

高等学校计算机基础教育规划教材

# C程序设计 教程与实验

(第2版)

吉顺如 主 编  
陶 恂 曾祥绪 副主编

清华大学出版社



高等学校计算机基础教育规划教材

# C程序设计 教程与实验

(第2版)

吉顺如 主 编  
陶 恂 曾祥绪 副主编

清华大学出版社  
北京

## 内 容 简 介

本书在内容编排上力求重点突出、难点分散,在语言描述上注重概念清晰、通俗易懂,并通过大量的例题分析将理论知识与实践相结合,以期逐步提高学生编写程序的能力。

全书共分10章,内容包括C语言程序设计概述、顺序结构程序设计、选择结构程序设计、循环结构程序设计、数组、函数、指针、结构体与共用体、文件以及单片机C语言编程实例。本书每章均有精心设计的难易适当的习题供学生课后练习,每章的上机实验题包括选择题、填空题及编程题,可以让学生在反复实践中提高设计程序和调试程序的能力。

本书可作为高等院校理工科各专业“高级语言程序设计”课程的教材,也可供对程序设计有兴趣的读者参考。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

### 图书在版编目(CIP)数据

C 程序设计教程与实验/吉顺如主编.—2版.—北京:清华大学出版社,2017  
(高等学校计算机基础教育规划教材)

ISBN 978-7-302-46822-6

I. ①C… II. ①吉… III. ①C语言—程序设计—高等学校—教材 IV. ①TP312.8

中国版本图书馆 CIP 数据核字(2017)第 053153 号

责任编辑:刘翰鹏

封面设计:何凤霞

责任校对:赵琳爽

责任印制:李红英

出版发行:清华大学出版社

网 址:<http://www.tup.com.cn>, <http://www.wqbook.com>

地 址:北京清华大学学研大厦 A 座 邮 编:100084

社 总 机:010-62770175 邮 购:010-62786544

投稿与读者服务:010-62776969, [c-service@tup.tsinghua.edu.cn](mailto:c-service@tup.tsinghua.edu.cn)

质量反馈:010-62772015, [zhiliang@tup.tsinghua.edu.cn](mailto:zhiliang@tup.tsinghua.edu.cn)

课件下载:<http://www.tup.com.cn>, 010-62770175-4278

印 装 者:清华大学印刷厂

经 销:全国新华书店

开 本:185mm×260mm 印 张:20.5 字 数:467千字

版 次:2011年6月第1版 2017年5月第2版 印 次:2017年5月第1次印刷

印 数:1~2000

定 价:48.00元

产品编号:073466-01

C语言是国内外广泛使用的计算机程序设计语言,是高等院校理工科相关专业重要的专业基础课程。C语言功能丰富、使用灵活方便,程序执行效率高,可移植性好,既可以用来编写系统程序,又可以用来编写应用程序,因而越来越受到人们的欢迎。

本书是根据教育部《关于进一步加强高等学校计算机基础教学的意见》的教学基本要求和高等院校计算机基础教学改革的需要,结合作者多年讲授C语言程序设计课程的教学经验编写而成。

本书的教学目标是培养学生的逻辑思维能力和程序设计能力,因此在内容编排上力求重点突出、难点分散,在语言描述上注重概念清晰、通俗易懂,并通过大量的例题分析将理论知识与实践相结合,以期逐步提高学生编写程序的能力。

本书是在第1版的基础上将有些内容进行了整合。从原来的12章改成10章,内容安排如下。

第1章主要介绍C程序的基本结构、C程序的基本要素以及C程序的集成开发环境。

第2章主要介绍运算符和表达式、C语言的基本语句、输入与输出函数以及算法的概念。

第3章主要介绍关系运算符、逻辑运算符以及选择结构控制语句。

第4章主要介绍循环控制语句、循环的嵌套。

第5章主要介绍一维数组、二维数组和字符串。

第6章主要介绍结构化与模块化程序设计思想、函数的定义与调用、变量的作用域和存储类别。

第7章主要介绍指针的概念、指针变量的定义以及指针与数组、指针与函数的关系。

第8章主要介绍结构体、链表、共用体的定义及应用。

第9章主要介绍数据文件的基本操作函数和使用规则。

第10章通过一款智能充电机的编程实例,介绍C语言在单片机编程中的具体应用。

由于C语言程序设计是一门理论性、实践性均较强的课程,为了帮助学生掌握有关的基本概念和程序设计方法,每章后面均精心设计了难易适当、题型丰富的习题供学生课后练习。每章的实验题均包含选择题、填空题和编程题,可以使学生在反复实践中提高设计程序和调试程序的能力。

本书有对应的课程网站和微信订阅号,可以观看部分教学视频、算法动画、学习中会遇到的常见问题解答和计算机二级考试的相关资料。网址为 <http://cprogram.sdju.edu.cn>, 微信订阅号的 ID 为 ProgramDesign。

本书由上海电机学院的吉顺如主编,计春雷主审。第 2 版编写分工:第 1 章由吉顺如、任远编写;第 2、5、9 章及附录由吉顺如编写;第 3 章由陶恂编写;第 4 章由张艳编写;第 6 章由任远编写;第 7、8 章由王中华、吉顺如编写;第 10 章由曾祥绪编写。全书由吉顺如统稿。

在本书的编写过程中得到许多教师的帮助,在此表示诚挚的谢意。由于编者水平有限,书中不足之处在所难免,恳请读者批评指正。

编 者

2017 年 3 月

# 目 录

Contents

第 1 章 C 语言程序设计概述 .....	1
1.1 C 程序简介 .....	1
1.2 C 程序的结构与书写格式 .....	3
1.3 C 语言的特点 .....	4
1.4 C 程序的开发过程 .....	5
1.5 C 程序的基本要素 .....	6
1.5.1 标识符 .....	6
1.5.2 数据类型、常量和变量 .....	7
1.6 C 程序的集成开发环境 .....	11
1.6.1 Dev-C++ .....	11
1.6.2 Visual Studio 2015 .....	17
1.7 习题 .....	27
1.8 实验 1 熟悉 C 程序编程环境 .....	28
第 2 章 顺序结构程序设计 .....	31
2.1 运算符和表达式 .....	31
2.1.1 算术运算符和算术表达式 .....	31
2.1.2 赋值运算符和赋值表达式 .....	32
2.1.3 逗号运算符和逗号表达式 .....	33
2.1.4 强制类型转换运算符 .....	34
2.1.5 sizeof 运算符 .....	34
2.1.6 不同数据类型的数据间的混合运算 .....	35
2.1.7 自增、自减运算符 .....	36
2.1.8 位运算符 .....	36
2.2 C 语言的基本语句 .....	37
2.3 输入与输出函数 .....	38
2.3.1 格式输入/输出函数 .....	38
2.3.2 字符输入/输出函数 .....	43
2.4 算法简介 .....	44
2.4.1 算法的概念 .....	44

2.4.2	算法的特性 .....	45
2.4.3	算法的表示 .....	46
2.5	应用举例 .....	48
2.6	习题 .....	51
2.7	实验 2 顺序结构程序设计 .....	53
<b>第 3 章</b>	<b>选择结构程序设计 .....</b>	<b>57</b>
3.1	关系运算符和逻辑运算符 .....	57
3.1.1	关系运算符 .....	57
3.1.2	逻辑运算符 .....	58
3.2	选择结构控制语句 .....	59
3.2.1	if 语句 .....	59
3.2.2	条件运算符 .....	62
3.2.3	switch 语句 .....	63
3.3	应用举例 .....	65
3.4	习题 .....	71
3.5	实验 3 选择结构程序设计 .....	75
<b>第 4 章</b>	<b>循环结构程序设计 .....</b>	<b>79</b>
4.1	循环结构控制语句 .....	79
4.1.1	while 循环语句 .....	79
4.1.2	do...while 循环语句 .....	80
4.1.3	for 循环语句 .....	82
4.1.4	break 语句和 continue 语句 .....	83
4.2	循环的嵌套 .....	85
4.3	应用举例 .....	86
4.4	习题 .....	90
4.5	实验 4 循环结构程序设计 .....	95
<b>第 5 章</b>	<b>数组 .....</b>	<b>99</b>
5.1	一维数组 .....	99
5.1.1	一维数组的定义和引用 .....	99
5.1.2	一维数组的初始化 .....	100
5.1.3	一维数组的应用举例 .....	101
5.2	二维数组 .....	104
5.2.1	二维数组的定义和引用 .....	104
5.2.2	二维数组的初始化 .....	105
5.2.3	二维数组的应用举例 .....	106

5.3 字符串 .....	109
5.3.1 字符串与字符数组 .....	109
5.3.2 字符串处理函数 .....	111
5.4 应用举例 .....	115
5.5 习题 .....	124
5.6 实验 5 数组程序设计 .....	128
<b>第 6 章 函数</b> .....	<b>132</b>
6.1 结构化与模块化程序设计思想 .....	132
6.2 函数的定义与调用 .....	134
6.2.1 函数的定义 .....	134
6.2.2 函数的声明 .....	135
6.2.3 函数的调用 .....	136
6.2.4 函数的参数和返回值 .....	138
6.2.5 数组名作函数参数 .....	139
6.3 递归函数 .....	141
6.4 变量的作用域和存储类别 .....	144
6.4.1 变量的作用域 .....	144
6.4.2 变量的存储类别 .....	146
6.5 函数与带参数的宏 .....	148
6.6 应用举例 .....	149
6.7 习题 .....	154
6.8 实验 6 函数程序设计 .....	157
<b>第 7 章 指针</b> .....	<b>163</b>
7.1 指针概述 .....	163
7.1.1 指针的概念 .....	163
7.1.2 指针变量的定义 .....	165
7.1.3 指针变量的引用 .....	165
7.2 指针与数组 .....	169
7.2.1 通过指针引用数组元素 .....	170
7.2.2 指针与数组名 .....	174
7.3 指针与字符串 .....	176
7.4 指针与函数 .....	178
7.4.1 指针作为函数参数 .....	178
7.4.2 返回指针值的函数 .....	182
7.4.3 指向函数的指针 .....	184
7.5 多级指针 .....	186



7.5.1	指向指针的指针	187
7.5.2	指针数组	188
7.5.3	指向多维数组行的指针	193
7.6	应用举例	196
7.7	指针小结	198
7.8	习题	199
7.9	实验 7 指针程序设计	204
<b>第 8 章 结构体与共用体</b>		<b>208</b>
8.1	结构体概述	208
8.1.1	结构体变量的定义和初始化	208
8.1.2	结构体成员的引用	210
8.2	结构体数组	212
8.3	指向结构体的指针	214
8.4	结构体与函数	218
8.5	链表	219
8.5.1	动态内存管理	220
8.5.2	链表概述	222
8.5.3	链表的相关操作	223
8.6	共用体概述	230
8.6.1	共用体变量的定义和初始化	230
8.6.2	共用体变量的引用	231
8.7	类型定义符 typedef	233
8.8	应用举例	236
8.9	习题	239
8.10	实验 8 结构体与链表程序设计	245
<b>第 9 章 文件</b>		<b>250</b>
9.1	文件概述	250
9.2	常用文件操作函数	251
9.2.1	文件的打开与关闭	251
9.2.2	文件的读/写	253
9.2.3	文件的定位	260
9.2.4	文件读/写中的检测	262
9.3	应用举例	262
9.4	习题	268
9.5	实验 9 文件程序设计	272

第 10 章 单片机 C 语言编程实例 .....	277
10.1 功能要求 .....	277
10.2 硬件线路 .....	277
10.3 编程分解 .....	279
10.3.1 充电控制 .....	279
10.3.2 时钟中断 .....	280
10.3.3 端口操作 .....	281
10.4 单片机 C 语言编程小结 .....	282
10.5 智能充电机完整源代码 .....	283
附录 .....	295
附录 A 常用字符与 ASCII 代码对照表 .....	295
附录 B 编译预处理指令 .....	296
附录 C 运算符和结合性 .....	301
附录 D C 库函数 .....	302
附录 E C 程序中常见错误分析 .....	307
参考文献 .....	315

# C 语言程序设计概述

计算机处理问题是由程序来控制的。程序是人们根据解决问题的思路,利用计算机程序设计语言编制的能完成一定功能的指令序列。程序设计是指用计算机程序设计语言编制计算机程序的过程。

C 语言是一种计算机程序设计语言,起源于 20 世纪 70 年代,最初用于编写 UNIX 操作系统,后来由于 C 语言强大的功能及可移植性,使 C 语言迅速得到推广,并成为世界上应用最为广泛的程序设计语言。

本章通过几个简单的例子来认识 C 语言,了解 C 语言的结构特点、书写格式以及在 Dev-C++ 集成环境和 Visual Studio 2015 集成环境下调试 C 语言程序的方法。

## 1.1 C 程序简介

为了说明 C 程序的结构特点以及书写格式,下面通过几个例子来认识一下 C 程序。

**【例 1-1】** 在屏幕上输出一行信息。

```
#include <stdio.h>                /* 编译预处理命令 */
int main(void)                    /* 主函数 */
{                                  /* 函数体开始 */
    printf("This is my first C Program!\n"); //在屏幕上输出信息
    return 0;
}                                  /* 函数体结束 */
```

运行结果为:

```
This is my first C Program!
```

程序说明:

(1) 程序的第 1 行是 C 语言中以 # 开头的编译预处理命令,以 .h 为扩展名的文件称为头文件,通过文件包含命令 #include 将头文件 stdio.h 包含进自己编写的 C 程序中。所谓头文件是系统内置的已经编写好的程序,用户通过文件包含命令实现对头文件的调用,头文件用尖括号 <> 或双引号 " " 括起来。此处的 stdio.h 是标准的输入/输出函数头文件。若要在程序中实现输入/输出,需使用系统内置的库函数,如该程序中的 printf() 函数,则必须在程序的开始处写上预处理命令 #include <stdio.h> 或 #include "stdio.h", 详见附录 B。

(2) 程序的第 2~6 行是 C 程序主函数 main() 的定义。main 前面的 int 表明该函数

将返回一个整数值。圆括号中的 void 表明 main 函数没有参数。在 C 语言程序中,必须有而且只能有 1 个 main() 主函数,C 程序的执行总是从 main() 函数开始。从 { 开始,到 } 结束的部分称为函数体。第 4 行 printf() 函数的功能是在屏幕上输出信息,双引号内的 \n 表示换行,即信息在屏幕上输出后,光标定位到下一行。最后的分号“;”是 C 程序语句结束的标志。

(3) 在 C 语言中,以 /\* 开头、\*/ 结束的内容或以 // 开头的内容是程序的注释。注释是对代码的“提示”,注释可以出现在程序的任何位置,用以帮助理解程序。运行程序时,注释部分将不被执行。一般地,程序块的注释采用 /\* ... \*/ ,行注释采用 “//...”。

(4) 函数体中最后一条语句“return 0;”有两个作用:一是使 main 函数终止(从而结束程序);二是指出 main 函数的返回值是 0,这个值表明程序正常终止。如果 main 函数的末尾没有 return 语句,程序仍然能终止,但是,许多编译器会产生一条警告信息。

**【例 1-2】** 从键盘输入两个整数,求它们的乘积。

```
#include <stdio.h>
int main(void)
{
    int a,b,cj;                //定义 3 个整型变量
    printf("Please Input Two Integers:\n"); //在屏幕上输出提示信息
    scanf("a=%d,b=%d",&a,&b); //从键盘输入两个整数分别放入变量 a 和 b 中
    cj=a * b;                 //将 a 和 b 的乘积赋给变量 cj
    printf("cj=%d\n",cj);    //在屏幕上输出乘积 cj 的值
    return 0;
}
```

运行结果为:

```
Please Input Two Integers:
a=7,b=9
cj=63
```

**说明:**

(1) 程序的第 4 行是变量的定义语句。C 语言用变量来存放数据,此处用关键词 int 定义 3 个整型变量  $a$ 、 $b$ 、 $cj$ ,表示这 3 个变量中可以存放整数。在定义变量时,多个变量之间以逗号“,”分隔。

(2) 程序的第 6 行 scanf() 函数的功能是从键盘输入两个整数分别放入变量  $a$  和  $b$  中,其中 %d 是格式控制符,表示从键盘输入数据的类型是十进制整数;& 是取地址符,表示从键盘输入的数放到 & 符号后面变量所对应的存储地址中。假如从键盘输入的两个数是 7 和 9,则此处 scanf() 函数的输入方式为:  $a=7, b=9$ 。注意:双引号中普通字符原样输入。

(3) 程序的第 7 行是赋值语句,“=”是赋值运算符,其功能是把“=”右边表达式  $a * b$  的乘积值赋给左边的变量  $cj$ 。

(4) 程序的第 8 行功能是在屏幕上输出乘积  $cj$  的值,其输出格式是双引号中的普通字符原样输出。%d 表示输出数据的类型是十进制整数,乘积  $cj$  的值对应输出在 %d 的

位置。

**【例 1-3】** 从键盘输入两个实数,比较它们的大小,在屏幕上输出其中较小的数。

```
#include <stdio.h>
float min1(float x,float y);           //函数声明语句
int main(void)
{
    float a,b,min;                     //定义 3 个 float 实数类型的变量
    printf("Please Input a,b: ");      //输出提示信息
    scanf("%f,%f",&a,&b);             //从键盘输入两个数分别存入变量 a 和 b
    min=min1(a,b);                     //调用函数 min1(),并将返回值赋给变量 min
    printf("min=%f\n",min);           //输出结果
    return 0;
}
float min1(float x,float y)           //定义函数 min1()
{
    float z;                            //定义变量 z 是 float 实数类型
    if(x<y)                              //条件判断语句,判断 x 是否小于 y
        z=x;                             //如果 x<y,则执行该行赋值语句
    else                                  //否则,即 x 不小于 y,则执行下一行赋值语句
        z=y;
    return(z);                           //返回 z 的值
}
```

运行结果为:

```
Please Input a,b: 1.2,5.4 ↵
min=1.200000
```

说明:

(1) 本程序中定义了 main()和 min1()两个函数。其中 main()函数是 C 语言程序必不可少的,min1()函数是用户根据功能需求自己定义的,称为用户自定义函数。这两个函数是相互平行的,它们可以通过调用发生联系。所有被调用的函数都必须先定义后使用,main()主函数不可以被其他函数调用。

(2) 程序的第 2 行是一条函数声明语句,当被调用函数写在主调函数的后面时,必须对被调用函数进行声明,以使系统在编译时识别。

(3) 程序的第 7 行 scanf()函数和第 9 行 printf()函数中的“%f”是格式控制符,表示输入和输出数据的类型是 float 类型,即实型,也称为单精度浮点型。

(4) 程序的第 12~20 行是用户自定义函数 min1(),其功能是比较两个实数的大小,并将其中较小的数返回调用该函数的语句。

## 1.2 C 程序的结构与书写格式

通过 1.1 节中的 3 个例子,可以看出 C 语言程序的结构和书写格式如下。

### 1. C 程序的结构

(1) C 程序由函数构成。一个 C 源程序可以由一个 main()主函数和若干个用户自

定义函数构成,其中 `main()` 主函数必须有而且只能有一个,且 `main()` 函数以“`return 0;`”语句结束。函数是 C 程序的基本单位。

(2) 函数由函数头部和函数体两部分组成。如例 1-3 中的用户自定义函数 `min1()`,其中 `float min1(float x, float y)` 是函数首部,包括函数名 `min1`、函数类型 `float`、参数类型 `float` 和参数 `x` 和 `y`。注意:函数名后面的一对圆括号不可少。此函数中各项的含义为:

float	min1	(float	x,	float	y)
↓	↓	↓	↓	↓	↓
函数类型	函数名	参数类型	参数名	参数类型	参数名

函数体是函数首部后面一对大括号 `{}` 内的内容。函数体一般由两部分组成:对所使用的变量进行定义的说明部分和完成各种操作的执行部分。

(3) C 程序的执行总是从 `main()` 函数开始,并到 `main()` 函数结束。`main()` 函数在整个程序中的位置可以任意。

(4) C 程序的语句以分号“`;`”结束。

(5) C 程序中包含注释,以方便理解程序,注释不参与程序的执行。

## 2. C 程序的书写格式

为便于阅读和理解,C 程序的书写一般遵循以下规则。

(1) 一个语句占一行。

用 C 语言书写程序时较为自由,既可以一行写一个语句,也可以一行写多个语句,还可以一个语句分多行来写,但为了清晰起见,建议一个语句占一行。

(2) 英文字母严格区分大小写。

C 程序中英文字母严格区分大小写,一般书写 C 程序时使用小写字母。C 语言规定了 32 个有特定意义的单词,称为关键字,这些关键字在使用时必须是小写字母。

(3) 采用缩进格式的书写方法。

为了看清 C 程序的层次结构,便于阅读和理解程序,C 程序一般都采用缩进格式的书写方法。缩进格式要求在书写程序时,不同结构层次的语句,从不同的起始位置开始,同一结构层次中的语句,缩进同样个数的字符位置。

(4) 为了便于阅读和理解程序,在程序中适当添加注释信息。

在编写程序时应力求遵循以上书写规则,以养成良好的编程习惯。

## 1.3 C 语言的特点

计算机程序设计语言一般可以分为机器语言、汇编语言和高级语言。

机器语言是计算机能够直接识别的语言,其操作指令只能由二进制代码 0 和 1 构成。汇编语言是为方便记忆和编写程序,用一些助记符表示二进制代码,是一种与机器语言对应的符号化的语言,汇编语言编写的程序不能被计算机识别,必须通过专门的汇编程序将符号转换成二进制代码才能执行。高级语言是 20 世纪 50 年代发展起来使用人们习惯的自然语言编写程序的计算机语言,高级语言编写的程序计算机也不能直接识别和执行,必须通过专门的编译程序转换成机器语言才能执行,其转换的方式有两种:一种是解释方

式,即将高级语言编写的程序翻译一句执行一句;另一种是编译方式,即将高级语言编写的程序文件全部翻译成机器语言,生成可执行文件后再执行。C语言采用的是编译方式。

C语言是一种介于汇编语言和高级语言之间的程序设计语言,它有以下特点。

#### (1) 程序结构简洁紧凑

C程序由若干函数构成,各函数是相互独立的,它们通过调用发生联系,C语言是一种模块化程序设计语言。程序书写形式自由。

#### (2) 表达能力强且应用灵活

C语言运算符丰富,共有34种,可以组成各种类型的表达式以提高运算效率。C语言数据类型丰富,包括整型、实型、字符型、数组类型、指针类型、结构体类型、共用体类型等,能实现各种数据结构的运算,从而可以适应不同的功能需求。

#### (3) 生成的目标程序质量好,执行效率高

C语言具有汇编语言的许多特性,允许直接访问物理地址,能进行位操作,可以直接对硬件进行操作。用C语言编写的程序经编译后生成的可执行代码比用汇编语言编写的代码执行效率仅低10%~20%。

#### (4) C程序可移植性好

C语言通过调用标准输入/输出库函数实现输入/输出功能,因此C语言不依赖于计算机硬件系统,从而便于在不同的计算机之间实现程序的移植。

由于C语言具有上述众多特点,已经成为程序设计的主要语言之一,被广泛应用于计算机的系统软件和应用软件的开发。

## 1.4 C程序的开发过程

一个C程序从编写到得到最终结果,其开发过程如图1-1所示。

### 1. 编辑源程序

用高级语言编写的程序称为源程序。将C语言编写的源程序输入编辑器,并保存为文件,后缀名为.c。

### 2. 编译程序

编译程序就是将C源程序转换成机器语言程序,编译的作用是对源程序进行词法检查和语法检查,如果没有错误,则生成目标程序,文件后缀名是.obj;如果存在错误,则编译系统给出的出错信息分为两种,一种是错误(error),一种是警告(warning)。凡是检查出error类的错误,就不能生成目标程序,必须改正后重新编译。

编译中没有出现错误,只能说明程序中没有词法和语法错误。

### 3. 连接程序

使用系统的“连接程序”将目标文件与系统的库文件和系统提供的其他信息连接起来,生成可执行的二进制

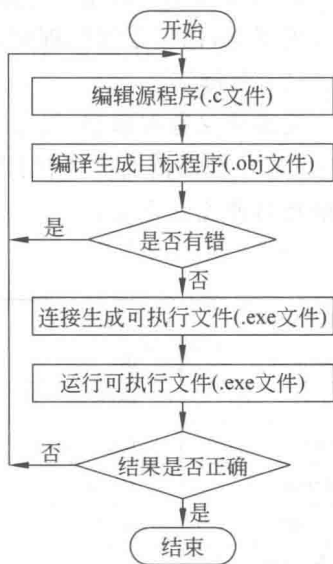


图1-1 C程序的开发过程

文件,后缀名是 .exe。

#### 4. 运行程序

运行 .exe 可执行文件,得到运行结果。若运行结果不正确,则需检查并修改源程序,重复上述步骤,直到得到正确的运行结果为止。

## 1.5 C 程序的基本要素

### 1.5.1 标识符

程序中变量、类型、函数和标号等的名称统称为标识符。在 C 语言中,有 2 类标识符。

#### (1) 用户自定义标识符

由用户根据需要自行定义的标识符,通常用作函数名、变量名等,如自定义函数名 Add,变量名  $x$ 、 $y$ 、 $sum$  等。

标识符的命名必须遵循一定的规则。C 语言规定,标识符只能由字母(A~Z, a~z)、数字(0~9)和下划线(\_)组成,并且其第一个字符必须是字母或下划线。以下标识符是合法的:

Average, x, x3, BOOK\_1, sum5, \_123

以下标识符是非法的,不能作为变量名或函数名:

6s(以数字开头),s \* T(出现非法字符 \*),bowy-1(出现非法字符-(减号))

C 语言是区分大小写的,即在标识符中,大小写字母是有区别的。例如 BOOK 和 book 是两个不同的标识符。

标识符虽然可由用户随意定义,但它是用于标识某个量的符号,因此,命名应尽量有相应的意义,以便于阅读理解,做到“见名知意”。

#### (2) 关键字

关键字又称保留字,是系统已有的标识符,表 1-1 中列出的关键字(C99 新增 5 个关键字)对 C 编译器而言都有着特殊的意义,用户只能按其预先规定的含义来使用它们,而不能擅自改变其含义。

表 1-1 关键字

auto	break	case	char	const	continue
default	do	double	else	enum	extern
float	for	goto	if	inline <sup>①</sup>	int
long	register	restrict <sup>①</sup>	return	short	signed
sizeof	static	struct	switch	typedef	union
unsigned	void	volatile	while	_Bool <sup>①</sup>	_Complex <sup>①</sup>
_Imaginary <sup>①</sup>					

注:① 仅 C99 有。



## 1.5.2 数据类型、常量和变量

在C语言中,数据都是有类型的,即通过类型说明数据的种类。数据有常量和变量之分。

### 1. 数据类型

C语言提供了丰富的数据类型,基本的数据类型有:整型、实型(浮点型)和字符型。它们都是系统定义的简单数据类型,称为标准数据类型。

在程序中直接给出的数据,计算机可以自动识别数据的类型,但当使用标识符来表示可变化的数据时,就需要考虑该数据变化的范围和精度。表1-2列出了整型数据所占存储空间与范围。

表 1-2 32 位机的整数类型

数据类型	字节数	数值范围
整型 int	4	$-2147483648 \sim 2147483647 (-2^{31} \sim 2^{31} - 1)$
无符号整型 unsigned int	4	$0 \sim 4294967295 (0 \sim 2^{32} - 1)$
短整型 short int	2	$-32768 \sim 32767 (-2^{15} \sim 2^{15} - 1)$
长整型 long int	4	$-2147483648 \sim 2147483647 (-2^{31} \sim 2^{31} - 1)$
无符号长整型 unsigned long int	4	$0 \sim 4294967295 (0 \sim 2^{32} - 1)$

另外,C99标准提供了两个额外的数据类型:long long int 和 unsigned long long int 以满足日益增长的对超大型整数的需求。同时为了支持64位运算的新处理器的能力,这两个long long 类型要求至少64位宽。long long int 的取值范围为: $-9223372036854775808 \sim 9223372036854775807 (-2^{63} \sim 2^{63} - 1)$ ,而 unsigned long long int 的取值范围为: $0 \sim 18446744073709551615 (0 \sim 2^{64} - 1)$ 。

C标准没有说明float,double 和 long double 类型提供的精度到底是多少,因为不同的计算机可以用不同的方法存储浮点数。大多数现代计算机都遵循IEEE754标准的规范。表1-3列出了根据IEEE标准实现时浮点类型数据所占存储空间、范围及精度。

表 1-3 浮点类型(IEEE 标准)

数据类型	字节数	有效位数	数值范围
单精度浮点型 float	4	6~7	$1.17519 * 10^{-38} \sim 3.40282 * 10^{38}$
双精度浮点型 double	8	15~16	$2.22507 * 10^{-308} \sim 1.79769 * 10^{308}$
扩展精度浮点型 long double	10(16)	18~19	约 $10^{-4932} \sim 10^{4932}$

说明:long double 类型的长度随着机器的不同而变化,而最常见的大小是80位和128位。

### 2. 常量

常量是在程序执行过程中其值不能被改变的量。常量有各种不同的数据类型,常量的类型通常由书写格式决定。例如:25是整型常量,25.36是浮点型常量,'A'是字符常量,"ABC"是字符串常量。