控制技术；网络技术；软件工程；多媒体技术等。

（四）技术基础扩展：编译原理与编译构造；知识工程——网络计算机环境下的知识处理。

（五）应用基础扩展：计算机辅助设计；单片机实用基础；图形、图像处理基础；传感器与测试技术；计算机外设与接口技术。

本系列教材主要是针对计算机教学编写的，供普通高校、社会民办大学、高等职业学校、业余大学等计算机专业本科或专科选用。其中一部分教材也适合非计算机专业本科教学使用。在这些教材的内容简介或前言中对使用范围均作了说明。

本系列教材在编写时，注重以下几点：(1) 面对计算机科学与技术发展的现实，在内容上应具有前瞻性；(2) 学以致用，既有系统的基础知识，又具有应用价值；(3) 具有科学性、严谨性。另外，力求排版紧凑，使有限的版面具有最大的信息量，以使读者得到实惠。

能否实现这些愿望，只有师生在教学实践中评价。我们期望得到师生的批评和指正。

高校计算机教学系列教材编委会

高校计算机教学系列教材编委会成员

前 言

人是通过计算机语言与计算机进行通信的。人们已创建了多种计算机语言，而C语言和C++语言是当今应用最广泛的计算机语言。随着计算机科学技术的发展，C语言也在不断地完善和发展。它所以常用不衰，是因为C语言具有独特的风格：语言简洁灵活，运算和数据结构丰富，具有结构化控制语句，程序执行效率高，可移植性好，功能强等；但是C语言毕竟是一种面向过程的编程语言，已不能满足软件开发的要求。

当前蓬勃发展的面向对象的程序设计，是计算机程序设计的崭新模式，使计算机解决问题的方式更接近于人类的思维活动。这是目前程序设计和软件开发的主流。C++语言是从C语言发展起来的，全面兼容C语言；C++语言又是面向对象程序设计语言的杰出代表，当前有逐步代替C语言的趋势。

本书是以初学计算机高级语言程序设计课程的高等院校（本、专科）学生为对象而编写的。考虑到C语言正在被广泛应用，又必须学习C++语言；又考虑到C++语言与C语言兼容的部分，虽然不是C++的主要部分，但是依然须涉及它。因此本书既介绍C语言，以它作为学习计算机高级语言的开始，并为学习C++语言做好准备，同时又介绍C++语言，以赶上时代的步伐。这样可使学生和广大读者由浅入深，打好基础再学习新理论、新技术和新方法。这样学习既系统，又便于掌握，还可节省学时。

有了C语言基础，学习和掌握C++语言就更容易。C的概念在C++中获得了长足的发展，学习了C++有助于深刻理解C。将C与C++结合起来学习，很有特点和优势：既解决了目前只熟悉C，不熟悉C++的问题，又降低了直接学习C++给初学者带来的难度。先易后难地组织教材和教学是可取的，如在C++中涉及到面向过程的语法基础，而在学习C语言的过程中，这种语法基础很容易被掌握，这就为学习C++铺平了道路。

本教材十分重视编译，用了不少篇幅进行讲解，以引导应用。因为C/C++程序只有经过编译才能运行，并得到运行结果；编程中的错误和问题也只有在这个过程中才能有效地

被发现。通过检验程序来弥补程序中的缺陷和改正程序中的错误是不可忽视的教学环节，是必须经过的过程。

本书是北京航空航天大学出版社出版的高校计算机教学系列教材之一。全书分为14章：第1～9章为基础部分，主要介绍C语言；第10～13章为提高部分，主要讲解C++语言；第14章简介Windows应用程序开发。因为学习C++语言时，涉及类的理论，以及抽象性、封装性、继承性和多态性等诸多属性。它们在Windows应用程序中得到了充分的发挥。只有应用程序，C++的特征才能展示出来。通过学习这一章，可使读者对C++语言涉及的理论、应用和发展有一个更全面的理解；可使读者更深刻地认识到，要成为一名出色的Windows程序员，还须继续深入学习和实践。

本书的起点是基础知识，始终以编程技术为主线，以应用解决问题为目标，内容由浅入深，循序渐进，重点突出，深入讲解。每章都配有习题，目的主要是帮助读者消化吸收本章的重点内容。选题时考虑了初学者的特点。书中全部例题都在VC++6.0环境下调试通过，并给出了运行结果。本书不再讲解DOS下运行的C语言编译软件，不仅仅因其太落后，而且对C和C++使用不同的编译软件也不方便。

在本书编写过程中，得到了北京航空航天大学和北京工业大学的许多老师和朋友的鼓励、支持和帮助，在此表示衷心的感谢；笔者参阅了国内外许多教材和文献，吸收了新的内容并借鉴了新的思路，对这些书的作者和译者提供的帮助，深表谢意。由于计算机科学与技术发展得极为迅速，又由于学识有限，书中会有不少疏漏和不妥之处，请读者批评指正。

陈炳和

2003年7月

目 录

第1章 引 论

本章沿着历史的轨迹，介绍面向过程的C语言和面向对象的C++语言。

1.1 C语言与C++语言

20世纪60年代，马丁·理查兹(Martin Richards)设计出BCPL(Basic Combined Programming Language)语言。1970年肯·汤普森(Ken Thompson)在BCPL语言的基础上，提出了比较实用的B语言。当时DEC公司的PDP—7计算机中的UNIX操作系统就是使用B语言开发的。C语言是在B语言的基础上于1972年由贝尔实验室的丹尼斯·M·里奇(Dennis·M·Ritchie)设计的一种程序设计语言。30多年来，C语言不断完善和发展，至今仍被业界普遍使用。C语言所以长用不衰，是因其有许多优点：语言简洁灵活，运算和数据结构丰富，具有结构化控制语句，程序执行效率高，同时具有汇编语言和高级语言的功能，以及可移植性好等；但是，毕竟C语言是一种面向过程的编程语言，已不能满足蓬勃发展的面向对象的软件开发方法的需要。1980年，美国电报电话公司(AT&T)贝尔实验室研究员比阿恩·斯特劳斯特鲁普(Bjarne Stroustrup)为C语言扩充了一系列新功能，并将其命名为带类的C(C with Class)，1983年正式将其命名为C++。C++支持面向对象的程序设计，经过不断的完善，成为目前的C++语言。

C++语言从C语言发展而来，全面兼容C语言，但是它与C语言的程序设计思想是完全不同的。考虑到C语言正在被广泛应用，又必须学习C++语言；又考虑到，虽然C++语言与C语言兼容的部分不是C++的主要部分，但是依然须涉及C。因此本书沿着历史的进程，先介绍C语言，再介绍C++语言。从基础进入，然后再迈向新理论、新方法和新潮流。

1.2 计算机程序设计语言概述

语言是由词汇和语法组成的符号系统，是人类交际的工具，人们运用它进行思维、表达及交流思想，通常称之为自然语言，例如汉语和英语等。计算机语言是计算机可以识别的语言，是操作人员和计算机进行交流的工具。计算机只能识别0和1两种数码组成的二进制数字(代码)。计算机完成数值计算、数据处理、图形图像处理、声音处理和实时控制等任务，是通过指令来实现的。每条指令都是用0和1(二进制数)组成的一串代码。其格式和所代表的含义，都具有严格的规定。指令的集合构成程序，也就是说，程序是由一系列指令所组成的。计算机语言也有自己的语句定义和语法结构，是人为规定的，在使用计算机语言时，必须遵守这些规定。

1.2.1 机器语言

指令是计算机可以执行的命令，通过指令可让计算机执行指定的操作。用0和1组成的二进制数字代码表示的指令，被通称为二进制指令或机器指令，而由二进制指令组成的语言称为机器语言(Machine Language)。

机器语言虽然是计算机唯一无须翻译可直接识别的语言，但是它与机器有关，针对一种计算机编写的程序，不能在另一种计算机上运行。机器语言不仅可移植性差，而且在编程时，须记住计算机的指令代码，这也是相当困难的。使用机器语言开发周期长，易于出错，局限性很大。

1.2.2 汇编语言

为了克服用机器语言编写程序的困难，于是将机器指令映射为一些容易被读懂和便于被记忆的助记符，如ADD(代表"加")，SUB(代表"减")等，这些是英文名称的缩写。这种用指令符号编码的语言，称之为汇编语言(Assemble Language)。汇编语言一般与机器语言一一对应(一条对一条，或一条对几条)。用汇编语言编写的程序被称为汇编语言程序。计算机硬件只能识别机器指令，两者转换需要一种翻译软件。汇编语言的翻译软件被称为汇编程序。它将源程序的助记符直接翻译成机器指令，再由计算机去识别和执行。翻译前的汇编语言程序被称为源程序，翻译出来的用机器语言表示的程序被称为目标程序。

汇编语言与机器语言一样，都是面向机器的语言，只是机器语言使用指令代码编写程序，而汇编语言使用指令助记符编写程序。汇编语言由于抽象的层次太低，编程人员仍须考虑大量的机器细节，且没有通用性。尽管如此，从机器语言到汇编语言，还是大大地前进了一步。

1.2.3 高级语言

虽然汇编语言占用内存小，用它编写的程序运行效率较高，常用它来编写系统软件、外部设备或端口数据输入/输出程序等；但是，不管是机器语言，还是汇编语言，都与具体机器有关，无通用性，被称为低级语言。

对于抽象层次比较高，不依赖于计算机类型的计算机语言，称之为高级语言(High Level Language)。计算机不能直接识别和运行用高级语言编写的程序，必须用翻译程序将用高级语言编写的程序(源程序)翻译成机器语言程序(目标程序)。这种翻译程序又分为编译程序和解释程序两种。

编译程序的功能是把高级语言所编写的程序(源程序)翻译成机器语言程序(目标程序)，再由计算机去执行，如Fortran语言编译程序。

解释程序的功能是将高级语言编写的程序(源程序)，按语句的顺序逐句进行分析、翻译。解释一句，执行一句，不保存解释后的机器代码，下次运行此程序时，再重新解释执行，如Basic语言解释程序。

前述的机器语言和汇编语言是面向机器的低级语言；现在被广泛应用的几种语言如Fortran，Basic，Pascal及C等，都是面向过程的高级语言；目前最热门的C++等高级语言，是面向对象的语言。计算机语言每前进一步，都使编程语言与人类自然语言(或数学语言)更加接近。

1.3 结构化程序设计与面向对象的程序设计

本节介绍程序设计的一般方法。程序设计技术包括程序设计方法和程序设计语言。20世纪60年代末开始发展起来的结构化程序设计方法，使程序设计的水平提高了一大步。C语言用函数来组织程序，是一种结构化程序设计语言。面向对象的程序设计方法是在结构化程序设计的基础上发展起来的新方法。它以抽象的类的理论为基础，引入对象这一构建程序的基本构件，以崭新的概念和方法，提供给人们期待的特性，即用一般的思维方法来开发软件。C++语言即为面向对象的程序设计语言的代表。下面分别作介绍。

1.3.1 程序设计

计算机系统包括硬件和软件两大部分。软件包括各种程序和相关的文档资料。软件的开发和生产已成为新兴的产业。软件工程是采用工程的概念、技术和方法来开发和维护软件，简言之，软件工程是计算机软件开发和维护的工程科学。计算机工作是用程序来控制的。程序规定了计算机完成任务的步骤。程序设计的任务就是用程序设计语言编写程序，然后由计算机来执行，去完成某一既定的任务。

1. 程序设计的步骤

程序设计是一项充满智慧和创造力的工作，也是极其艰苦、复杂和富有吸引力的工作。如何进行程序设计，是进行程序设计时必然要面对的问题。概括起来，要经过以下几步。

第1步 进行系统分析：这是由系统分析员和用户共同完成的工作。主要任务是深刻了解用户的需求，确定解决问题和完成任务的方案，研究出一个逻辑模型。

第2步 确定算法：算法是用计算机解决问题的步骤。算法首先要正确，同时要考虑时间和空间的复杂度，要以最少的时空代价来解决问题和完成任务。在这一步要设计出一个系统模型。

第3步 编写程序：用程序设计语言把算法描述出来，并输入到计算机。

第4步 运行与调试：在计算机上对程序进行编辑、编译、连接和运行，发现错误，进行修改。此过程反复进行，直到得到运行结果——说明程序已无语法错误和语义错误。

为了检查程序有无逻辑错误，还要对程序进行测试。所谓逻辑错误，是指算法本身的错误，或指程序未能真正体现算法。测试的任务就是发现软件中的错误。任何一个软件产品交付使用前，都要经过严格的测试。

第5步 完成文档资料的整理工作：包括使用说明书。用户根据程序的提示，输入数据(信息)，程序就能操纵计算机完成既定的任务。

2. 流程图

在程序的设计过程中，描述算法有许多方法，最常见的是流程图。这种描述算法的工具由几种基本的框图组成，如图1.3-1所示。图中判断框(菱形)有一个入口二个出口，根据给定的条件，判断执行何种操作。连接点(圆圈)用于将不同处的流程线连接起来，以免流程线的交叉或过长。

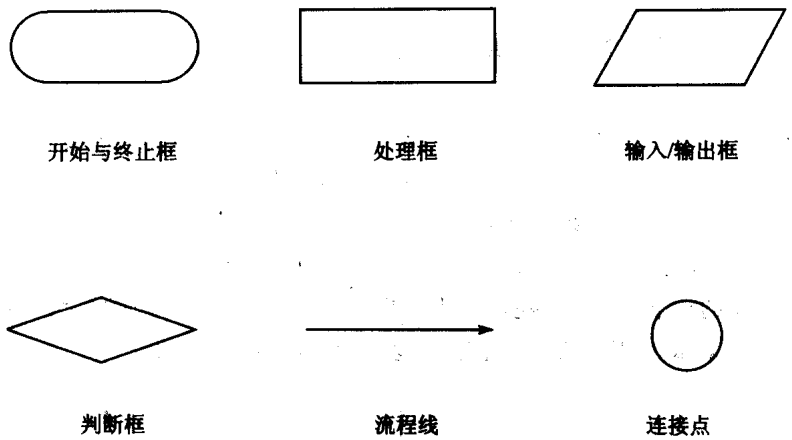


图 1.3-1 流程图的基本框

3. 程序结构

对于程序设计，目前使用最广的方法有两种：结构化程序设计方法和面向对象的程序设计方法。C语言是结构化程序设计语言，具有完善的程序控制结构与模块化程序设计手段；而C++语言是面向对象的程序设计方法的代表性语言。下面将分别加以介绍。

1.3.2 结构化程序设计

结构化程序设计是20世纪60年代末被提出来的，到70年代末已得到很大的发展。结构化程序设计主要采用自上而下、逐步细化的设计方法。

结构化程序有3种基本结构：顺序结构、选择结构和循环结构。

1. 顺序结构

顺序结构是自顶向下、顺序地执行程序中每一条语句，如图1.3-2所示。

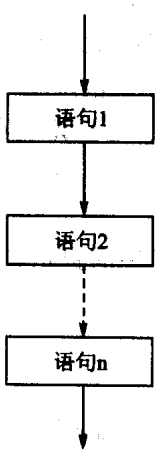


图 1.3-2 顺序结构流程图

2. 选择结构

选择结构是当判断表达式(判断条件)为真时，执行语句1；否则执行语句2，如图1.3-3所示。

3. 循环结构

循环结构分为当型循环和直达型循环两种。当型循环结构的特点是，当判断条件表达式为真时，执行循环体；否则退出循环，如图1.3-4(a)所示。直达型循环结构的特点是，先执行循环体，直到判断条件表达式为假时，退出循环，如图1.3-4(b)所示。

由上述3种基本结构构成的程序被称为结构化程序，换句话说，任何一个结构化程序都可以分解为一个个基本结构。程

序结构按功能划分为基本模块，每个模块是一个相对独立的程序段（子模块）。模块间的相对独立性，使每个模块可独立地进行分析、设计、编码、调试和修改。每一模块也都是由顺序、选择和循环3种基本结构组成的。C语言中各模块是用函数来实现的；有些程序设计语言是用子程序来实现的。

结构化程序设计方法可以看做是处理一系列问题的过程。它强调的是如何做。程序是一些过程或子过程的集合，这些过程通过参数传递数据。

结构化程序设计方法虽然是处理复杂问题的一种有效方法，但是，随着这种方法的被广泛应用，其存在的问题也充分暴露出来。最

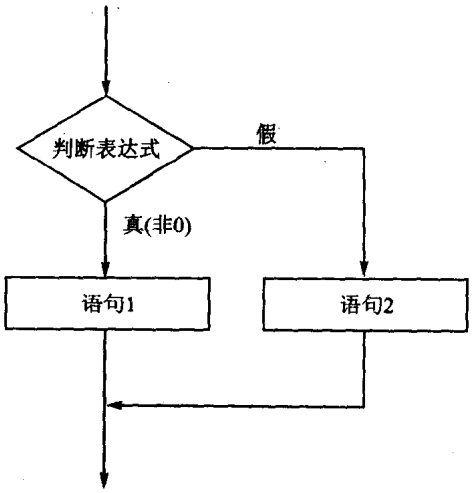
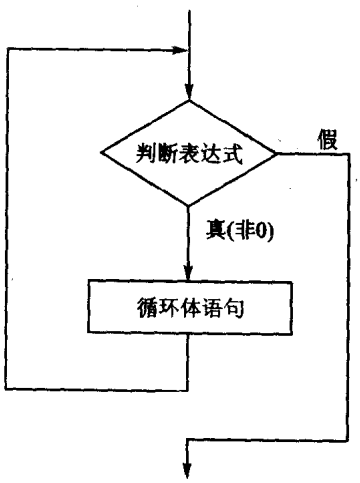
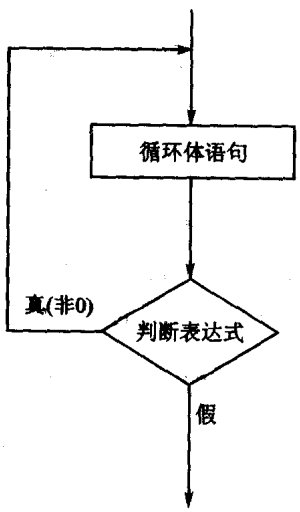


图 1.3-3 选择结构流程图



(a) 当型循环结构



(b)直达型循环结构

图 1.3-4 循环结构流程图

为要害的问题是，数据与处理数据分离，使程序难以理解和维护，软件的重用性差。随着软件开发技术的进步，出现了图形用户接口（GUI）和事件驱动等新技术和新概念，使程序明显变大了，用结构化方法已难以描述和维护。20世纪80年代出现了面向对象的设计方法。

1.3.3 面向对象的程序设计

面向对象的程序设计方法简称OOP（Object-Oriented Programming）方法。为使程序设计能更直接地描述客观世界中存在的事物及其相互关系，克服程序的复杂性问题，面向对象的程序设计方法不再将问题分解为过程，而是将问题分解为对象（事物）。对象将自己的属性（数据）和方法（数据的操作）封装成一个整体。对象间的相互作用通过消息传送来实现。它吸收