

21世纪高等学校规划教材

C 语言程序设计

马建生 谭 征 李 玲 编著



中国电力出版社

www.infopower.com.cn

TP312/2558

21 世纪高等学校规

2007

C 语言程序设计

马建生 谭 征 李 玲 编著



中国电力出版社

www.infopower.com.cn

内容提要

作者结合自己多年的教学经验,从一个初学者的角度出发,以由浅入深、循序渐进的原则编写了本书。全书共分12章,前两章详细介绍了C语言的基础知识和基本语法规则,第3~5章详细介绍了C语言的基本输入输出函数及顺序、选择、循环三种程序设计结构,第6章介绍了函数的定义、调用以及全局变量与局部变量的使用等,第7章介绍了宏定义、文件包含和条件编译的使用,第8章介绍了一维数组、二维数组的定义和使用以及常用字符串处理函数,第9章介绍了指针的定义和指针在程序中的重要地位和广泛应用,第10章介绍了结构体、共用体和枚举类型的定义和使用、链表的基本操作等,第11章介绍了位运算的使用,第12章介绍了缓冲区系统文件的常用处理方法。每章章末还配有精选的习题,以帮助读者更好地掌握所学知识。

本书结构清晰、层次分明,既可作为非计算机专业学生学习C语言的教材,也可以作为全国计算机等级考试的参考教材,同时也很适合广大计算机软件爱好者用以迅速、熟练地掌握C语言。

图书在版编目(CIP)数据

C语言程序设计 / 马建生, 谭征, 李玲编著. —北京: 中国电力出版社, 2007

21世纪高等学校规划教材

ISBN 978-7-5083-5683-9

I. C… II. ①马…②谭…③李… 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(2007)第106591号

丛 书 名: 21世纪高等学校规划教材

书 名: C语言程序设计

出版发行: 中国电力出版社

地 址: 北京市三里河路6号

邮政编码: 100044

电 话: (010) 68362602

传 真: (010) 68316497, 88383619

服务电话: (010) 58383411

传 真: (010) 58383267

E-mail: infopower@cepp.com.cn

印 刷: 北京市同江印刷厂印刷

开本尺寸: 185mm×260mm 印 张: 13.5 字 数: 326千字

书 号: ISBN 978-7-5083-5683-9

版 次: 2007年8月北京第1版

印 次: 2007年8月第1次印刷

印 数: 0001—3000册

定 价: 21.00元

敬告读者

本书封面贴有防伪标签,加热后中心图案消失

本书如有印装质量问题,我社发行部负责退换

版 权 专 有 翻 印 必 究

前 言

C 语言是一种结构化、模块化、可编译的通用程序设计语言，具有表达能力强、代码质量高和可移植性好等特点，并兼备高级语言和低级语言的许多优点，已成为目前国际上广泛应用的现代主流程序设计语言，深受广大软件爱好者的喜爱。随着 C 语言应用范围的不断扩大，无论是计算机专业人员还是一般的软件爱好者，都将 C 语言作为学习软件编程的首选语言。

近年来，高等院校在计算机专业和非计算机专业中都开设了“C 语言程序设计”课程。另外，越来越多的非计算机专业的学生在参加全国计算机等级考试时也都选择了 C 语言程序设计模块。为了满足广大学生学习这门语言的需要，我们结合自己多年的教学经验，从一个初学者的角度出发，按由浅入深、循序渐进的顺序，用更为易懂的语言编写了这本教材。本书可作为非计算机专业学生学习 C 语言的教材，也可以选作全国计算机等级考试的参考教材，同时也很适合广大计算机软件爱好者用以迅速、熟练地掌握 C 语言。

本书共分为 12 章。前两章详细地介绍了 C 语言的基础知识、C 语言的发展和 C 语言的基本语法规则；第 3~5 章详细介绍了 C 语言基本输入输出函数及顺序、选择、循环三种程序设计结构；第 6 章介绍了函数的定义、调用以及全局变量与局部变量的使用等；第 7 章介绍了宏定义、文件包含和条件编译的使用；第 8 章介绍了一维数组、二维数组的定义和使用，以及常用字符串处理函数；第 9 章介绍了指针的定义和指针在程序中的重要地位和广泛应用；第 10 章介绍了结构体、共用体和枚举类型的定义和使用、链表的基本操作等；第 11 章介绍了位运算的使用；第 12 章介绍了缓冲区系统文件的常用处理方法。

本书结构清晰，层次分明，通俗易懂，通过大量例题来验证语法和说明程序的设计方法，并且为了使读者更好地掌握各章节内容，章末还配有大量精选的习题。通过本书的学习，读者既能迅速掌握 C 语言的基础知识，又能很快地学会 C 语言的编程技术，提高解决实际问题的能力。习题答案将在配套的教材中给出。

本书的第 1、3、4、5、6、9、10 章由谭征老师编写，第 2、7、8 章由李玲老师编写，第 11、12 章由马建生老师编写，并由马建生老师负责全书统稿。

在本书编写过程中，编者参考了大量有关 C 语言程序设计的书籍和资料，在此对这些参考文献的作者表示感谢。

限于编者的水平，书中难免存在疏漏之处，敬请有关专家和广大读者不吝指正，在此表示诚挚的谢意。

编 者
2007 年

目 录

前 言

第 1 章 概述	1
1.1 C 语言与程序设计	1
1.2 算法及其表示	2
1.3 C 程序风格及书写规则	3
1.4 结构化程序设计方法	4
习题	6
第 2 章 C 语言基本数据类型及运算符	7
2.1 C 语言的基本构成成分	7
2.2 C 语言的基本数据类型	9
2.3 整型数据在内存中的存储形式	13
2.4 浮点型数据在内存中的存储形式	14
2.5 字符型数据在内存中的存储形式	15
2.6 运算符与表达式	15
习题	24
第 3 章 顺序结构程序设计	27
3.1 C 语句简介	27
3.2 字符数据的输入和输出	27
3.3 格式输入与输出函数	28
3.4 顺序结构程序设计举例	33
习题	34
第 4 章 选择结构的程序设计	38
4.1 if 语句	38
4.2 switch 语句	42
4.3 goto 语句及语句标号	43
习题	43
第 5 章 循环结构的程序设计	47
5.1 while 语句	47
5.2 do-while 语句	48
5.3 for 语句	49
5.4 循环嵌套	51
5.5 break 语句和 continue 语句	52
5.6 程序举例	54

习题	57
第 6 章 函数	62
6.1 函数的定义和返回值	62
6.2 函数的调用	64
6.3 函数的参数传递	66
6.4 函数的声明（说明）和函数的原型	68
6.5 函数的嵌套调用和递归调用	69
6.6 函数应用举例	72
6.7 局部变量和全局变量	74
6.8 变量的存储类别	76
6.9 函数的存储类别	81
习题	82
第 7 章 编译预处理	90
7.1 宏定义	90
7.2 文件包含	95
7.3 条件编译	98
习题	99
第 8 章 数组	105
8.1 一维数组	105
8.2 二维数组	109
8.3 字符数组与字符串	113
习题	121
第 9 章 指针	127
9.1 地址和指针的概念	127
9.2 指针的定义和使用	128
9.3 指针的运算	131
9.4 指针作函数的参数	132
9.5 返回值为指针的函数	134
9.6 数组和指针	134
9.7 字符串和指针	146
9.8 指向函数的指针变量	151
习题	153
第 10 章 结构体和共用体	159
10.1 结构体类型和结构体变量	159
10.2 结构体数组和结构体类型的指针	163
10.3 用结构体变量处理链表	167
10.4 共用体	172
10.5 用 typedef 定义数据类型名	174
习题	175

第 11 章 位运算	180
11.1 位运算符.....	180
11.2 位运算.....	180
习题.....	182
第 12 章 文件	185
12.1 C 文件的概念.....	185
12.2 C 文件的打开与关闭.....	187
12.3 文件关闭函数.....	188
12.4 文件的读写操作函数.....	189
12.5 文件定位.....	195
12.6 应用举例.....	197
习题.....	198
附录 1 ASCII 字符编码表	201
附录 2 C 运算符的优先级和结合方向	202
附录 3 常用的 C 库函数	203
参考文献	208

第 1 章 概 述

1.1 C 语言与程序设计

1.1.1 C 程序简介

C 语言是目前国际上最流行的程序设计语言之一。在程序设计的发展过程中,出现过各种计算机语言,最早的程序是用二进制代码编写的,称为“机器语言”,又称为“低级语言”。随着计算机语言的发展,又出现了汇编语言,汇编语言是用“助记符”代替二进制代码,所以又称为“符号语言”。机器语言和汇编语言都是“面向机器的语言”,都被称为“低级语言”。为了摆脱机器指令的束缚,人们又发明了“高级语言”。目前所使用的程序设计语言都被称为“高级语言”,这些高级语言更接近于人类的自然语言和数学语言,易于学习和操作,C 语言就是其中的一种。

用高级语言编写的程序叫源程序。源程序不能在计算机上直接运行,因为计算机只能接受 0 和 1 组成的二进制程序(又称目标程序),高级语言源程序必须通过系统软件将其翻译成二进制程序后才能被执行。翻译程序有两种执行方式:一种是通过“解释程序”将源程序翻译一句执行一句,这种执行方式称为“解释执行”方式;另一种是通过“编译程序”将源程序全部翻译成二进制程序后再执行,此种执行方式称为“编译执行”方式。大多数高级语言采用“编译执行”方式,C 语言也采用编译方式。从源程序到目标程序,其操作过程如图 1-1 所示。

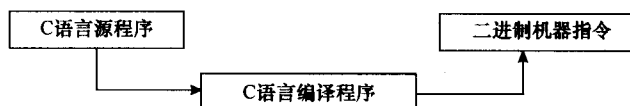


图 1-1 C 程序编译过程

由 C 语句构成的指令序列称为 C 源程序。C 源程序(.c 或.cpp)编辑好后,经过 C 语言编译程序编译,系统会自动生成二进制程序(.obj),称为“目标文件”。将目标文件和其他文件以及系统所提供的库函数通过“连接程序(link)”进行连接,生成可执行文件(.exe)才能执行。

1.1.2 程序与程序设计

程序是使用计算机语言对解决某个问题的方法和步骤进行描述。人们将计算机要解决的问题写成一定形式的指令,存储在计算机内部,由计算机连续执行。我们把这种可连续执行的指令集合叫作“程序”。程序设计就是在某一程序语言环境下,编写出能够使计算机理解并执行的程序代码。

简单的程序设计一般由以下几部分组成：

(1) 根据用户需求确定数据结构。数据结构包括数据的组织形式、存储形式和操作。根据需求任务书的要求，指定输入的数据和输出结果，并确定数据结构。

(2) 算法设计。算法是解决问题的方法和步骤。进行程序设计时，必须根据确定的数据结构选择或设计相应的算法。

(3) 编写程序代码。根据确定的数据结构和算法，选择一种程序设计语言和环境编写出程序代码，保存到计算机上。这一步骤又称为编程。

(4) 程序调试。又称软件测试，主要目的是发现软件中的错误并加以修正。程序调试可以人工进行，也可以利用辅助工具进行。

(5) 整理并编写软件设计时所需的各种文档。

1.2 算 法 及 其 表 示

1.2.1 算法的概念和特征

算法是指为了解决某个特定问题而采用的确定且有效的步骤。计算机算法可分为两大类：数值运算和非数值运算。数值运算的目的是求数值解，例如求方程的根、求圆的面积、求 n 的阶乘等，都属于数值运算。非数值运算包括的面十分广泛，主要用于事务管理，例如人事管理、图书管理、学籍管理等。算法有以下五个特性：

(1) 有穷性。一个算法应当包含有限个操作步骤；也就是说，在执行若干个操作之后，算法将结束，而且每一步都在合理的时间内完成。

(2) 确定性。算法中的每一条指令必须有确切的含义，不能有二义性，对于相同的输入必须能得出相同的结果。

(3) 可行性。算法中的每一步都应当有效执行，通过基本运算后能够实现目标。

(4) 有零个或多个输入。在计算机上实现算法，所需的处理数据要在程序执行时通过输入得到。也有的程序不需要输入数据。

(5) 有一个或多个输出。算法的目的是求解（结果），结果要通过输出得到。

1.2.2 算法的表示

算法可以用各种描述方法进行描述，常用的有自然语言、伪代码、传统流程图和 N-S 流程图。传统流程图由几种基本框图组成，如图 1-2 所示。



图 1-2 传统流程图的基本符号

传统流程图用流程线指明各个框的执行顺序。对流程线没有严格的限制，因此使用者可以随心所欲地使用流程线；过多地使用流程线，流程图会显得杂乱无章。虽然用传统流程图表示算法直观形象，能比较清楚地表示各个框之间的逻辑关系，但是，传统流程图的描述要占用大量的篇幅，尤其是面对比较复杂的程序时，大量的流程线如同乱麻，阅读费

时费力。

1973年,美国学者 I.Nassi 和 B.Shneiderman 提出了一种新的流程图形式。在这种流程图中,完全取消了带箭头的流程线,全部算法写在一个矩形框内。这种流程图被称为 N-S 流程图(用两位学者名字的第一个字母命名),它比较适合于结构化的程序设计,因而很受欢迎。

1.3 C 程序风格及书写规则

1.3.1 C 程序的构成

C 程序是由一个主函数(main()函数)和若干个(或 0 个)用户函数组成的。主函数和用户函数的位置是任意的,但它们的调用关系是一定的;即主函数可以调用任何用户函数,用户函数间可以互相调用,但不能调用主函数。用户函数甚至可以调用自己,这种调用称为递归调用。

C 程序总是从 main()函数开始执行,而不论 main()函数在整个程序中的位置如何。程序从主函数的第一条语句开始执行,直到主函数的最后一条语句结束。

一个函数由两部分组成,即函数说明部分和函数体部分。

(1) 函数说明部分(又称函数头)。函数说明部分包括函数值类型、函数存储类型、函数标识符、函数的参数和参数的类型。其格式是函数名后紧接着的一对圆括号,函数的参数写在圆括号里。

(2) 函数体。即用大括号括起来的部分。函数体内有若干条语句,它们能完成各种操作,具体函数的功能都写在函数体中。C 语言允许函数体内为空,如:

```
main()
{ int a,b,c;
  scanf("%d%d",&a,&b);
  c=add(a,b);
  printf("c=%d",c);
}
int add(int x,int y)
{ int z;
  z=x+y;
  return(z);
}
```

1.3.2 C 程序的书写规则

C 程序书写格式随意,除了保留字外,任何地方都可以插入空格或回车换行符。

为了便于阅读程序,建议采用格式化的书写格式,采用缩格纵向对齐方式。可以在程序的任何一处插入“注释”。注释是以“/*”开始,到“*/”结束的任意字符序列。注释语句是非执行语句,不参加编译,也不会出现在目标文件中,只起到帮助阅读程序的作用,如同读文章加上注释一样。

1.4 结构化程序设计方法

1.4.1 结构化程序设计的三种基本结构

一个结构化的程序就是用高级语言表示的结构化算法。一个结构化的程序由三种基本结构构成，这种结构便于阅读、修改和维护。

(1) 顺序结构。顺序结构是最常用的一种基本结构。它的特点是：由顺序结构组织的程序语句按它们书写的先后顺序依次执行。没有跳转，也没有分支。C 语言中的输入、输出、赋值等操作都可以按照顺序结构构成。顺序结构可以用如图 1-3 中的两种流程图来描述。

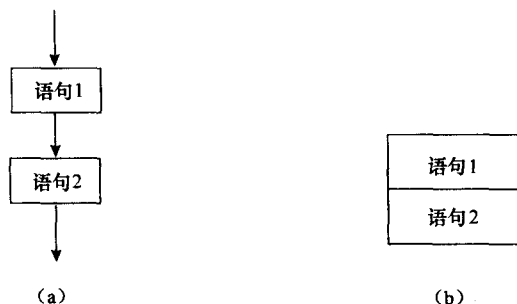


图 1-3 顺序结构

(a) 传统流程图描述的顺序结构；(b) N-S 流程图描述的顺序结构

(2) 选择结构又称分支结构。如图 1-4 所示，此结构中一定会包含一个判断框，系统根据给定的条件是否成立来选择执行语句序列 1 或语句序列 2 的内容。

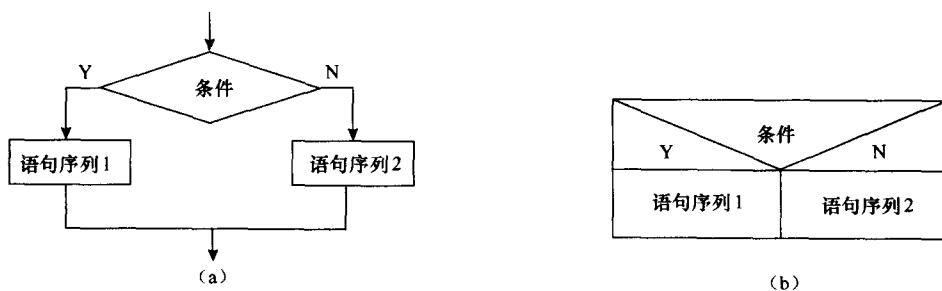


图 1-4 分支结构

(a) 传统流程图；(b) N-S 流程图

(3) 循环结构又称为重复结构，即重复执行某些操作。循环结构分为两种：一种是当型循环；一种是直到型循环。图 1-5 所示为当型循环。当型循环的特点是：当条件成立时，执行循环体；否则就不执行循环体。图 1-6 所示为直到型循环。直到型循环的执行特点是：执行循环体直到指定的条件满足，就不再执行循环体。对一个问题既可以用当型循环处理，也可以用直到型循环处理。

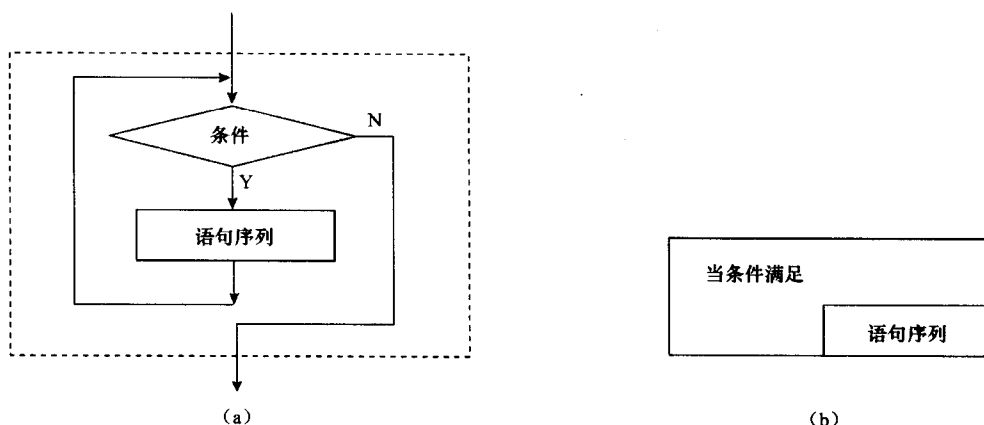


图 1-5 当型循环

(a) 传统流程图; (b) N-S 流程图

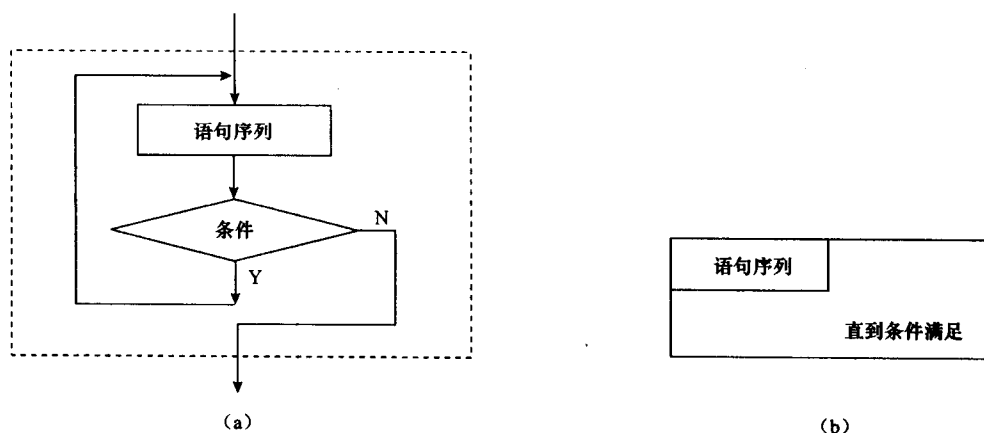


图 1-6 直到型循环

(a) 传统流程图; (b) N-S 流程图

以上三种基本结构有一些共同的特点:

- (1) 只有一个入口。
- (2) 只有一个出口。
- (3) 结构内的每一部分都有机会被执行到。
- (4) 结构内不出现无限循环。

已经证明, 以上三种基本结构组成的算法结构可以解决任何复杂的问题。

例如: 求 Fibonacci 数列的前 20 个数。这个数列有如下特点, 第 1、2 项均为 1, 从第 3 项开始, 每一项是前两项之和, 即:

$$f_1=1 \quad (n=1)$$

$$f_2=1 \quad (n=2)$$

$$f_n=f_{n-1}+f_{n-2} \quad (n \geq 3)$$

分析: 已知第 1 个数为 $f_1=1$, 第 2 个数为 $f_2=1$ 。其算法是: 通过 f_1 和 f_2 求出下一对数,

即新的 f_1 和 f_2 (迭代)。计算公式是: $f_1=f_1+f_2$; $f_2=f_2+f_1$ 。已给出第 1 对数, 只需再求其余 18 个即可。

根据此思路, 画出 N-S 流程图, 如图 1-7 所示。变量 i 用来控制循环的次数, 当 i 的值小于等于 10 时, 执行循环体。每次执行循环都会求出两个数据项。

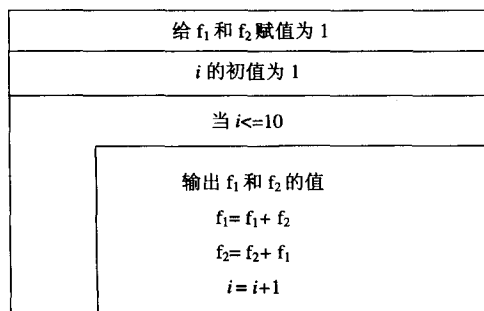


图 1-7 N-S 流程图

1.4.2 模块化程序设计

所谓“模块化结构”是一种程序设计的方法, 即在处理大型的复杂任务时, 我们可以将大的任务分成若干个子任务, 分配给不同的程序设计人员, 每个子任务又可以分成更小的任务, 这样每个人承担的任务量只是完成一个简单的功能。这样多人合作就可以完成一个大的任务。由许多个模块组成了一个结构化程序。用这种方法设计的程序结构称为“模块化结构”。

在设计程序结构时, 通常采用“自顶向下, 逐步细化”的方法, 软件设计人员在进行程序设计的时候, 首先应该先考虑主程序中的算法, 再逐步完成子程序的调用。

由若干个模块组成的程序靠主程序的调动分头执行各个子模块, 共同完成复杂任务。这个主程序又称“主控模块”。C 语言中没有子程序的概念, 它提供的函数可以完成子程序的全部功能。C 语言允许对函数进行单独的编译, 从而可以实现模块化; 另外还提供了丰富的数据类型。

C 语言处理问题的过程可概括为: “自顶向下, 逐步细化, 模块化设计, 结构化编码”。

习 题

1.1 C 语言源程序文件名的后缀是_____, 经过编译后, 生成文件的后缀是_____, 经过连接后, 生成文件的后缀是_____。

1.2 结构化程序由_____, _____、_____三种基本结构组成。

1.3 一个 C 程序是由若干个函数构成的, 其中必须有一个_____函数。一个 C 语句最少应包含一个_____。

1.4 程序设计语言的主要构成要素有哪些?

第 2 章 C 语言基本数据类型及运算符

在介绍基本的数据类型以前，先介绍 C 语言程序的基本构成成分。

2.1 C 语言的基本构成成分

2.1.1 字符集

任何一种高级语言都有自己的基本词汇和语法规则，程序代码都是由这些基本词汇符号按照语法规则编写的，C 语言也是这样，它规定了如下所需的基本字符集和标识符。

- (1) 英文字母：a~z，A~Z。
- (2) 阿拉伯数字：0~9。
- (3) 特殊符号（见本书第 2.6 节）。

2.1.2 标识符

C 语言的标识符主要用来表示常量、变量、函数和类型符的名字，起标识作用，它包括保留字、用户定义标识符和预定义标识符三类。

1. 保留字

在 C 语言中有特殊含义的英文单词称为“保留字”。作为保留字的每一个单词都有特定含义，不允许另作它用，具体分类如下。

- (1) 用于数据类型说明的保留字，如表 2-1 所示。

表 2-1 数据类型符

数据类型符	数据类型	数据类型符	数据类型
char	字符型	double	双精度型
int	整型	struct	结构型
short	短整型	union	共用型
long	长整型	typedef	类型定义型
signed int	带符号整型	enum	枚举型
unsigned int	无符号整型	void	空类型
float	浮点型	const	常量

- (2) 用于存储类型说明的保留字，如表 2-2 所示。

表 2-2 存储类型符

存储类型符	存储类型	存储类型符	存储类型
auto	自动	static	静态
register	寄存器	extern	外部

(3) 其他保留字，如表 2-3 所示。

表 2-3 其他保留字

保留字	中文含义	保留字	中文含义	保留字	中文含义
break	中止	else	否则	sizeof	计算字节数
case	情况	for	对于	switch	开关
continue	继续	goto	转向	volatile	可变的
default	缺省	if	如果	while	当
do	做	return	返回		

2. 预定义标识符

在 C 语言程序中，有的操作是在程序预处理时完成的，定义这种语句使用的保留字称为预处理标识符，如表 2-4 所示。

说明：用户可以使用这些标识符定义符号常量和宏等，但不要在程序中随意使用，以免造成混淆。

表 2-4 预定义标识符

保留字	中文含义	保留字	中文含义
define	宏定义	include	包含
undef	撤消定义	ifdef	如果定义
ifndef	如果未定义	endif	编译结束
line	行		

3. 用户定义标识符

用户定义标识符是程序员根据自己的需要而定义的一类标识符，用来标识变量、符号常量、用户定义函数、数组、类型名和文件指针等对象的名字。对于这种类型的标识符，C 语言有如下规定：

(1) 标识符由字母、数字和下划线三种字符组成，且第一个字符必须为字母或下划线。程序中使用的用户标识符除要遵循标识符命名规则外，还应注意“见名知义”。

(2) 用户选取的标识符不能是 C 语言预留的保留字。

(3) C 语言是区分大小写字母的。因此，sum、Sum 和 SUM 是不同的标识符。

例如，以下是一些合法的标识符：

sum	average	day	a2	_above
x_1_2_3	BASIC	yes	H	student_1

而下面是一些不合法的标识符：

a? (含有非法字符?)	c.g (含有非法字符小数点)
A\$123 (含有非法字符\$)	#33 (含有非法字符#)
123H (非字母或下划线开头)	a-0 (含有非法字符-)
printf (使用了系统保留字)	\n (使用了系统转义字符\n)

(4) C语言中标识符的长度(字符个数)无统一规定,随系统不同而异。有的规定不超过32个字符,有的规定不超过8个字符。通常,DOS系统规定字符长度为8个。如果标识符的长度超过了系统所规定的长度,规定范围内字符有效,而后面的字符则不被识别。

例如,有2个长度超过8个字符的文件标识符 student_one 和 student_two,系统会认为它们是相同的标识符。这是因为二者的前8个字符相同。为了将二者加以区别,可将它们改为 stud_1 和 stud_2 来加以区分。

2.2 C语言的基本数据类型

用C语言编写的用户应用程序,离不开对数据进行操作,而数据必须以某种特定的形式存在。数据类型是根据数据在程序运行期间所占用的空间和取值不同划分的。C语言提供一组基本的数据类型,如整型、实型、字符型等。同时,C语言还具有构造数据类型的能力,即它能将一些基本的数据类型加以组合,构造出更复杂的数据类型。在这一章里,我们将介绍一些基本的数据类型。

2.2.1 常量

常量是在程序运行过程中其值不能被改变的量。在C语言中,常量有不同的类型,如10、15、-20是整型常量;5.5、15.5、20.5是实型常量;'a'、'b'、'c'是字符型常量;"student"是字符串常量。

1. 整型常量

在C语言中,整型常量有三种表示形式,分别是十进制整型、八进制整型和十六进制整型。其表示形式如表2-5所示。

表 2-5 整型常量的三种形式

进 制	表 达 方 式	举 例
八进制整型	由数字0开头	023, 067
十进制整型	与数学中的数字表示方法相同	123, -55
十六进制整型	由0X或0x开头	0x66, 0Xff

注意:不同进制的整型数所表示的数字的真实值是不同的。例如,023代表十进制数为19;067代表十进制数为55;0x66代表十进制数为102;0Xff代表十进制数为255。

整型常量分4种类型,即基本整型、短整型、长整型和无符号整型。

2. 实型常量

实型常量只用在十进制数中。它有单精度和双精度之分,其表示形式分为一般形式和指数形式两种。

一般形式的实数由整数部分和小数部分组成,例如:3.14、.777、-3.1415、-88.66。

指数形式的实数由尾数、e 或 E 和指数三部分组成，例如：0.33e5、5.55E-3、7.77e+15。其中 0.33、5.55、7.77 为尾数，e 和 E 后面的 5、-3、+15 均是指数。

C 语言规定，字母 e（或 E）之前必须有数字，且 e（或 E）后面的指数必须为整数。如写成 e4、6e8.3、e3、e 等都是不合法的指数形式。注意，字母 e（E）的前面以及数字之间不得插入空格。

3. 字符型常量

字符型常量是用一对单引号括起来的单个字符，在计算机的存储中占据一个字节，其中单引号为定界符。例如，'x'、'X'、'A'、'b'、'\$'、'*'等都是字符型常量。一个字符型常量的值就是该字符的 ASCII 码，如字符型常量'A'的 ASCII 码值为 65，字符型常量'a'的 ASCII 码值为 97。可见，单引号中的大写字母和小写字母代表两个不同的字符常量。

除以上形式的字符型常量外，C 语言还允许使用一种特殊形式的字符型常量，即转义字符型常量。转义字符型常量以“\”开头，后跟一个特定的字符，用来表示一些难以用一般形式表示的字符。转义字符看起来是多个字符，实际上却是作为一个字符来对待的。但如果反斜线之后的字符和它不构成合法的转义序列，则“\”不起转义作用而被忽略。常用的转义字符如表 2-6 所示。

表 2-6 转义字符

转义字符	含 义	转义字符	含 义
\n	回车换行	\f	换页
\t	水平制表符 (TAB)	\0	空操作字符 (ASCII 码为 0)
\v	纵向跳格	\'	单引号字符
\b	退格	\"	双引号字符
\r	回车	\a	响铃
\ddd	1~3 位八进制数所代表的字符	\xhh	1~2 位十六进制数所代表的字符

4. 字符串常量

在 C 语言中，用一对双引号括起来的零个或多个字符序列称为字符串常量。字符串中所含有字符的个数称为该字符串的长度。长度为零的字符串("")称为空串。例如，"C program"、"a"、""都为字符串常量。

若程序中使用了字符串，经编译后，系统会自动在每个字符串末尾加上空字符 '\0' 作为字符串的结尾标志，系统会以此判断字符串是否结束。所以，一个字符串常量在内存中所占的字节数是字符串的长度加 1。例如，"teacher" 的长度为 7，而在内存中所占的字节数为 8。

注意：'a' 和"a" 是不同的，前者是字符型常量，其长度为 1；而后者是以 '\0' 结束的字符串常量，在机器中占 2 个字节。

5. 符号常量

在 C 语言中可以用一个符号来代表一个常量，这个常量被称为符号常量。符号常量用预处理命令“#define”定义。符号常量一般用大写字母表示。符号常量在程序中必须先定义后使用。

符号常量定义的一般形式为：