

//
计算机与应用专业自学考试

高级语言程序设计 复习与应试指导

孙家骕 李颖 编著

1



北京大学出版社

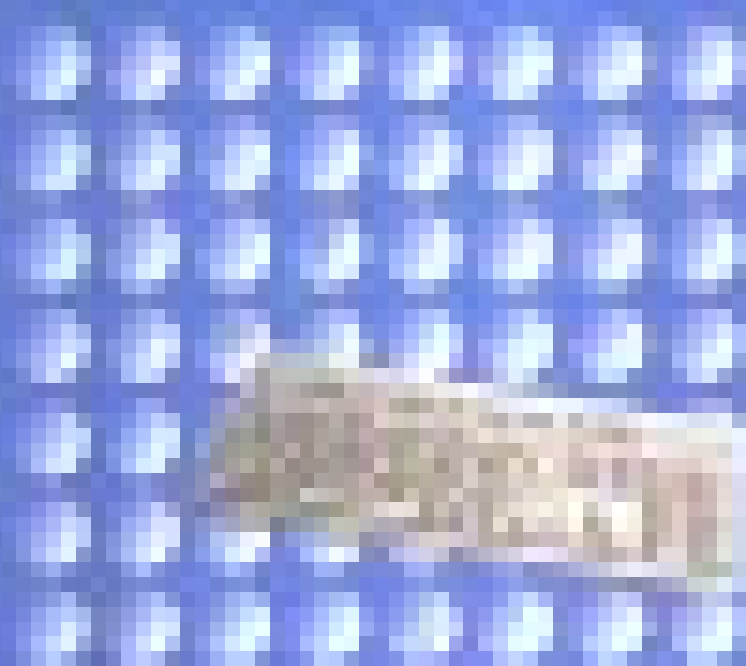
PEKING
UNIVERSITY
PRESS



计算机专业英语词汇表

高级语言程序设计 复习与应试指导

张圣明 李海 编著



清华大学出版社

计算机及应用专业自学考试

高级语言程序设计 复习与应试指导

孙家骅 李 颖 编著

北京大学出版社
北 京

内 容 简 介

本书根据全国高等教育自学考试指导委员会制定的《高级语言程序设计自学考试大纲》的要求,准确、详细地介绍、讲解了C语言的语法、语义及各种语言成分的用法。为便于自学考生的学习,书中给出了详细的题例分析和解答,还配合各章内容设计了大量的习题,并给出了习题答案。本书特别适合自学考生使用,也可以作为软件开发人员学习用书。

图书在版编目(CIP)数据

高级语言程序设计复习与应试指导/孙家骥等编著. —北京:北京大学出版社,2001.7
(计算机及应用专业自学考试丛书)
ISBN 7-301-05111-5

I. 高… II. 孙… III. 高级语言-程序设计-高等教育-自学考试-自学参考资料
N. TP312

中国版本图书馆 CIP 数据核字(2001)第 046323 号

书 名: 高级语言程序设计复习与应试指导

著作责任者: 孙家骥 李颖

责任编辑: 沈承凤

标准书号: ISBN 7-301-05111-5/TP·0557

出版者: 北京大学出版社

地 址: 北京市海淀区中关村北京大学校内 100871

网 址: <http://cbs.pku.edu.cn>

电 话: 出版部 62752015 发行部 62754140 编辑室 62752038

电子信箱: zpup@pup.pku.edu.cn

排 版 者: 兴盛达打字服务社 (62549189)

印 刷 者: 北京大学印刷厂印刷

发 行 者: 北京大学出版社

经 销 者: 新华书店

787毫米×1092毫米 16开本 16.25印张 406千字

2001年8月第1版 2001年8月第1次印刷

定 价: 22.00元

前 言

20 多年高等教育自学考试事业的蓬勃发展已经证明,自学考试是继续教育的一种重要形式,它为我国社会主义建设事业培养了大量的有用人才。特别是近年来各行各业对于计算机专业人才的大量需求使得普通高校计算机专业的毕业生供不应求。因此,高教自考的“计算机及应用”专业受到了广大考生的热烈欢迎。但是对于自考生来说,这个专业的学习并不是一件轻松的事情。它要求考生必须熟练掌握所学课程的基本概念、基本理论以及基本的解题方法和技巧。

北京大学计算机科学技术系多年来一直承担着北京市“计算机软件及应用”专业(自 2001 年按照全国统考的新计划,改为“计算机及应用”专业)的主考任务,许多教师直接参与了大量的命题、阅卷等工作。他们不仅有着丰富的教学经验,同时在多年的主考工作中积累了大量的试题资料。他们对于自考的特点和自考生在学习中的困难有着比较深入的了解和切身的体会。为了帮助自考生切实掌握“计算机及应用专业”考试课程的重点和难点,北京大学出版社组织计算机系的教师编写了这套专业课程的辅导丛书。它按照考试计划所规定的课程来组织,每本辅导书都包含下述主要内容:

1. 以每门课程的考试大纲为基础,按照章节进行总结,提纲挈领,将零散的知识点串起来,为每个考生列出了包含课程主要知识点的系统的复习提纲。

2. 在每一章后面都通过大量的例题讲解各种试题的题型、相关的解题方法和规范,并根据历年阅卷的经验,进一步说明考生在解题中容易发生的错误和应该注意的问题。

3. 针对每一章的要求,为考生提供大量的练习题和答案,以方便考生循序渐进地巩固所学的知识,掌握解题的技巧。

4. 为了进一步培养比较灵活的分析问题和解决问题的能力,在全书的后面附有综合练习。在系统复习了全部内容后,考生可以用它进行自我测试和考前的模拟冲刺。

这套丛书配合考试用书,既可以在日常自学中作为辅导书,也可以用于考前的总复习,同时也为其他在校学生和科技人员学习相关课程提供指导和帮助。

我们期望这套丛书能够成为广大考生的良师益友,我们更诚挚地期待着读者的批评和指正。

北京大学计算机科学技术系

副系主任 屈婉玲

2001 年 7 月 5 日

编者的话

为了帮助参加全国高教自考“计算机及应用”专业的考生朋友们复习和应考“高级程序语言设计”，我们编写了这本辅导用书。

本书是根据全国高等教育自学考试指导委员会制定的“高级程序语言设计自学考试大纲”的要求，参考全国高等教育自学考试指导委员会组编的教材《高级程序语言设计》一书（迟成文主编、经济科学出版社 2000 年 3 月出版）编写的。

本书包含考试大纲要求的全部内容，全书共分 10 章。前 9 章介绍了高级程序设计语言 C 的语法、语义和用法。每章都由内容提要、题例分析、习题、习题答案及教材（指上面提到的《高级程序语言设计》一书）中习题答案五部分组成。

“内容提要”部分总结了相应章涉及的语言成分的形式、含义和用法及其中的难点；“题例分析”部分列举了各种典型的题目，给出了详细的分析和解答。通过这些例题，讲解了程序设计的方法，并指出了初学者容易犯的错误；“习题部分”给出了大量的题目，它们涉及了相应章节的所有知识点；“习题答案”部分给出了“习题部分”中的所有习题的详细解答；“教材中习题答案”部分给出了教材相应章节习题的详细解答。

本书的第 10 章给出了五套模拟试题，试题的题型、题量、难度及各题的分值都是参考《2000 年下半年高等教育自学考试全国统一命题考试高级语言程序设计》试题而设计的。

本书中的所有 C 语言程序都是遵循 ANSI 1990 C（最新）标准的，在 Turbo C 环境中编译这些程序，可能要删去 `#include<stdio.h>` 等语句。

本书主要内容由孙家骊博导编写，部分章节的习题解答由李颖老师完成。

由于水平所限，书中难免有疏误之处，敬请广大考生和读者提出宝贵意见。

编者

2001 年 7 月于北京大学

目 录

第一章 概述	(1)
1.1 内容提要	(1)
1.2 题例分析	(2)
1.3 习题	(4)
1.4 习题答案	(5)
1.5 教材中第 1 章习题(p. 9)答案	(5)
第二章 基本数据类型和运算	(7)
2.1 内容提要	(7)
2.2 题例分析.....	(11)
2.3 习题.....	(14)
2.4 习题答案.....	(18)
2.5 教材中第 2 章习题(p. 34—p. 36)答案	(19)
第三章 顺序结构、选择结构和循环结构的程序设计	(20)
3.1 内容提要.....	(20)
3.2 题例分析.....	(31)
3.3 习题.....	(37)
3.4 习题答案.....	(44)
3.5 教材中第 3 章习题(p. 66—p. 69)答案	(50)
第四章 数组	(55)
4.1 内容提要.....	(55)
4.2 题例分析.....	(62)
4.3 习题.....	(66)
4.4 习题答案.....	(73)
4.5 教材中第 4 章习题(p. 86—p. 89)答案	(79)
第五章 指针	(85)
5.1 内容提要.....	(85)
5.2 题例分析.....	(87)
5.3 习题.....	(93)
5.4 习题答案.....	(99)
5.5 教材中第 5 章习题(p. 110—p. 112)答案	(106)
第六章 函数	(112)
6.1 内容提要	(112)
6.2 题例分析	(122)
6.3 习题	(131)

6.4	习题答案	(137)
6.5	教材中第 6 章习题(p. 137—p. 140)答案	(143)
第七章	结构型、共用型和枚举型	(146)
7.1	内容提要	(146)
7.2	题例分析	(153)
7.3	习题	(162)
7.4	习题答案	(167)
7.5	教材中第 7 章习题(p. 168—p. 170)答案	(173)
第八章	文件.....	(176)
8.1	内容提要	(176)
8.2	题例分析	(179)
8.3	习题	(185)
8.4	习题答案	(190)
8.5	教材中第 8 章习题(书中 p. 191—p. 193)答案	(198)
第九章	编译预处理与带参数的主函数.....	(203)
9.1	内容提要	(203)
9.2	题例分析	(206)
9.3	习题	(209)
9.4	习题答案	(212)
9.5	教材中第 9 章习题(书中 p. 204—p. 205)答案	(216)
第十章	综合练习.....	(219)
10.1	模拟试题部分.....	(219)
10.2	试题答案.....	(243)

第一章 概 述

学习本章要达到的目标包括：理解程序、程序设计的含义；了解 C 语言程序的基本结构和基本语句；掌握 C 语言的标识符的构成规则以及熟记 C 语言的关键字（即教材中提到的保留字）。

1.1 内 容 提 要

1. 程序与程序设计

程序不仅包含对解决某个问题方法的步骤描述，还包含对解决该问题所要处理的对象的描述。程序中所描述的步骤应当是确定的，不能是似是而非的；应当是可以执行的，不能是根本无法执行的；应当是有穷多个，不能是无限多个。程序中的某些步骤可以循环执行多次。程序的描述手段（也称为程序设计语言）可以多种多样，可以用自然语言（如会议程序）；可以是图形语言（如描述武功的程序）；也可以是计算机语言（如计算机程序）。比如，烹调程序的某个步骤是加佐料，那么这种佐料必须是确定的，不能是可以这样理解，也可以那样理解；而且这种佐料是可以获取的，不能是根本不存在的。这样才能保证这一步骤可以执行。其中的搅拌这个步骤一般要执行多次。烹调程序还要包含对所用原材料的准确描述。

程序设计是设计、编写、调试程序的方法与过程。程序设计的目的是解决某一特定问题。为达到这一目的，要首先分析、理解该问题，在此基础上，设计出解决这个问题的方法的各个步骤，再用相应的程序语言描述这些步骤，编写出相应的程序；最后经过对程序的调试，找出并修改其中的错误，使程序达到解决该问题的要求。

2. C 程序基本结构

C 语言是一种用来进行人与计算机交流的高级程序设计语言，它是一种过程语言。C 程序是用 C 语言描述的程序，它由一系列的 C 语言成分（说明语句或称数据定义语句、可执行语句以及函数定义等）组成。一个完整的 C 程序包含一个或多个函数，其中有且只有一个函数称为主函数，它的名字为 main。任何一个 C 程序的执行，都是从它的主函数 main 开始。

3. C 语言的基本语句

C 语言的语句可分为如下几类：

- (1) 说明语句（或称数据定义语句）；
- (2) 表达式语句（包括赋值语句和函数调用语句）；
- (3) 流程控制语句（包括分支语句、循环语句、转移语句）；
- (4) 其它语句。

C 语言的所有语句，除去复合语（以符号 { 开始，以符号 } 结束）外，都是以分号“;”结束。

4. C 语言的字符集

C 语言的字符集分四大类：英文大小写字母 52 个；十进制数字 10 个；键盘符号 32 个；转义字符，它们以反斜杠字符（\）开始，后跟若干个字符。转义字符主要用于表示特殊字符，但是

也可以表示前三类的任意一个字符。

5. 标识符

C 语言的标识符是由 1 个或多个字母(52 个英文大小写字母)、下划线(_)及数字(0、1、2、...、9)组成的且第 1 个字符必须是字母或下划线的字符串。标识符的长度(所含字符个数)最大可以是 32(早期的 C 编译系统把标识符长度限制在 8 以内),如果用户写的标识符长度超过 32,那么超过部分被忽略。这就是说,两个长度大于或等于 32 的标识符,如果它们的前 32 个字符都对应相同,那么,不管其余的字符是否对应相同,这两个标识都被认为是同一个标识符。

1.2 题 例 分 析

【例 1.1】 下面是一个非常简单的 C 程序,它的功能是在屏幕上显示: Hello the world。该程序仅包含一个函数,即主函数 main。

程序 1.1

```
# include <stdio.h>
void main( )
{
    printf("Hello the world\n");
}
```

这个程序的第 1 行是文件包含命令(或称文件包含语句),其中尖括号(<>)内的是文件名。该语句的作用是表示用文件名所指文件的内容来替换这个文件包含语句。程序 1.1 中<>内文件 stdio.h 称为标准输入输出头文件,其中有 printf 的信息,由于程序 1.1 的主函数内调用了 printf 函数,所以使用文件包含语句把文件 stdio.h 的内容嵌入程序中。第 2 行是主函数说明,它由返回值类型 void、函数名 main 及可含可不含形式参数的圆括号对()构成。类型包含两个方面:类型的值集合及作用于类型值集合上的运算集合。void 是一种特殊的类型,它的值集合为空集合,可以看成没有值。所以返回值类型为 void 的主函数 main 实际上无返回值;从第 3 行的{到第 5 行的},是函数 main 的函数体,由于其功能简单,不含变量定义部分,只有一个函数调用语句 printf("Hello the world\n");

【例 1.2】 下面的程序用于计算给定三角形的面积,要求给出三角形的两条边长度及这两条边夹角的大小(按角度计)。由三角形面积公式知道,若两条边大小分别是 a 和 b,它们之间夹角为 C,则该三角形面积 $S = \frac{1}{2}ab\sin C$ 。

程序 1.2

```
# include <stdio.h>
# include <math.h>
# define PI 3.1415926
double area (double a, double b, double C)    /* 1 行 */
{
    double S;    /* 2 行 */
    S = 0.5 * a * b * sin(C/180.0 * PI);    /* 3 行 */
}
```

```

        return S;    /* 4 行 */
    }
    void main( )    /* 5 行 */
    {
        double a, b, C, S;    /* 6 行 */
        printf ("Input a, b and C please \n");    /* 7 行 */
        scanf ("%f%f%f", &a, &b, &C);    /* 8 行 */
        S=area(a, b, C);    /* 9 行 */
        printf ("The area is %10.2f \n", S);    /* 10 行 */
    }

```

该程序由两个函数构成：函数 area 和主函数 main。函数 area 用于计算三角形面积，它有 3 个参数 a、b 和 C，它们依次是三角形的两条边长和这两条边夹的角的大小（角度制）。为了有较高的精度，a、b、C 都是 double 类型，而函数 area 的返回值即为给定的三角形的面积，其类型也是 double。在函数 area 中，调用了正弦函数 sin，该函数要求的角度是弧度制的值，而参数 C 给出的是角度制值，因此要把 C 转换成弧度制值，即乘以 3.1415926（圆周率），再除以 180。即执行 $C/180.0 * \text{PI}$ （这里 PI 是符号常量，用 #define PI 3.1415926 加以定义。其含义是 PI 就代表 3.1415926，这也是为什么要在程序的第 3 行处加上 #define PI 3.1415926 的原因）。正弦函数 sin 是在文件 math.h（数学库头文件）中说明的，所以程序第 2 行出现包含语句 #include <math.h>。程序 1.2 中的“1 行”是函数 area 的函数说明部分，它包括函数返回值类型（这里是 double）、函数名（area）和函数的形式参数部分（这里是 double a、double b、double C）。其中的形式参数部分采用的是 ANSI 90 C 语言标准规定的写法，在早期的 C 语言版本中，形式参数写法如下：

```

(形式参数名 1, ..., 形式参数名 n)
数据类型 形式参数名 1, ..., 形式参数名 k;
:
数据类型 形式参数名 m, ..., 形式参数名 n;

```

按早期 C 语言版本标准，函数 area 的形式参数部分可以写成 (a, b, C)

```
double a, b, C;
```

建议读者采用 ANSI 90 C 标准规定（即前一种写法），因为它与 C++ 规定是一致的。

程序 1.2 中的“2 行”说明了一个 double 类型的变量 S，用于存储三角形面积值；“3 行”是根据三角形面积计算公式求出边长为 a、b 及这两个边夹角为 C（按角度制计）的三角形面积值，并把该值存储在变量 S 内；“4 行”是把计算出的三角形面积值（在 S 内）返回给函数（area 函数）的调用者；“5 行”是主函数 main 的说明部分，它的返回值类型为 void、函数名为 main。返回值类型为 void 的函数相当于无返回值。这个 main 函数没有形式参数（注意，任何 C 语言的函数都可以没有形式参数，但是圆括号不能没有）；“6 行”说明了四个类型都是 double 的变量 a、b、C 和 S，依次用于存储两个边长值、两边夹角值（按角度计）和三角形面积值。它们的类型之所以采用 double，是因为要获得较高的精度。如果采用 float 类型，一般也能满足要求。但是如果采用 int 或 long 类型，则会造成很大的误差（比如，若边长为 0.99，采用 int 或 long 型，就

会被近似成 0 了,这就使面积成为 0 了);“7 行”是函数调用语句,它调用库函数 printf,输出一个提示程序 1.2 的用户的信息:“Input a, b and C please”(请输入 a、b 和 C),通常这一信息显示在屏幕上。这样做是为了方便程序用户,提醒他下一步做什么;“8 行”是函数调用语句,它调用库函数 scanf,把用户输入的(通常是通过键盘输入)三角形的两个边长和这两个边的夹角(按角度制计大小)依次读入到变量 a、b 和 C 中;“9 行”是调用函数 area 来计算三角形面积,并把该面积值存储在变量 S 中。调用 area 时,要提供 3 个参数(两个边长和这两个边的夹角);“10 行”是函数调用语句,通过调用库函数 printf 把三角形面积以特定的格式(即小数点后面要 2 位,总的位数至少为 10 位)输出。

程序 1.2 经编译系统编译成可执行代码后便可执行。执行时,首先从主函数的第一句开始,即执行“7 行”中语句,输出“Input a, b and C please”,接着用户通过键盘输入某三角形的两条边长 a、b 及该两边夹角 C,比如,输入的是 3.0、4.0、90.0,输入完三个实数(相邻的实数用空格或换行符隔开)后,再键入回车键,计算机便继续执行“8 行”中的语句,把三个实数依次读入给变量 a、b、C,于是 a、b、C 的值依次是 3.0、4.0 和 90.0;计算机继续执行“9 行”的函数调用语句,并以 a、b、C 的值作为实在参数,传递给被调用函数 area,在函数中依次执行“3 行”和“4 行”的语句,先后计算求值、赋值,使 S 值为 6.0($0.5 * 3.0 * 4.0 * \sin(90/180.0 * 3.1415926)$),最后把 6.0 这一函数值返回给调用者,接着,调用者把 6.0 赋值给主函数 main 中的变量 S,再继续执行“10 行”的函数调用语句,就可以得到如下所示的输出:

The area is 6.00

(该面积是 6.00,其中₁₁表示空格)。

1.3 习 题

一、单项选择题

- 计算机可以直接执行的程序是 ()
 ① 机器语言程序 ② C 语言程序 ③ 高级语言程序 ④ 汇编语言程序
- 下列语言中面向机器的语言是 ()
 ① PASCAL 语言 ② 二进制语言 ③ C 语言 ④ Ada 语言
- 下列字符或字符序列中属于键盘符号的是 ()
 ① 0 ② \0 ③ \a ④ \n
- 下列正确的转义字符是 ()
 ① \1234 ② 057 ③ \ ④ \\060
- 下列是 C 语言的关键字的是 ()
 ① begin ② registor ③ const ④ end
- 下列不是 C 语言关键字的是 ()
 ① volatile ② auto ③ void ④ scanf
- 下列是 C 语言标识符的是 ()
 ① Aa+ ② --123 ③ a * bc ④ b&cd
- 下列不是 C 语言标识符的是 ()
 ① start ② _1-2-3 ③ A+B12 ④ e2001

9. 下列可以作为 C 语言变量名字的是 ()

- ① if ② else ③ const ④ size

10. 下列可以作为 C 语言注释的是 ()

- ① /* This is a key */ ② /* comment * \
③ (* pascal comment *) ④ */ a comment */

二、填空题

1. C 语言源程序恰好包含 1 个_____函数,它的名字是_____。
2. C 语言程序经过_____后,才能在计算机上执行。
3. C 语言的_____是保留的。
4. C 语言的标识符的第 1 个字符必须是_____或_____。
5. C 语言程序的执行是从_____函数开始,在_____函数中正常结束。
6. C 语言的主函数可以调用任意_____函数。
7. C 语言的标识符中只能含_____、_____和_____三种字符。
8. C 语言的任何关键字都仅含_____。
9. C 语言的复合语句是用_____括起来的一串语句。
10. C 语言的语句除复合语句外,都以_____结束。

1.4 习题答案

一、单项选择题

1. ① 2. ② 3. ① 4. ③ 5. ③
6. ④ 7. ② 8. ③ 9. ④ 10. ①

二、填空题

1. 主 main
2. 编译或解释
3. 关键字
4. 英文字母 下划线_
5. 主 主
6. 非主
7. 英文字母 十进制数字 下划线_
8. 小写英文字母
9. 一对花括号{ }
10. 分号;

1.5 教材中第 1 章习题(在 p. 9)答案

一、单项选择题

1. ① 2. ② 3. ① 4. ③ 5. ② 6. ④ 7. ④

二、填空题

1. 机器的 问题的
2. 源 解释 编译 目标
3. 反斜线\
4. 常量 变量 运算符 函数调用 表达式 保留字
5. 数据定义 赋值 函数调用 表达式 流程控制 复合 空 其它
6. 1 主 主
7. / * * /

【注】

2.5 节中给出的答案是根据教材叙述做出的,有些不够准确。如一、中的②,C 语言和 ALGOL 语言也不是面向问题的语言,它们应看成是面向过程的语言;二、中的①,高级语言不一定是面向问题的语言,比如 C、FORTRAN、PASCAL、ALGOL 等,都是高级语言,但又都不是面向问题的语言。还有二、中的 4 和 5 的答案也不够确切。

第二章 基本数据类型和运算

学习本章的目的是掌握 C 语言中的基本数据类型、基本数据类型的变量、常量以及施于其上的运算;在此基础上,掌握由变量、常量等运算对象及运算符构成的表达式。

2.1 内容提要

1. C 语言的基本数据类型

C 语言提供的基本数据类型可分成整型、实型、字符型三类。其中整型又分成短整型(short)、一般整型(int)和长整型(long);实型又分成单精度浮点型(float)和双精度浮点型(double)。C 语言的数据类型除上述基本数据类型外,还有构造类型(可分成数组类型、结构类型和共用类型,共用类型也称联合类型)、指针类型、枚举类型和空类型。各种数据类型都有自己的值集合及施于该值集合上的运算集合。

2. 常量

常量可分为整型、实型、字符型和字符串型四种常量。另外,还有一种命名常量,称为符号常量。

(1) 整常量也就是整常数,有三种写法:十进制、八进制及十六进制。十进制即为通常的写法;八进制写法是以 0 打头,十六进制写法是以 0x(小写 x)或 0X(大写 X)打头。每种写法都可以带正(+)号或负(-)号。例如,十进制整常数 0、+15、-120、126,八进制整常数 00、+015、-0270、01256,十六进制整常数 0X0、0x1A、+0x15、-0xa0、0X1B 等。在十六进制整数中,每一位的 10、11、12、13、14、15,依次用 a、b、c、d、e、f(或 A、B、C、D、E、F)表示。例如 0XA 表示的是十进制的 10,0x1B(或 0X1b)表示的是十进制的 27。

按整常数所占内存空间大小,又可分成一般整数和长整数两种。长整数是在整数后边加上一个 L(或小写字母 l)。例如 1L、25l、12L、017L、0XABL 等。长整数实际上是长整型(long)的常数,通常,它们在内存占有 4 个字节的存储空间。

如果一般整数用 2 个字节(即 16 位二进制)表示,那么,最大值为 32767,最小值为 -32768。如果要表示的值为大于 32767 或小于 -32768,则应用长整数形式表示。例如 32768L、-32769l、100000l、-200000l 等。

(2) 实常量就是实数。由于计算机硬件上的限制,实常量集合仅是数学上的实数集合的一个有穷子集合。实数只用十进制数,表示方法有两种:一般形式(也称通常表示法)和指数形式(也称科学表示法)。实数的一般形式由整数、小数点以及小数三部分构成,整数和小数部分都可以省略,但是不能同时省略。例如 12.56、12.、.56 都是正确的实数形式,实数的指数形式由尾数部分和指数部分构成。尾数部分可以是一个整数或一个实数的一般形式;指数部分由一个 e(或 E)及整数(称为指数)构成,这种形式的实数值为尾数 * 10^{指数}。例如 12.56e-2、32E4、.15e+2、2.e3 都是正确的实数形式,它们的值依次为 0.1256、320000.0、15.0 和 2000.0。实数的这两种形式都可以带一个符号(正号或负号),例如,+12.、-.15、-12.57、+1E2、

-15.6e-2、-.5e3。

上述两种形式的实数的类型都是 double 类型。若表示 float 型的实数,应当加上后缀 f(或 F),例如 1.25 类型为 double,而 1.25f 类型为 float。1e5、-2.e-3、.12e3 等类型都是 double,而 1e5f、-2.e-3F、.12e3f 的类型都是 float。

(3) 字符常量就是字符。字符的表示有两种方式:一对单引号括起来的一个字符或一对单引号括起来的转义字符序列。其中,转义字符序列是反斜线(\)后跟一个字符,或\后跟 1 位到 3 位八进制数,或\后跟 2 位十六进制数(十六进制数以 x 或 X 开头)。每个字符都有一个 ASCII 代码值,\后跟的八进制数或十六进制数的值就是要表示的字符的 ASCII 代码值。用转义字符序列的方式可以表示所有的字符。例如,'a'、'+'、','都是用第 1 种方式表示的字符。'\n'、'\a'、'\123'、'\x41'、'\x2B'、'\x61'、'\47'都是用第 2 种方式表示的字符,它们依次是回车换行符、响铃符、字符 S、字符 A、字符+、字符 a 及字符,。

(4) 字符串常量是一对双引号括起来的字符串,串中的字符可以是单个字符,也可以是转义字符序列。例如,"abcd"、"\141b\143\144"、"\x61\x62\x63\x64"表示的都是字符串 abcd。而"\0601\0623"表示的是字符串 0123。要特别注意,字符串常量所占的内存字节数等于该字符串长度加 1,这是因为要用一个字节存放字符'\0',它用作字符串的结束标记。

(5) 符号常量是一种取了名字的常量,通过"#define 标识符 常量"来定义符号常量,例如,#define PI 3.1415926 定义了一个符号常量,其名字为 PI,其值为实数 3.1415926,这样做可提高程序的可读性和可修改性。

3. 变量

通常,变量的值在程序运行过程中可以改变(除去用 const 修饰的变量)。一个变量应当有类型(其值的类型),还可能有初始值,还有名字,此外,还可能有存储类型及类型修饰。

int a;是一个最简单的变量说明;它说明了一个类型为 int、名字为 a 的变量。变量 a 没有初始值,也没有存储类说明,也没有类型修饰。int b=10;说明了一个名字为 b、类型为 int、初始值为 10 的变量。static int c=100;说明了一个名字为 c、类型为 int、初值为 100、存储类为 static(静态)的变量。const static int d=1000;说明了一个名字为 d、类型为 int、初值为 1000、存储类为 static(静态)、类型修饰为 const 的变量,用 const 修饰的变量,其值在程序运行中不能被修改。所以任何对 d 赋值的语句都是错误的。另一个类型修饰符是 volatile(易变的),用 volatile 修饰的变量,其值极易改变,甚至读该变量时都会改变它的值。例如,

```
volatile int v=100,
int s=5, t=5;
s=v+10;
t=v+10;
```

在上述两个语句中的 v 值可能不相等,这是因为 v 是受 volatile 修饰的,在第 1 个语句中对 v 值的引用(读)会改变 v 的值,所以在第 2 个语句中的 v 值与第 1 个语句中 v 值不相等。volatile 的作用是指示编译程序对含有用 volatile 修饰的变量的表达式不做优化。在变量说明中,类型和变量名字是必须有的,其它各项(初始值、存储类、类型修饰)可以根据需要取舍。

(1) 变量的存储类型有 4 种: auto(自动型)、register(寄存器型)、static(静态型)和 extern(外部参照型)。

(2) 变量的类型修饰符有 2 个: const 和 volatile。

4. 运算符

C 语言有丰富的运算,总计有 47 种,其中有些运算共用同一个运算符。例如,乘运算和间接取运算用的是同一个运算符 *;取地址运算和按位逻辑与运算用的是同一个运算符 &。学习本节一定要记住各运算符的写法、功能、优先级和结合性。表 2.1 给出了 C 运算符优先级和结合性。

表 2.1 C 运算符的优先级及结合性

高 ↓ 低	() [] --> . 后++ 后--	左结合
	! ~ 前++ 前-- - (type) * &, sizeof +	右结合
	* / %	左结合
	+ -	左结合
	<< >>	左结合
	<= < >= >	左结合
	== !=	左结合
	&	左结合
	~	左结合
		左结合
	&&	左结合
		左结合
	?:	右结合
	= += -= *= /= %= <<= >>= &= ^= =	右结合
	,	左结合

5. 表达式

C 语言中表达式是递归定义的:

① 一个常量、一个变量(包括简单类型的变量、数组变量的一个元素、结构变量或共同体变量的一个成分)、一个函数调用都是表达式;

② 前缀单目运算符(+、-、!、~、++、--等)后面跟上(1)中的表达式也是一个表达式;

③ ①中的表达式后面跟上一个后缀单目运算符(++、--等)也是一个表达式;

④ 两个表达式中间加上一个双目运算符(如+、-、*、/、%等)也是表达式;

⑤ 表达式? 表达式:表达式也是表达式;

⑥ 有限次使用②--⑤的规则,也可以得到一个表达式。

学习本节时,要特别注意几点:表达式类型的处理;表达式中的哪个子表达式不计算;表达式中各运算符的计算次序。

(1) 表达式类型处理

表达式求值过程中要对参与运算的对象的类型进行处理,这种处理分两步进行:首先对在有运算符存在的表达式中,把所有的类型为 char 和 short(包括 signed 和 unsigned 的 char 和 short)的对象类型转换成 int 类型;其次,经上述转换后,若参与一个运算的几个对象类型仍不相同,则进行如下规定的转换:

若某个运算对象为 double 类型

则另一个对象类型转换为 double 类型