C及C++程序设计

钮 焱 许新民 严运国 主编

科学出版社
www.sciencep.com

C及C++程序设计

钮 焱 许新民 严运国 主编

科学出版社
北京

05
02

内 容 简 介

本书是C及C++程序设计课程教材，内容包括：程序设计概论，C语言的数据类型、运算符与表达式，顺序结构程序设计，流程控制语句，数组，函数，指针，结构体、共用体和枚举类型，文件，综合运用，C++与面向对象的程序设计。为便于教学和自学，本书配有《C及C++程序设计实验教程》。

本书特色明显，特别适合于计算机语言的初学者，适用于大、中专院校非计算机专业本、专科学生，也可供高等职业技术学院学生、成人与职业学校学生、等级考试考生、计算机培训班学员及计算机C语言自学者、爱好者使用。

图书在版编目(CIP)数据

C及C++程序设计/钮焱等主编. —北京：科学出版社，2003.1

ISBN 7-03-011045-5

Ⅰ.C… Ⅱ.钮… Ⅲ.C语言-程序设计-教材 Ⅳ.TP312

中国版本图书馆CIP数据核字（2002）第101096号

责任编辑：冯贵层/责任校对：王望荣

责任印制：高 嵘/封面设计：深白广告

科学出版社出版

北京东黄城根北街16号

邮政编码：100717

http://www.sciencep.com

武汉大学出版社印刷总厂印刷

科学出版社出版 各地新华书店经销

*

2003年1月第 一 版 开本：787×1092 1/16

2003年1月第一次印刷 印张：14 1/2

印数：1—10 000 字数：333 000

定价：19.60元

（如有印装质量问题，我社负责调换）

《C及C++程序设计》编委会

主　编　钮　焱　许新民　严运国

编　委　（按姓氏笔画为序）

方风波　孙晓莉　许新民　李　军　吴更生

严运国　李振立　张　群　钮　焱　高明成

袁　圆　徐振平　曾凡军　熊　英　蔡明文

前　言

进入21世纪，以计算机和网络技术为代表的IT产业获得了突飞猛进的发展，由互联网构筑的信息高速公路加速了全球数字化、信息化的进程，把人类社会推进到信息时代。

在信息时代，为人们广泛接受的计算机语言种类达到数十种，常见的入门语言也有Basic，FoxPro等多种，经过我们多年的教学实践摸索，感受到C语言应成为首选的计算机入门语言，理由为：

(1) C语言严谨、结构化，程序书写效率高，可学习到顺序、选择、循环等程序设计的基本思想。

(2) C语言可学以致用，现实环境中，程序设计者可运用它编写应用系统。

(3) 同其他语言相比，C语言概念多、规则多、使用灵活、容易出错，对初学者有一定的难度，但学生掌握C语言、理解C++有关概念以后，再学习Java，Visual Basic，PowerBuilder，Delphi等其他实用语言时，将感觉非常轻松，同时，也为进一步学习Visual C++，C++ Builder打下了基础。

我们根据当前高等学校学生的特点，结合计算机技术的发展趋势，在多年、多所学校的非计算机专业C语言教学经验基础上，以最常用的微机Turbo C为框架，编写了本书。全书共分十一章，第一章介绍程序设计概论，第二章介绍C语言的数据类型、运算符与表达式，第三章介绍顺序结构程序设计，第四章介绍流程控制语句，第五章介绍数组，第六章介绍函数，第七章介绍指针，第八章介绍结构体、共用体和枚举类型，第九章介绍文件，第十章针对近年来全国计算机等级考试，介绍了典型的选择题、填空题和改错题，着眼于综合运用，第十一章介绍C++与面向对象的程序设计基本思想。为便于教学和自学，本书配有《C及C++程序设计实验教程》。

本书特色明显，特别适合于计算机语言的初学者，适用于大、中专院校非计算机专业本、专科学生，也可供高等职业技术学院学生、成人与职业学校学生、等级考试考生、计算机培训班学员及计算机C语言自学者、爱好者使用。

本书由钮焱、许新民、严运国主编，方风波、孙晓莉、李军、吴更生、李振立、张群、高明成、袁圆、徐振平、曾凡军、熊英、蔡明文为编委，钮焱、许新民、严运国统稿，最后由钮焱定稿。

由于编者的编写时间仓促，水平有限，缺点和疏漏在所难免，敬请读者不吝批评指正。

编者

2002年10月

此为试读，需要完整PDF请访问：www.ertongbook.com

目　录

第一章　程序设计概论

语言是信息的载体，是人类交换信息的中间媒介。语言是由语句组成，语句是由单词组成，语言的词法、语法、语义、语用是研究自然语言的重要因素。计算机语言是人与计算机之间交流信息的工具，是由计算机能够识别的语句组成，它使用一整套带有严格规定的符号体系来描述计算机语言的词法、语法、语义、语用。词法负责从构成源程序的字符串中识别出一个个具有独立意义的最小语法单位(单词)；语法涉及语言的构成规律，确定程序的结构形式；语义说明语句代表的含义及该语句的执行过程；语用指出语句的实际用途。

1.1　程序设计基本技术

1.1.1　程序设计语言

从广义上理解，程序是指处理事情的先后次序。在计算机语言中，程序定义为完成特定任务的计算机指令的集合。用户编写的程序称为源程序，源程序通过翻译形成目标程序(.OBJ文件)，将目标程序与函数库连接后，形成运行程序(.EXE文件)。程序设计指用户通过编写源程序，翻译源程序为目标程序，连接目标程序与函数库，形成运行文件并整理设计文档的全过程。

计算机指令是指挥计算机完成某一操作的命令，它由操作码和操作数组成。计算机指令的执行过程可分为4个基本操作：取指令、分析指令、执行指令、停止操作。

(1) 取指令：从存储器中取出要执行的指令，送往CPU内部的指令寄存器暂存。

(2) 分析指令：将指令送往指令译码器，译出指令对应的微操作。

(3) 执行指令：根据指令译码器向各部件发出的控制信号，完成指令规定的操作。

(4) 停止操作：检查是否有复位信号、中断请求、DMA传送等，为执行下一指令作准备。

程序设计语言是计算机能够接受、能够理解的全部指令的集合。随着计算机技术的发展，程序设计语言也在不断向前发展，它先后经历了机器语言、汇编语言、高级语言、专用语言(指非过程化的高级语言)和面向对象的程序设计语言等几个阶段。

1. 机器语言

机器指令的操作码和操作数都是由“0”、“1”二进制代码组成的序列，表示给计算机的控制电路加上“低电平”、“高电平”，指挥计算机完成某一特定的操作，所以计算机能识别机器指令。每条机器指令代表计算机可执行的一个基本操作，不同的计算机的机器指令系统不相同，其对应的操作也不相同，这种形式的指令系统是面向机器的。

机器语言是机器指令的集合，用机器语言编写的程序计算机能够直接执行，执行速度快，效率高。它的缺点是可读性差，难于编程，容易出错，程序的检查和调试都很困难。由于不同类型的计算机使用不同的机器语言，造成使用机器语言编写的程序的通用性和可移植性差，只有熟练的计算机专业人员才能掌握。

此为试读，需要完整PDF请访问：www.ertongbook.com

2. 汇编语言

为了克服机器语言读写的困难，人们用助记符描述机器指令，称为汇编指令。汇编指令与机器指令之间存在一一对应的关系。计算机不能直接执行汇编指令，需要用专门的汇编程序翻译成机器指令后才能在计算机上执行。汇编程序是由软件开发商提供的将汇编指令翻译成机器指令的程序。

汇编语言是一种用助记符表示的程序设计语言。用户编写汇编语言源程序，由汇编程序对源程序进行汇编，翻译成目标程序，其过程如图1.1所示。

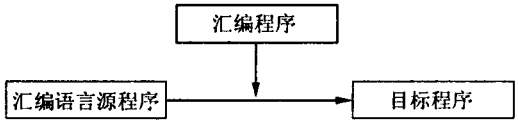


图1.1 汇编过程

机器语言和汇编语言都属于低级语言，都是面向机器的语言，但汇编语言比机器语言更易于理解和记忆。机器指令难于阅读、难于理解、难于记忆、难于编程。例如，将寄存器A中的数据送到寄存器B的机器指令为：1000100111000011(用十六进制表示为89C3)，而用汇编指令表示为：MOV BX,AX，显然比机器指令便于理解和记忆。

3. 高级语言

高级语言是更接近于自然语言或数学语言的程序设计语言。它彻底摆脱了依赖于机器硬件的指令系统，使用自然语言中的英语单词和数字符号编写程序，有专门的词汇和语法规则，是面向应用的计算机语言。

高级语言不再依赖于具体的计算机，用高级语言编制的程序可以在不同的计算机上使用，程序具有可移植性。编程时用户不再考虑计算机的内部结构和硬件环境，可以集中精力考虑解题的算法和数据结构，编写高质量的程序。高级语言又称为算法语言，常用的高级语言有Basic、Fortran、Pascal、C、Foxbase等。

高级语言不能直接在计算机上执行，计算机只能识别机器语言程序。高级语言编写的源程序必须被翻译成二进制代码的程序，计算机才能执行。翻译的方式有两种，即编译和解释。

编译方式由编译程序一次性对源程序翻译并优化，形成机器指令表示的目标程序，通过与函数库连接形成运行程序，计算机执行运行程序并得到运行结果。编译程序是由软件开发商提供的执行翻译工作的程序，编译方式的工作过程如图1.2所示。

解释程序是厂家生产的对源程序逐句翻译并逐句执行的语言处理程序。解释方式指解释程序对源程序逐句翻译并逐句执行的过程。与编译方式不同，解释方式不是一次将源程序翻译成目标程序后再执行，而是边翻译边执行，解释方式的工作过程如图1.3所示。

4. 专用语言

指为解决特定的应用而设计的非过程化的程序设计语言。例如，计算机辅助设计(CAD)系统中的绘图语言，数据库管理系统(DBMS)的数据查询语言等等。使用这类语言编程不需要描述为解决某一问题所需的具体实现过程，只需指出“做什么”即可。

5. 面向对象的程序设计语言

对象指人们进行研究的事情和物体，如计算机、显示器、打印机等。每个对象都有各自的属

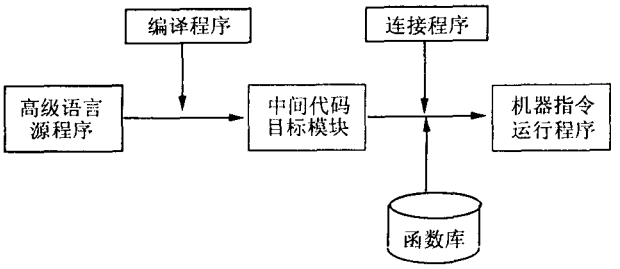


图1.2　编译方式工作过程

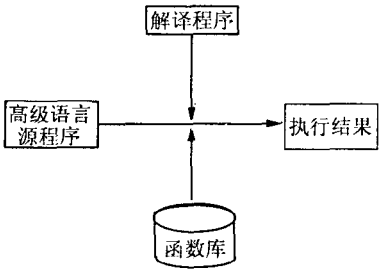


图1.3　解释方式工作过程

性、方法、集合或事件：属性用来描述对象的特性；方法用来描述执行对象的动作；集合指多个对象构成的整体值；事件指用户作用于对象的某一操作，如〈单击〉鼠标器的Click 事件。面向对象的程序设计是将对象的属性值(数据)和作用在这些数据上的方法结合在一起的程序设计方法，对象包括数据和操作这些数据的方法，由事件驱动操作这些对象。在面向对象的程序设计语言中用class(类)数据类型组织对象，一个class(类)提供一组值和作用于这些值的操作。类是从实体对象中抽象出来的模型，由类描述形成的对象是类的实例。类具有封装性、继承性和多态性。

1.1.2　程序设计的基本过程

程序设计是人们使用计算机语言编制解决实际问题的源程序，经过计算机编辑、翻译和处理，执行该程序并分析输出结果，通过反复调试直到结果完全正确为止，最后整理设计文档的全过程。较小规模的程序设计由一个程序员完成，大型程序设计是由多个程序员分工，共同协作完成的，因此必须经过多种测试，并详细整理设计文档。程序设计的基本过程如图1.4 所示。

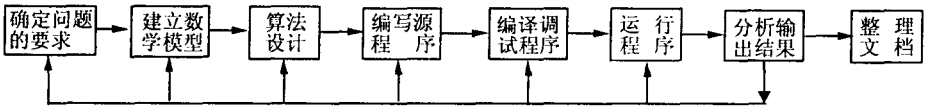


图1.4　程序设计的基本过程框图

1. 确定问题要求

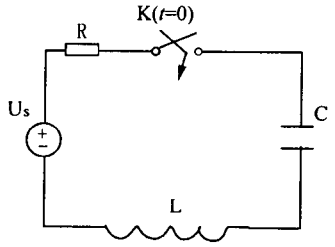
程序设计的第一步是对实际问题进行分析，明确问题的已知条件，从中获得必要的输入数

据，明确问题要求做什么，需要什么样的数据输出。

分析问题的要求、弄清问题的性质、发现问题的特点、确定解决问题的目标能够使我们采取有效的方法解决问题。

2. 建立数学模型

要用计算机解决实际问题，首先要用理想模型来模拟实际问题。理想模型是从实体中抽象出来并能用数学表达式精确定义的实体。理想模型用理想化的元件表示实体，称为物理模型。例如，电路中的电阻、电感、电容、电压源、电流源等都是理想化的元件，理想元件组成的电路为物理模型。图1.5所示的物理模型是由电阻、电感、电容和电压源串联的电路，该电路可以用微分方程描述：



$$LC\frac{d^2u_c}{dt^2}+RC\frac{du_c}{dt}+u_c=u_s$$

$$u_c(0)=0$$

$$\left.\frac{du_c}{dt}\right|_{t=0}=0$$

图1.5 *RLC* 串联电路的物理模型和数学模型

我们把描述电阻、电感、电容和电压源串联电路的微分方程称为数学模型。这种把错综复杂的实际问题进行简化抽象，用数学公式来描述实际问题的运动和变化过程，称为建立数学模型的过程。数学模型是反映事物本质的数学表达式，实际问题的精度要求不同，选择的数学模型也就不同。一般元器件都具有一定的非线性，只能在一定精度范围内用理想元件近似，如果精度要求更高的话，就必须使用非线性元件来模拟实际元件，这样会增加分析问题的难度。

对微分方程计算机不能直接求解，必须使用数值分析方法（计算方法）将微分变成差分，才能用计算机求解。

在建立数学模型的过程中，分析问题的要求、确定解决问题的目标、明确问题的输入数据和输出信息、理解问题的约束限制条件，是选择制定数学模型、建立数学模型的关键步骤，是解决问题的关键所在。

3. 算法设计

建立好数学模型，选择好数值分析方法，就可以逐步设计解题的步骤，即确定算法。注意不要将计算方法与算法混淆起来，两者的区别在于：计算方法是指求解复杂函数的数值分析方法；算法是指为解决某一特定问题而进行一步一步有穷的操作过程，是一组规则的集合，可以用传统流程图、结构化流程图等描述。编写程序代码之前，设计好算法，画出流程图，往往会起到事半功倍的效果。技术水平高的程序员应积累一些经典算法，如求和求阶乘算法、素数算法、排序算法等。对于小型程序设计，可直接使用经典算法的组合完成整个程序的算法设计。

4. 编写源程序

源程序是用计算机语言描述的算法，编写源程序的过程是用计算机语言替换流程图的描述方式，将算法描述工具表示的算法转换成计算机语言来描述。

5. 编译调试程序

使用软件厂商生产的编译程序调试源程序，修改程序中的错误，通过反复编译调试，直到没有语法错误为止。

6. 运行程序

将编译源程序形成的目标文件与函数库连接，形成运行文件，运行该程序，检查程序的输出结果是否正确，如发现错误，检查程序的逻辑是否正确，主要解决算法设计错误。

7. 分析输出结果

对照问题要求，分析输出结果，检查是否达到问题要求的功能和精度要求，输出信息的格式是否符合要求。输入不同的数据，检查是否还存在逻辑错误。出错后，分清是算法错误还是数学模型选择错误，修改源程序后再从第5步开始执行。

8. 整理文档

当程序准确无误后，要认真地整理文档，包括设计要求、设计思路、设计过程、使用的算法、数据结构、输出信息及格式，在源程序中要用注释语句加上必要的说明。有很多程序员不注意整理文档，编制的程序过一段时间后自己也看不懂，更不要说给别人看，对此要引以为戒，要学会从学习程序设计开始就养成良好的习惯。

1.1.3 算法及算法描述

算法是为解决某一特定问题而进行一步一步操作过程的精确描述，是有限步、可执行、有确定结果的操作序列。

1. 算法的特征

算法具有如下特征：

(1) 有穷性：每一个算法分成有限个操作步骤，每一步在有限的时间内完成。

(2) 确定性：算法的每一步都有明确的含义，不含歧义，每一步命令只能产生惟一的一组动作。

(3) 有效性：算法的每一步都能有效地执行，并输出确定的结果，只要有一个不可执行的操作，该算法即无效。

(4) 有零个或多个输入：可以没有输入，也可以有一个或多个输入。

(5) 有一个或多个输出：至少有一个输出，没有输出的算法是没有意义的。

2. 三种基本结构

编制程序时，对任何复杂的问题都可以用三种基本结构来描述其算法，基本结构之间形成顺序执行的关系。语句块是语句的集合，是满足这三种结构的基本单元。三种基本结构为：顺序结构、分支选择结构、循环结构。

(1) 顺序结构：是最基本、最简单的程序结构，在此结构内各语句块是按照它们出现的先后次序依次执行的。

(2) 分支选择结构：首先根据给定的条件判断条件是否为真，条件为真，选择满足条件的路径(语句块)执行，条件为假，选择不满足条件的路径(语句块)执行。

(3) 循环结构：是一种重复处理的程序结构，当满足(或不满足)某个指定的条件时反复执行语句块(循环体)，否则跳出循环，执行循环体外的下一条语句。

使用这三种基本结构编制程序，不同基本结构之间的关系简单清晰，容易阅读、理解、维护、修改。

3. 算法的描述方法

算法是解题方法和解题过程的精确描述。描述算法的工具很多，常用的有自然语言、传统流程图、N-S 流程图、伪代码等多种，它们各有优缺点。

(1) 自然语言：指对人们日常使用的语言加以简化来描述算法。自然语言通俗易懂，人们易于接受，但自然语言描述冗长、不严谨，有时有很大的随意性和歧义性。

例1.1 用自然语言描述求和算法，求1+2+3+4+5+6+7+8+9+10。

用S1 表示第1 步，S2 表示第2 步，依此类推；用sum 存储累加和，在循环体外赋初值为0，循环变量为n，初值设为1；重复执行10 次操作“sum+n⇒sum；n+1⇒n”，结果在sum 所得到的值是1 到10 的累加和。具体算法如下：

S1：0⇒sum

S2：1⇒n

S3：如果n ≤ 10，执行以下操作，否则转到S4

S3.1：sum+n ⇒ sum

S3.2：n+1 ⇒ n

S3.3：转到S3

S4：输出sum 的值

S5：结束

(2) 传统流程图：是用不同几何形状的线框、流线和文字说明来描述算法。传统流程图的常用符号及其意义如表1.1 所示。

表1.1 传统流程图常用的符号及意义

符号	符号名称	意　义	实例
	起止框	算法的开始或结束	开始　结束
	处理框	算法的一个处理步骤	1⇒n
	调用框	调用模块或函数	abc(x,n)
	流程线	算法执行的流程方向	
	判断框	选择控制，根据框中的条件确定从哪条分支执行	N　n<=10　Y
	输入输出框	数据的输入，结果的输出	输入
	连接点	将算法中画在不同处的流程连续起来	A　A
	注释框	用于书写算法中的注释内容	FOR循环

用传统流程图表示三种基本结构如表1.3 所示。

(3) N-S 流程图：又称结构化流程图，是在美国I. Nassi 和B. Schneiderman 两学者提出的描述方法基础上形成的结构化流程图。由于是以他们的名字的缩写命名，故称N-S 流程图。N-

S流程图由一系列矩形框顺序排列而成，各个矩形框只能顺序执行，每一个矩形框表示一个基本结构。矩形框内的分割线将矩形框分割成不同的部分，形成三种基本结构：顺序结构、分支结构、循环结构。表1.2所示为N-S流程图的基本结构框。

表1.2 N-S流程图的基本结构框

结构名称	矩形框的分割	基本结构框
顺序结构		语句块A 语句块B
选择结构		是 条件 否 语句块A　语句块B
循环结构 当型循环		当条件满足 语句块
直到型循环		语句块 直到条件满足

例 1.2 分别用传统流程图和N-S流程图画出求1+2+3+4+5+6+7+8+9+10的算法。

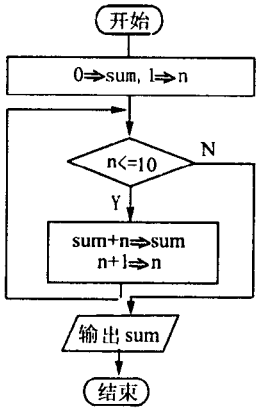


图1.6 用传统流程图描述的算法

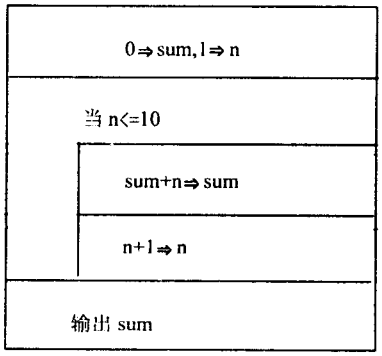


图1.7 用N-S流程图描述的算法

(4) 伪代码：是一种过程设计语言(PDL)，它没有严格的语法规则，可以把算法写得很具体，也可以把算法写得很概括，使用灵活，形式接近高级语言，与PASCAL语言很类似，在算法设计中经常使用。

例1.3 用伪代码描述求1+2+3+4+5+6+7+8+9+10的算法。

为sum、n赋初值：

DO WHILE (n≤10)

　sum=sum+n

此为试读，需要完整PDF请访问：www.ertongbook.com