

C 语言概述

学习目标

- 掌握 C 程序的发展史与特点。
- 掌握 C 语言的基本语法。
- 掌握设计简单 C 程序的方法。
- 掌握 C 程序的开发过程。

1.1 初识 C 语言

语言是人与人之间交流的工具。程序设计语言是人与计算机交流的工具,C 语言是其中的一种。程序是使用程序设计语言编写出的一些语句序列,是人和计算机交流的方式。

1.1.1 C 语言的发展史

第 1 阶段: A 语言

C 语言的发展颇为有趣,真要寻根探源,它的原型还是 ALGOL60 语言,也称 A 语言。

1960 年国际计算机委员会设计了一种面向问题的结构化程序设计语言 ALGOL60 语言,但它不能直接对硬件进行操作,所以不宜用来编写系统程序。

第 2 阶段: CPL 语言

1963 年,英国剑桥大学在 ALGOL60 基础上推出 CPL (combined programming language)语言,该语言比较接近于硬件,但规模比较大,难以实用。

第 3 阶段: BCPL 语言

1967 年,英国剑桥大学的马丁·理查德(Martin Richards)对 CPL 语言进行了简化,推出了 BCPL(basic combined programming language)语言。

第 4 阶段: B 语言

1970 年,美国贝尔实验室的肯尼思·汤普森(Kenneth Lane Thompson)以 BCPL 语言为基础,又做了进一步简化,设计出了很简单且接近硬件的 B 语言(取 BCPL 的第一个字母),并用 B 语言编写了第一个 UNIX 操作系统,在 DEC 公司的 PDP-7 小型机上运行。

此时的 B 语言过于简单,而且功能有限。

第 5 阶段: C 语言

1972 年,美国贝尔实验室的丹尼斯·麦卡利斯泰尔·里奇(Dennis MacAlistair Ritchie)在 B 语言的基础上设计出了 C 语言(取 BCPL 的第二个字母)。C 语言既保持了 BCPL 和 B 语言的优点(接近硬件),又克服了它们的缺点(过于简单、数据无类型等)。

1973 年肯尼思·汤普森和丹尼斯·麦卡利斯泰尔·里奇合作将 90% 以上的 UNIX 操作系统用 C 语言进行改写(即 UNIX 第 5 版)。UNIX 操作系统由于使用了 C 语言而取得成功,几乎成为 16 位微机的标准操作系统。C 语言从诞生开始就由于编制 UNIX 操作系统的成功而引起了人们的关注。

1973 年之后,C 语言的发展相当迅速。1975 年 UNIX 第 6 版公布,1977 年又研制成功不依赖具体机器的 C 语言编译文本《可移植 C 语言编译程序》,推动了 C 语言在各种机型上的广泛应用。

1978 年,UNIX 第 7 版研制成功。以其中的 C 编译系统为基础,丹尼斯·麦卡利斯泰尔·里奇被称作 C 语言之父(UNIX 之父)和布赖恩·威尔森·克尼汉(Brian Wilson Kernighan)合著了影响深远的名著 *The C Programming Language*,被称为标准 C。

1983 年,美国国家标准化协会(ANSI)根据 C 语言问世以来各种版本对 C 语言的发展和扩充,制定了新的标准草案,即 83 ANSI C。

1987 年,ANSI 又公布了新标准,即 87 ANSI C。

1989 年,ANSI 又公布了新的标准,即 ANSI X 3.159—1989(简称 C89)。

1990 年,国际标准化组织(ISO)接受 C89,作为国际标准 ISO/IEC 9899: 1990,称为标准 C,简称 C90。ISO 的 C90 和 ANSI 的 C89 基本上是相同。

1995 年,ISO 对 C90 做了一些修订,称为 C95。

1999 年 C 语言标准又得到改进,这就是 ISO/IEC 9899: 1999,俗称 C99,该标准于 2000 年 3 月被 ANSI 采用。但目前各软件厂商提供的 C 语言编译系统都未实现 C99 建议的功能,例如 Microsoft(微软)和 Borland 公司,大多数是以 C89 为基础进行开发的。但厂商提供的不同版本的 C 编译系统所实现的语言功能和语法规则又略有差别。因此读者应了解所用的 C 语言编译系统的特点。本书的叙述基本上以 C89 为基础。

目前,在计算机上广泛使用的 C 语言编译系统有 Turbo C、Borland C++ Builder、Microsoft Visual C++(简称 VC++)以及 Microsoft Visual C++.NET 等。本书使用的开发环境是 VC++ 6.0 系统。

计算机编程语言总的来说可以分为机器语言、汇编语言和高级语言三大类。C 语言是目前国际上广泛流行的计算机高级语言,既可用于编写系统软件,也可用于编写应用软件,集汇编语言和高级语言的优点于一身。因此熟练掌握 C 语言是计算机开发人员的一项基本功。

1.1.2 C 语言的特点

每一种语言都有自己的特点,C 语言也不例外,所以才有了语言的更替,有了不同语

言的使用范围。C语言主要有以下特点。

1. 简洁紧凑、灵活方便

C语言一共只有32个关键字,9种控制语句,程序书写自由,主要用小写字母表示。它把高级语言的基本结构和语句与低级语言的实用性结合起来。

2. 运算符丰富

C语言的运算符包含的范围广泛,共有44种运算符。C语言把括号、赋值、强制类型转换等都作为运算符处理,从而使C语言的运算类型极其丰富,表达式类型多样化。灵活使用各种运算符,可以实现在其他高级语言中难以实现的运算。

3. 数据类型丰富

C语言的数据类型有整型、实型、字符型、数组类型、指针类型、结构体类型、共用体类型等,能用来实现各种复杂的数据结构的运算。C语言引入了指针概念,使程序效率更高。另外,C语言具有强大的图形功能,支持多种显示器和驱动器,且计算功能、逻辑判断功能强大。

4. C语言是结构化和模块化的语言

C语言具有结构化的控制语句,控制程序流向,是一种结构化程序设计语言。这种结构化方式可使程序层次清晰,便于使用、维护和调试。C语言是以函数形式提供给用户的,用函数作为程序的模块单位,这些函数可方便地调用,便于实现程序的模块化。

5. C语言语法限制不太严格,程序设计自由度大

一般的高级语言语法检查比较严格,能够检查出几乎所有的语法错误。而C语言允许程序编写者有较大的自由度。C语言程序生成代码质量高,程序执行效率高。

6. C语言可以进行底层开发

C语言允许直接访问物理地址,可以直接对硬件进行操作,C语言可以像汇编语言一样对位、字节和地址进行操作,而这三者是计算机最基本的工作单元,可以用来编写系统软件。

7. C语言适用范围大,可移植性好

C语言适用于多种操作系统,如DOS、UNIX、Windows等,也适用于多种机型。

综上所述,C语言是一种功能很强的语言。但是,它也有一些不足之处:C语言语法限制不严格,虽然可使熟练的程序员编程更灵活,但安全性低;运算符丰富,完成功能强,但难记、难掌握。因此,学习、使用C语言不妨先学基础部分,先用起来,用熟练后再学习语法规则,进而全面掌握C语言。

1.1.3 第一个 C 程序

下面通过一个例题来认识 C 程序。

【例 1-1】 在计算机屏幕上输出一行字符。

```
1  #include <stdio.h>
2  void main()           /* 函数定义,函数名称为 main(),称作主函数 */
3  {
4      printf("This is a C program.\n");
5  }
```

说明:

(1) 程序第 1 行代码的作用是进行相关的预处理操作。其中字符#是预处理标志,用来对文本进行预处理操作,include 是预处理命令,它后面跟一对尖括号,表示头文件在尖括号内读入,stdio.h 就是标准输入/输出头文件,因为程序要使用输入/输出函数,所以一般 C 程序的开头都要写上这样一行命令。放在源程序的最前面,用来提供输入/输出函数的声明。

(2) 程序第 2 行代码声明了一个 main()函数,该函数是程序的入口。main()是函数名字,表示“主函数”;main()前面的 void 表示此函数是“空类型”,void 是“空”的意思,即执行此函数后不产生一个函数值。每一个 C 程序都必须有一个 main()函数,程序总是从 main()函数开始执行。程序中以/* 开头到 */结尾中的所有内容表示注释。在编译时注释行被忽略,并不产生代码行。

(3) 程序第 3 行,函数体由花括号“{”表示开始。

(4) 程序第 4 行代码调用了一个用于格式化输出的 printf()函数,在程序中 printf()函数的作用是把括号内双引号之间的字符串按原样输出。\\n 是换行符,即在输出 This is a C program.后回车换行。这行程序末尾的分号,则是 C 语句结束的标志。

(5) 程序第 5 行,函数体由花括号}括起来,表示程序结束。

运行结果为

```
This is a C program.
```

1.2 C 语言的语法基础

1.2.1 C 语言字符集

C 语言的基本符号可分为 4 类,归纳如下。

- (1) 英文字母: 大小写各 26 个,共计 52 个。
- (2) 阿拉伯数字: 0、1、2、3、4、5、6、7、8、9,共 10 个数字。
- (3) 下画线: _。
- (4) 特殊符号: 通常由 1 或 2 个符号组成,主要用来表示运算符。下面列举几种常用

运算符。

- 算术运算符：+、-、*、/、%、++、--。
- 关系运算符：<、>、>=、<=、==、!=。
- 逻辑运算符：&&、||、!。
- 位运算符：&、|、~、^、>>、<<。
- 条件运算符：?:。
- 赋值运算符：=。
- 其他分隔符：()、[]、{ }、...、,、;。

1.2.2 标识符

1. 标识符的概念

标识符就是用来标识变量名、符号常量名、函数名、数组名、数据类型名、宏以及文件名等的有效字符序列。简单地说,标识符就是一个名字。

2. 定义规则

(1) 标识符只能由字母、数字和下画线 3 种字符组成,且第一个字符必须为字母或下画线。变量名、函数名等用小写字母标识,而符号常量全用大写字母表示,函数名和外部变量名由小于 6 个字符的字符串组成,系统变量由下画线起头构成。

(2) 在 C 语言中,大小写字母不等效。因此,a 和 A、I 和 i、Sum 和 sum 分别是两个不同的标识符。

(3) 用户自定义的标识符不能与 C 语言的保留字(关键字)同名,也不能与 C 语言库函数同名。

(4) 标识符应当直观且可以拼读,让读者看了就能了解其用途。标识符最好采用英文单词或其组合,不要太复杂,且用词要准确,便于记忆和阅读。切忌使用汉语拼音来命名。

(5) 在 C 语言中,标识符的长度可以是一个或多个字符,但是只有前面 32 个字符有效,即系统能识别的标识符最大长度为 32。

有的系统取 8 个字符,假如程序中出现的变量名长度大于 8 个字符,则只有前面 8 个字符有效,后面的不被识别。例如,下面的 3 个变量名将被当作同一个标识符处理: student_name、student_number、student_sex。因此,为了程序的可移植性及阅读程序的方便,建议标识符的长度不要超过 8 个字符。

(6) 尽量避免名字中出现数字编号,如 value1、value2 等,除非逻辑上需要编号。例如,下面为合法的标识符: year、month、student_name、sum2_3、PRICE;下面为不合法的标识符: M.D.Jones、#a、\$432、3b、?c、a=b。

1.2.3 保留字

保留字又称为关键字,就是具有特定含义的标识符,用户不能用作自定义标识符。C 语

言提供的标准保留字有 32 个,非标准保留字有 27 个,另外还有特定字。

1. 标准保留字

(1) 与数据类型有关的标准保留字有 14 个,包括 char、int、float、double、signed、unsigned、short、long、void、struct、union、typedef、enum、sizeof。

(2) 与存储类别有关的标准保留字有 4 个,包括 auto、extern、register、static。

(3) 与程序控制结构有关的标准保留字有 12 个,包括 do、while、for、if、else、switch、case、default、goto、continue、break、return。

(4) 其他的标准保留字有 2 个,包括 const、volatile。

2. 非标准保留字

非标准保留字有 27 个,包括 asm、-AX、-AH、-AL、cdecl、-BX、-BH、-BL、near、-CX、-CH、-CL、far、-DX、-DH、-DL、huge、-BP、-SP、-DI、interrupt、-SI、-cs、-ds、pascal、-es、-ss。

3. 特定字

特定字有 7 个,包括 #include、#define、#ifdef、#ifndef、#undef、#endif、#elif。

特定字用在预处理语句中,虽不是保留字,但人们习惯将它们看成保留字,并赋予了特定的含义。

1.3 设计简单的 C 程序

一个完整的计算机程序通常具备三个功能,即输入数据、数据运算和输出结果。在 C 语言中,数据运算主要是由赋值语句完成的,而 C 语言本身不提供输入/输出语句,在 C 语言中,数据的输入与输出则需要调用 C 编译系统提供的 scanf()函数与 printf()函数来实现。在系统学习 C 语言程序设计之前,本节先简要介绍赋值语句、scanf()函数和 printf()函数的使用,以便能进行简单的程序设计,方便以后各章内容的叙述。

1.3.1 赋值语句

1. 赋值运算符

赋值符号“=”就是赋值运算符,它的作用是将一个数据赋给一个变量。

例如:

a=5,其作用是执行一次赋值操作(或称赋值运算),将常量 5 赋给变量 a。

x=a/b,将表达式 a/b 的值赋给变量 x。

2. 赋值表达式

由赋值运算符连接起来的式子叫作赋值表达式。其一般形式是:

左值表达式=右值表达式

左值表达式必须是变量,右边可以是表达式,赋值表达式的操作是先计算赋值号“=”右边表达式的值,再赋给左边的变量,即将数值存入变量的存储单元中。

例如:

```
a=b=c=10 等价于 a=(b=(c=10))
```

即先将常量 10 赋给变量 c,然后再将 10 赋给变量 b,最后将 10 赋给变量 a。

3. 赋值语句的格式

赋值表达式后加上“;”就构成了赋值语句。

赋值语句的格式是:

左值表达式=右值表达式;

例如:

```
x=10;  
x=2 * y+3;
```

都是合法语句。

```
3=x-2 * y;
```

是不合法语句。因为 3 是常数,不是变量。

C 语言允许在同一个语句中进行反复赋值。

例如:

```
a=b=c=1;
```

其操作是将 1 赋值给变量 c,将变量 c 的值 1 赋值给变量 b,再将变量 b 的值 1 赋值给变量 a。其作用等同于“a=1;b=1;c=1;”。这在有些语言中是不允许的。

1.3.2 printf()函数

1. printf()函数的调用格式

printf()函数的调用格式是:

```
printf(格式控制,输出表列);
```

printf()函数的主要功能是按“格式控制”所指定的格式,从标准输出设备上输出“输出表列”中列出的各输出项。在 printf()函数结尾加上“;”就构成了格式输出语句。

2. printf()函数的格式说明

(1) “格式控制”是由双引号括起来的字符串,包括格式说明和普通字符。格式说明由 % 和其后的格式字符组成,用来指定输出数据的输出格式。不同类型的数据需要不同的格式说明符。表 1-1 列出了 printf()函数常用格式字符。

表 1-1 printf()函数常用格式字符

格式字符	输出类型	说 明
d	整型数据	以带符号的十进制形式输出整数(正数符号不输出)
o	整型数据	以八进制无符号形式输出整数(不输出前导符 0)
x	整型数据	以十六进制无符号形式输出整数(不输出前导符 0x 或 0X)
u	整型数据	以带符号的十进制形式输出整数
c	字符型数据	以字符形式输出,只输出一个字符
s	字符型数据	以字符串形式输出
f	实型数据	以小数形式输出单、双精度数,隐含输出 6 位小数
e	实型数据	以标准指数形式输出单、双精度数,数字部分小数位数为 6 位

(2) “输出表列”由若干个变量或表达式组成,之间用逗号“,”隔开。

例如:

```
printf("a=%d,b=%d",a,b);
```

【例 1-2】 printf()函数的使用。

```
1  #include <stdio.h>
2  void main()
3  {
4      int a,b,c,d;
5      a=10;
6      b=20;
7      c=50;
8      d=25;
9      printf("a=%d,b=%d,c=%d,d=%d\n",a,b,c,d);
10     printf("a+b=%d,a-b=%d\n",a+b,a-b);
11     printf("a * b=%d,c/d=%d\n",a * b,c/d);
12 }
```

说明:

(1) 程序第 4 行是声明部分,定义 a、b、c 和 d 四个整型变量。

(2) 程序第 5~8 行是赋值语句,分别将常量 10、20、50 和 25 赋给变量 a、b、c 和 d。

(3) 程序第 9~11 行是格式输出语句。

运行结果为

```
a=10,b=20,c=50,d=25
a+b=30,a-b=-10
a * b=200,c/d=2
```

1.3.3 scanf()函数

1. scanf()函数的调用格式

scanf()函数的调用格式是:

```
scanf(格式控制,地址表列);
```

scanf()函数的主要功能是按所指定的格式从标准输入设备读入数据,并将数据存入地址表列所指定的存储单元中。在scanf()函数结尾加上“;”就构成了格式输入语句。

2. scanf()函数的格式说明

(1) “格式控制”是由双引号括起来的字符串,仅包括格式说明部分,格式说明由“%”和类型说明符组成,用于指定输入数据的类型。scanf()函数用到的格式字符如表 1-2 所示。

表 1-2 scanf()函数格式字符

格式字符	输入类型	说 明
d	整型数据	输入十进制整数
o	整型数据	输入八进制整数
x	整型数据	输入十六进制整数
u	整型数据	输入无符号十进制整数
c	字符型数据	输入单个字符
s	字符型数据	输入字符串
f	实型数据	输入实数,可用小数形式或指数形式输入
e	实型数据	与 f 作用相同

(2) “地址表列”是由若干个变量的地址组成,就是在变量名前加“&”,当变量地址有多个时,各变量地址之间用逗号“,”隔开。“地址表列”中的地址个数必须与格式参数的个数相同,并且依次匹配。例如:

```
scanf("%d%d%d", &a, &b, &c)
```

当从键盘输入数据时,输入的数值数据之间用间隔符(空格符、制表符或回车符)隔开,间隔符数量不限。最后一定要按 Enter 键,scanf()函数才能接收从键盘输入的数据。

1.3.4 库函数和头文件

前面介绍的 printf()函数和 scanf()函数是 C 编译系统提供的库函数。库函数不是 C 语言的组成部分,而是由 C 编译系统提供一些非常有用的功能函数,例如,各种输入/输出函数、数学函数、字符串处理函数等,可供用户在自己程序中直接调用。这些库函数

的说明、类型和宏定义都分门别类地保存在相应的“头文件”(也称包含文件或标题文件)中。在C语言的编译系统中,提供了若干个“头文件”。“头文件”以.h为扩展名,且为文本文件。当使用系统提供的库函数时,只需在程序开始加上预处理语句: #include <头文件> 或者 #include "头文件"。这条语句指明用#include命令将“头文件”中的内容包含到用户源程序中,使之成为源程序的一部分。

printf()函数和scanf()函数是在头文件stdio.h中定义的,stdio是standard input & output的缩写,h是head的缩写,因此,在调用它们之前,只要在程序开始加上预处理语句#include <stdio.h>或者#include "stdio.h",然后就可以在程序中使用scanf()函数和printf()函数实现输入/输出功能。

C语言有丰富的库函数,应该尽量利用。

【例 1-3】 计算任意立方体的表面积和体积。

设一个立方体的三条边分别是 a 、 b 、 c ,表面积为 s ,体积为 v ,程序代码如下:

```
1  #include <stdio.h>
2  void main()
3  {
4      int a,b,c,s,v;           /* 定义三条边为 a、b 和 c,表面积为 s,体积为 v */
5      scanf("%d%d%d",&a,&b,&c); /* 输入三条边的长度 */
6      s=2*(a*b+b*c+a*c);      /* 计算表面积 */
7      v=a*b*c;                /* 计算体积 */
8      printf("s=%d\n",s);     /* 输出表面积 */
9      printf("v=%d\n",v);     /* 输出体积 */
10 }
```

说明:

(1) 程序第4行代码是声明部分,定义三条边为 a 、 b 和 c ,表面积为 s ,体积为 v ,均为整型变量。

(2) 程序第5行代码,当运行到“scanf(“%d%d%d”,&a,&b,&c);”时,系统停下等待用户将整型数据从键盘输入给变量 a 、 b 、 c 。其中,&a、&b、&c中的&是地址运算符,&a的含义是变量 a 在内存中的地址。scanf()函数按照 a 、 b 、 c 在内存中的地址&a、&b、&c给 a 、 b 、 c 赋值。

scanf()函数中的“%d%d%d”表示按十进制整数形式输入数据。从键盘输入时,两个数据之间用一个或多个空格间隔,也可用Enter键、Tab键间隔。不能用逗号间隔两个数据。

(3) 程序第6行是赋值语句,将表面积的值赋值给变量 s 。

(4) 程序第7行是赋值语句,将体积的值赋值给变量 v 。

(5) 程序第8行是格式输出语句,printf()函数括号中双引号括起来的“s=”按原样输出,%d表示“以十进制整数类型”输出变量 s 的值,printf()函数括号内逗号右边 s 是要输出的变量。

(6) 程序第9行是格式输出语句,输出变量 v 的值。