

# C语言 程序设计

苑鸿骥 钟文峰 主编  
谭晓芳 张朋 晏志英 副主编

 人民邮电出版社  
POSTS & TELECOM PRESS

# C 语言程序设计

苑鸿骥 钟文峰 主 编  
谭晓芳 张 朋 晏志英 副主编

人民邮电出版社

## 图书在版编目 (CIP) 数据

C 语言程序设计 / 苑鸿骥, 钟文峰主编. —北京: 人民邮电出版社, 2006.8  
ISBN 7-115-15024-9

I. C... II. ①苑...②钟... III. C 语言—程序设计 IV. TP312

中国版本图书馆 CIP 数据核字 (2006) 第 079494 号

### 内 容 提 要

本书在内容组织上根据作者在 C 语言教学过程中遇到的问题和教学经验进行了有效的改进, 突出 C 语言学习的目的, 使得学习和掌握 C 语言更加容易和快捷。本书主要内容包括 C 语言概述, 算法, 数据类型、运算符和表达式, 基本输入/输出和顺序结构程序设计, 选择结构程序设计, 循环结构程序设计、数组, 函数, 预处理命令, 指针, 结构体、共用体、枚举型及文件。在这些内容的讲解过程中, 坚持语言为程序设计服务的原则, 突出实用性。

本书为高等院校 C 语言课程教材, 也可供计算机爱好者自学使用, 同时也可作为从事软件开发、程序设计及计算机教学人员的参考书。

### C 语言程序设计

- 
- ◆ 主 编 苑鸿骥 钟文峰
  - ◆ 副 主 编 谭晓芳 张 朋 晏志英  
责任编辑 赵慧君
  - ◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号  
邮编 100061 电子函件 315@ptpress.com.cn  
网址 <http://www.ptpress.com.cn>  
北京通州大中印刷厂印刷  
新华书店总店北京发行所经销
  - ◆ 开本: 787×1092 1/16  
印张: 15.75  
字数: 378 千字 2006 年 8 月第 1 版  
印数: 1-5 000 册 2006 年 8 月北京第 1 次印刷

ISBN 7-115-15024-9/TP · 5566

定价: 27.00 元

读者服务热线: (010) 67170985 印装质量热线: (010) 67129223

## 编者的话

---

C 语言是一种非常实用且应用广泛的程序设计语言。它的功能强大、使用灵活、可移植性好，既具有高级语言的优点，又具有低级语言的特点；既可用于编写系统软件，也可用于编写应用软件。C 语言的语法规则清晰、便于掌握和记忆，是大多数人学习计算机程序设计的入门语言。

本书在内容组织上根据作者在 C 语言教学过程中遇到的问题和教学经验进行了有效的改进，突出了 C 语言的学习目的，使得学习和掌握 C 语言更加容易和快捷。本书具有以下特点。

(1) 内容结构严谨，由浅入深，循序渐进。本书系统全面地讲述了 C 语言的相关知识，在内容安排上由浅入深，循序渐进，使读者不断有收获，不断提高。在保持学习兴趣的前提下完整掌握 C 语言的相关知识。

(2) 突出 C 语言的学习目的。通过 C 语言的学习，重要的是掌握结构化程序设计的基本方法，逐步形成结构化程序设计的思想。本书通过大量例题，根据教师的教学经验，将 C 语言的典型应用加以讲述，使读者能够真正掌握利用 C 语言编写结构化程序的基本方法，同时提高阅读程序的能力。

(3) 概念准确，讲解细致，文字通俗易懂。本书概念严谨准确，对 C 语言的内容进行了详细的讲解，文字通俗易懂。对难点问题配有图解，帮助读者加深理解。本书便于教师讲授，也便于学生自学，也可以作为计算机等级考试的参考用书。

学习程序设计课程时，应该把精力放在最基本、最常用的内容上，开始时不必在每一个细节上死记硬背，在有一定基础后再深入到一些常见的细节，有一些细节是需要通过较长期的实践才能熟练掌握的。程序设计是一门实践性很强的课程，既要掌握概念，又要动手编程，还要上机调试运行。希望读者一定要重视实践环节，包括编程和上机，把重点放在编写程序和调试程序上。

本书由南昌理工学院苑鸿骥教授组织编写并主编全书，钟文峰协助苑鸿骥教授统稿并审阅全书。其中，晏志英编写了第 1 章、第 3 章和第 9 章，谭晓芳编写了第 2 章、第 5 章和第 8 章，陈黎艳编写了第 4 章和第 11 章，陆志香编写了第 6 章和第 7 章，钟文峰编写了第 10 章和第 12 章及附录。另外，张朋、孙素敏、刘肃平、黄栋等同志对本书提供了帮助，在此一并表示感谢。

由于时间紧迫和编者水平有限，书中错误在所难免，敬请广大读者提出宝贵意见和建议。

编者  
2006 年 7 月

# 目 录

<b>第 1 章 C 语言概述</b> .....	1
1.1 C 语言的发展历史及其基本特性.....	1
1.2 简单的 C 语言程序介绍.....	2
1.3 C 语言程序的上机步骤.....	4
1.4 小结.....	6
习题.....	7
<b>第 2 章 算法</b> .....	8
2.1 算法的概念.....	8
2.2 简单算法举例.....	8
2.3 算法的特性.....	10
2.4 算法的表示.....	11
2.4.1 用自然语言表示算法.....	11
2.4.2 用流程图表示算法.....	11
2.4.3 程序的 3 种基本结构.....	12
2.4.4 用 N-S 流程图表示算法.....	14
2.4.5 用计算机语言表示算法.....	15
2.5 结构化程序设计方法.....	16
2.6 小结.....	16
习题.....	17
<b>第 3 章 数据类型、运算符和表达式</b> .....	18
3.1 C 语言的基本数据类型.....	18
3.1.1 常量和变量.....	18
3.1.2 整型数据.....	20
3.1.3 实型数据.....	22
3.1.4 字符型数据和字符串常量.....	23
3.2 变量赋初值.....	27
3.3 各类数值型数据间的混合运算.....	28
3.4 C 语言的运算符和表达式.....	29
3.4.1 算术运算符和算术表达式.....	29
3.4.2 赋值运算符和赋值表达式.....	33
3.4.3 逗号运算符和逗号表达式.....	35
3.4.4 位运算符.....	36

3.4.5 长度运算符	38
3.5 小结	39
习题	39
<b>第 4 章 基本输入/输出和顺序结构程序设计</b>	<b>42</b>
4.1 C 语句概述	42
4.2 数据输入/输出的概念	43
4.3 字符数据的输入/输出	44
4.3.1 putchar 函数 (字符输出函数)	44
4.3.2 getchar 函数 (字符输入函数)	44
4.4 格式输入/输出	45
4.4.1 printf 函数 (格式输出函数)	45
4.4.2 scanf 函数 (格式输入函数)	49
4.5 常用函数的使用	52
4.5.1 数学函数	52
4.5.2 字符处理函数	53
4.6 顺序结构程序设计举例	54
4.7 小结	57
习题	57
<b>第 5 章 选择结构程序设计</b>	<b>60</b>
5.1 关系运算符和关系表达式	60
5.1.1 关系运算符	60
5.1.2 关系表达式	61
5.2 逻辑运算符和表达式	62
5.2.1 逻辑运算符及其优先次序	62
5.2.2 逻辑表达式	62
5.3 if 语句	65
5.3.1 if 语句的一般形式	65
5.3.2 if 语句的嵌套	69
5.3.3 条件运算符和条件表达式	70
5.4 switch 语句	71
5.5 程序设计举例	73
5.6 小结	75
习题	75
<b>第 6 章 循环结构程序设计</b>	<b>78</b>
6.1 while 语句	78
6.2 do-while 语句	79
6.3 for 语句	80

6.4 循环的嵌套	82
6.5 break 语句和 continue 语句	85
6.5.1 break 语句	85
6.5.2 continue 语句	86
6.6 程序举例	87
6.7 小结	91
习题	91
<b>第 7 章 数组</b>	<b>97</b>
7.1 一维数组的定义和引用	97
7.1.1 一维数组的定义	97
7.1.2 一维数组的引用	98
7.1.3 一维数组的初始化	98
7.1.4 一维数组程序举例	99
7.2 二维数组的定义和使用	100
7.2.1 二维数组的定义	100
7.2.2 二维数组的引用	101
7.2.3 二维数组的初始化	101
7.3 字符数组与字符串	104
7.3.1 字符数组的定义和引用	104
7.3.2 字符数组的初始化	104
7.3.3 字符数组的输入与输出	105
7.3.4 字符串处理函数	107
7.4 程序举例	110
7.5 小结	113
习题	114
<b>第 8 章 函数</b>	<b>119</b>
8.1 函数的定义和调用	120
8.1.1 函数定义的一般形式	120
8.1.2 函数的参数和函数的值	122
8.1.3 函数的调用	125
8.2 函数的嵌套调用	128
8.3 函数的递归调用	129
8.4 变量的作用域和存储类别	133
8.4.1 局部变量和全局变量	133
8.4.2 变量的存储类别	136
8.5 小结	139
习题	140

<b>第 9 章 预处理命令</b> .....	144
9.1 宏定义 .....	144
9.1.1 不带参数的宏定义 .....	144
9.1.2 带参数的宏定义 .....	146
9.2 文件包含处理 .....	148
9.3 条件编译 .....	150
9.4 小结 .....	152
习题 .....	152
<b>第 10 章 指针</b> .....	155
10.1 指针的概念 .....	155
10.1.1 变量的地址与变量的内容 .....	155
10.1.2 直接访问与间接访问 .....	156
10.1.3 指针与指针变量 .....	156
10.2 指针变量的定义与引用 .....	156
10.2.1 指针变量的定义 .....	156
10.2.2 指针变量的引用 .....	157
10.3 指针运算 .....	160
10.3.1 指针的算术运算 .....	160
10.3.2 指针的关系运算 .....	160
10.4 指针和数组 .....	161
10.4.1 指针与一维数组 .....	161
10.4.2 指针与二维数组 .....	163
10.5 指针与字符串 .....	166
10.6 指针数组和指向指针的指针 .....	169
10.6.1 指针数组 .....	169
10.6.2 指向指针的指针 .....	171
10.7 指针与内存的动态分配 .....	173
10.8 指针与数组作为函数的参数 .....	175
10.8.1 指针变量作为函数的参数 .....	175
10.8.2 数组名作为函数的参数 .....	177
10.9 带参数的 main 函数 .....	180
10.10 返回指针值的函数 .....	182
10.11 函数指针的定义与引用 .....	184
10.11.1 函数指针的定义 .....	184
10.11.2 函数指针变量的赋值 .....	185
10.11.3 函数指针变量的引用 .....	185
10.11.4 函数指针变量作为函数的参数 .....	186
10.12 小结 .....	188



习题	190
<b>第 11 章 结构体、共用体和枚举型</b>	<b>193</b>
11.1 概述	193
11.2 结构体	193
11.2.1 结构体类型的定义	193
11.2.2 结构体变量的定义和引用	194
11.2.3 简化结构体类型名	197
11.2.4 结构体数组	198
11.3 指向结构体类型数据的指针	201
11.3.1 指向结构体变量的指针	201
11.3.2 指向结构体类型数据的指针	202
11.4 线性链表	204
11.4.1 链表概述	204
11.4.2 建立单链表	205
11.4.3 在单链表中插入节点	206
11.5 共用体	207
11.5.1 共用体类型的定义	207
11.5.2 共用体变量的定义和引用	208
11.6 枚举类型	210
11.7 程序举例	212
11.8 小结	216
习题	216
<b>第 12 章 文件</b>	<b>219</b>
12.1 文件概述	219
12.2 文件类型及文件变量的定义	219
12.3 文件的打开与关闭	221
12.3.1 文件的打开	221
12.3.2 文件的关闭	222
12.4 文件的读写	223
12.4.1 fprintf 函数和 fscanf 函数	223
12.4.2 fputc 函数和 fgetc 函数	224
12.4.3 fputs 函数和 fgets 函数	225
12.4.4 fread 函数和 fwrite 函数	227
12.5 文件的定位	228
12.5.1 rewind 函数	228
12.5.2 fseek 函数	229
12.5.3 ftell 函数	230
12.6 文件应用举例	230

---

---

12.7 小结.....	232
习题.....	232
<b>附录</b> .....	<b>235</b>
附录 I 常见字符与 ASCII 对照表.....	235
附录 II 由 ANSI 标准定义的 32 个关键字.....	236
附录 III 运算符的优先级和结合性.....	237
附录 IV Turbo C 2.0 常用库函数.....	237
<b>参考文献</b> .....	<b>242</b>

# 第 1 章

## C 语言概述

C 语言是近年来在国内外得到迅速推广应用的一种计算机语言，本章学习的目的是让初学者在深入学习 C 语言之前，首先对 C 语言有个初步的了解，使他们知道 C 语言的基本特性及学习方法。

### 1.1 C 语言的发展历史及其基本特性

#### 1. C 语言的发展历史

C 语言作为计算机的一种高级语言，既有一般高级语言特性，又有低级语言的特性，所以它既适合编写系统程序又适合编写应用程序，已在国际上广泛流行。

C 语言是 1972 年由美国的 Dennis Ritchie 设计发明的，并首次在 UNIX 操作系统的 DECPDP-11 计算机上使用。C 语言的前身是 ALGOL（算法语言），1963 年英国剑桥大学在 ALGOL 基础上增添了处理硬件的能力，并命名为“复合程序设计语言（Comblined Programming Language, CPL）”。CPL 由于规模大，学习和掌握困难，所以没有流行开来。1967 年剑桥大学的马丁·理查德对 CPL 进行简化，推出“基本复合程序设计语言（Basic Comblined Programming Language, BCPL）”。1970 年美国贝尔实验室的肯·汤普逊对 BCPL 进行了进一步的简化，突出了硬件处理功能能力，并取了“BCPL”的第一个字母“B”作为新语言的名称。1972 年贝尔实验室的布朗.W.肯尼汉和丹尼斯.M.利奇对 B 语言进行了完善和扩充，并取了“BCPL”的第二个字母“C”作为新语言的名称，此时 C 语言就问世了。

随着微型计算机的日益普及，出现了许多 C 语言版本，为了统一标准，美国国家标准化协会（ANSI）于 1987 年制定了 C 语言的标准，称为“ANSI C”，成为现行的 C 语言标准。

#### 2. C 语言的基本特性

C 语言发展如此迅速，而且成为最受欢迎的语言之一，主要是由于它具有强大的功能。现在它已经成为在微、小、大、巨型计算机上，从系统程序设计到工程应用程序都能使用的一种高级程序设计语言。归纳起来，C 语言具有以下特点。

(1) C 语言是高、低级兼容语言。它比其他高级语言更接近硬件，比低级语言更接近算法，程序易编、易读、易查错、易修改。

(2) C 语言是一种结构化程序设计语言。它提供了完整的程序控制语句，可使程序层次清晰，便于使用、维护及调试。C 语言是以函数形式提供给用户的，这些函数可方便地调用，并具有多种循环、选择语句来控制程序流向，从而使程序完全结构化。

(3) C 语言的数据类型和运算符十分丰富。程序设计和算法描述更为简单和方便。

(4) C 语言功能齐全。因为引入了指针，所以使得程序运行效率更高。另外，C 语言也具有强大的图形功能，支持多种显示器和驱动器，而且计算机的逻辑判断功能也比较强大。

(5) C 语言是一种模块化程序设计语言。适合大型软件的研制和调试。

(6) C 语言程序的可移植性好。C 语言本身不依赖于机器硬件系统，从而便于在硬件结构不同的机种间和各种操作系统中实现程序的移植。

C 语言的优点很多，但也有不足之处应引起注意。C 语言语法限制不太严格，程序设计时自由度大。C 语言对变量的类型使用比较灵活，例如，整型和字符型数据可以通用。C 语言允许程序设计者有较大的自由度，放宽了对语法的检查。因此，程序员应认真仔细地检查程序，确保其正确性，不要过分依赖 C 语言编译程序去查错。不过，程序员使用 C 语言编写程序时会感到限制少，灵活性大，功能强，可以编写出任何类型的程序。

关于以上特点，初学者可以先了解一下，等到本书内容全部学完后，就能深刻体会到以上所提到的各点。

## 1.2 简单的 C 语言程序介绍

C 语言程序是由 C 语言组成的语句序列，通常一个 C 语言程序包括一个或多个函数，其中必定有一个函数称为主函数，其函数名为 `main`。下面来看几个简单的 C 语言程序例子，以对 C 语言有一个初步的认识，进而了解编写 C 语言程序的一些基本问题。

C 语言像其他语言一样，按其规定的格式和提供的语句，由用户编写程序。请看下面几段 C 语言源程序。

**【例 1.1】** 输出一行信息的 C 程序。

```
main()          /*主函数*/
{
    printf("How are you ?\n");
}
```

该程序只由一个主函数构成，程序的第一行 `main` 为主函数名，主函数名后的一对圆括弧 `()` 是函数的组成部分。C 程序中的所有函数的函数名后都必须带一对圆括弧。程序的第二至第四行为函数体部分，函数体要用一对大括号 `{ }` 括住。程序的第三行为一个函数调用语句，用 `printf` 函数来输出数据，该函数的使用将在第 4 章中详细介绍。程序中的分号 `;` 是 C 语句的结束标志，规定 C 语句都必须以分号结尾。

程序运行结果为

```
How are you ?
```

**【例 1.2】** 求两个整数之和，两个整数由键盘输入。

```
main ()
{
    int x,y,z;
    scanf("%d,%d",&x,&y); /*输入两值分别放入变量 x、y 中*/
```

```

z=x+y;
printf("sum=%d\n",z);
}

```

该C语言程序只由main一个函数组成。程序的第三行为定义变量部分，在这里定义了3个整型变量x，y和z，其中的“int”为数据类型符，表示整型；“x,y,z”为3个变量的名称，定义变量时变量名之间用逗号（,）隔开。程序的第四行为函数调用语句，通过scanf函数给变量x和y输入数据，具体使用将在第4章中详细介绍。程序的第五行为赋值语句，将变量x与y两数之和赋予变量z，即让变量z存放两变量x、y的和。程序的第六行用于输出变量z的值。

程序运行结果为

78, -30✓ （注：本书中所有的“✓”均表示回车符号，不再另行说明）

sum=48

**【例 1.3】** 求3个整数中的最小值，3个整数由键盘输入。

```

/*求 x1,x2,x3 三个数中的最小值的函数*/
int min(int x1,int x2,int x3) /*x1,x2、x3为3个int型的形参*/
{ int x; /*定义int型变量x*/
  x=x1; /*先把x1作为最小值并存放于x中*/
  if(x>x2) x=x2; /*如果x2小于x，则把x2放于x中*/
  if(x>x3) x=x3; /*如果x3小于x，则把x3放于x中*/
  return(x); /*将所求的最小值x返回到主调函数中*/
}
/*完成3个整数的输入，并输出这3个数中最小值的函数*/
main()
{ int x,y,z,k;
  scanf("%d,%d,%d",&x,&y,&z);
  k=min(x,y,z); /*调用min函数求x,y,z三个数中最小值*/
  printf("min=%d\n",k);
}

```

该C语言程序由min、main这两个函数构成，它们各有一定的功能。

程序运行结果为

15, 53, -57✓

min=-57

从上述3个C语言程序例子，可以看出以下几个C语言程序的特点。

(1) C语言程序是由函数构成的。

每个C语言源程序必须有一个而且只能有一个称做主函数的main函数，除主函数外，可以没有其他函数（如例1.1，例1.2），也可以有一个或多个其他函数（如例1.3）。因此函数是构成C语言程序的基本单位。

(2) 每个函数（包括主函数）的定义分为两部分：函数说明部分和函数体。

函数说明部分的格式如下：

返回值的类型 函数名 (数据类型符 形式参数 1, ..., 数据类型符 形式参数  $n$ )

函数体的格式如下:

```
{ 变量定义部分  
 实现函数功能的语句串  
}
```

当然, 函数体中也可以没有任何内容, 只有一对 {}, 这在一些特殊场合可能会遇到。

(3) C 语言源程序中的所有语句都必须用一个分号 (;) 作为终止符, 否则 C 语言认为该语句没有结束。但在预处理命令、函数头中右大括号 (}) 之后不能加分号。

(4) C 语言中的字符有大、小的区别, 习惯上用小写字母。只有常量、宏定义等用大写字母, 所有的关键字必须小写, 如 `int`、`if`、`return` 等。同样, 使用变量时也必须注意变量名的大、小写, 相同字母的大、小写代表不同的变量, 如 `a` 与 `A` 表示两个不同的变量。

(5) 可以用 `/*.....*/` 对 C 语言程序中的任何部分做注解, 一个好的有使用价值的源程序都应当加上必要的注解, 以增强程序的可读性。通常放在一段程序的开始, 用以说明该段程序的功能; 或者放在某个语句的后面, 对该语句进行说明。源程序编译时, 注解部分将不参加编译, 也不会出现在目标程序中。

(6) C 语言程序的书写格式很灵活, 一个语句可以写在多行上, 在一行上也可以书写多个语句。在程序清单中的任何一处都可以插入空格符号或回车符号。通常书写程序时采用缩格并对齐的书写方式。

(7) C 语言程序的执行总是从主函数开始, 并在主函数中结束。主函数的位置是任意的, 可以在程序的开头、两个函数间或程序的结尾。

(8) 主函数可以调用任何非主函数, 任何非主函数可以相互调用, 但不能调用主函数。

(9) C 语言本身没有输入、输出语句, 输入、输出的操作是由库函数 `scanf` 和 `printf` 来完成的。

## 1.3 C 语言程序的上机步骤

本书以 Turbo C 2.0 为例, 介绍 Turbo C 2.0 集成开发环境的使用, 它可运行于 PC 系列微机, 也可在 DOS 环境中使用。假设 Turbo C 2.0 已经安装在 C 盘的 `C:\tc` 目录中。如果在实际操作中, 安装的位置不是在 `C:\tc`, 操作时就应该按 Turbo C 实际安装的位置使用正确的路径。

### 1. 编辑源程序

在 DOS 环境下有以下两种进入编辑状态的方法。

(1) 设编辑的源文件名为 `file.c`, 可输入命令:

```
TC file ✓
```

如果它是一个新文件, 则 Turbo C 屏幕窗口为空白, 等待用户输入源文件; 如果它是一个已存在的文件, 则系统将其调入内存并在屏幕窗口显示其内容, 用户可以根据需要进行修改。

(2) 输入命令:

```
TC ✓
```

在 Windows 环境下可以直接双击 TC 文件。  
这时，屏幕出现一个窗口，窗口示意图如图 1.1 所示。

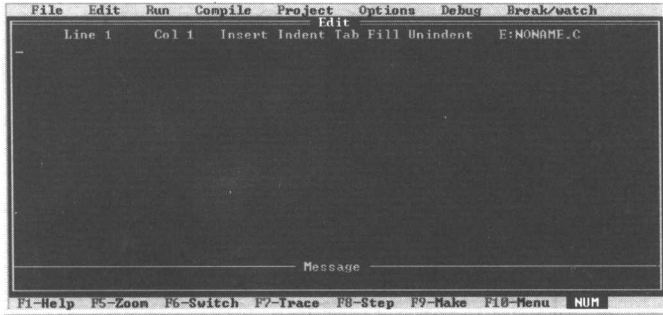


图 1.1 Turbo C 2.0 主屏幕

图 1.1 中，最顶上一行是 Turbo C 2.0 的主菜单，中间窗口为编辑窗口，接下来是信息窗口，最底下一行为参考行。以后的编辑、编译、调试以及运行都将在这个主屏幕中进行。这里简单介绍主菜单的内容及相应的操作方式。

在主菜单中，除 Edit 外，其他各项均有子菜单，只要用“Alt”键加上菜单上某项中的第一个字母（即大写字母），就可以进入该项的子菜单中。如按“Alt+F”组合键，就进入 File 菜单项。

对各主菜单及其子菜单的操作相似，下面仅以 File 菜单为例讲解操作方法。

按“Alt+F”组合键，或按“F10”键，激活主菜单，然后按“←”键或“→”键移动屏幕上的光标到 File 上后，再按回车键，就可以进入 File 菜单，其子菜单如图 1.2 所示。用“↑”键或“↓”键将屏幕上的光标移到某项后按回车键，即可执行该项命令。

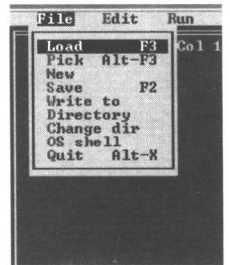


图 1.2 File 菜单

### 2. 保存源文件

方法一：按“F2”键，将以原文件名存盘。

方法二：选取菜单项“File/Write to”，将以新文件名存盘。

### 3. 编译、连接生成目标文件

方法一：分步执行。

按“Alt+F9”组合键，编译代码文件 (.obj)；再按“F9”键，连接生成可执行的目标文件 (.exe)。或通过“Compile/Compile to OBJ”、“Compile/LINK EXE file”菜单命令来完成。

方法二：连续执行。

按“F9”键，首先编译并生成中间代码文件 (.obj)；接着自动连接生成可执行的目标文件 (.exe)。或通过“Compile/Make EXE file”菜单命令来完成。

### 4. 运行目标文件

按“Ctrl+F9”组合键，将运行刚编译连接的目标文件。或通过“Run/Run”菜单命令来完成。

### 5. 显示运行结果

按“Alt+F5”组合键，可以暂时退出 Turbo C，进入 DOS 环境，用户在屏幕上可以看到程序运行的结果。按任意键返回 Turbo C 状态。或通过“Run/User screen”菜单命令来完成。

注意：如果程序运行结果有错，可以转入前面第三步，重新修改源文件、编译、连接、运行。C 语言程序的开发过程如图 1.3 所示。

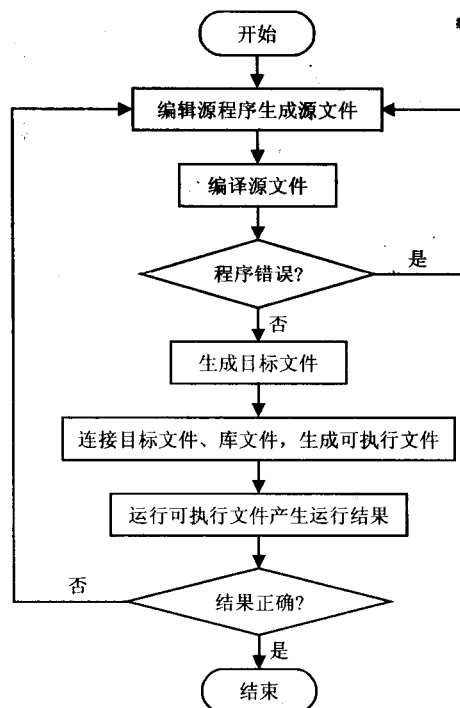


图 1.3 C 语言程序的开发过程

## 1.4 小 结

本章主要介绍了 C 语言的来源及特性, C 语言程序的基本结构, 并且还详细介绍了 Turbo C 集成环境的应用, 新建源程序的方法, 输入、编译、调试和运行 C 语言程序的基本方法和步骤。

在 C 语言编写和开发过程中要注意以下几个问题。

- (1) 编写程序要规范, 培养良好的程序设计风格, 最好采用缩格并对齐的书写方式。
- (2) C 语言程序由一个或多个函数构成, 有且只有一个主函数 (main 函数)。
- (3) C 语言程序中的所有语句都必须以分号 (;) 作为结束符, 通常每一行为一条基本语句, 复合语句必须用 “{}” 括起来。
- (4) C 语言中的所有关键字 (将在第 3 章中介绍) 都必须小写, 不能另作他用, 例如想把 “int” 标识符作为一变量名使用是不允许的。
- (5) 要熟练掌握源程序的编辑、保存、编译、连接及运行的操作方法, 了解程序调试的基本概念。
- (6) C 语言程序的执行总是从主函数 (main 函数) 开始, 并在主函数中结束。



## 习 题

### 一、简答题

1. 简述 C 语言的发展过程。
2. C 语言有哪些主要特点？

### 二、程序分析题

写出下列 C 语言程序的输出结果。

```
main()  
{ printf("\n");  
  printf("Welcome to Beijing \n 2008!\n");  
}
```

### 三、填空题

1. C 语言程序是由\_\_\_\_\_构成的，其中有且只能有一个\_\_\_\_\_函数，该函数名为\_\_\_\_\_。
2. 用高级语言编写的程序称为\_\_\_\_\_程序，它要经过\_\_\_\_\_程序一次编译产生\_\_\_\_\_程序然后执行，或经过\_\_\_\_\_程序翻译一句执行一句的方式执行。
3. 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\_\_\_\_\_。

### 四、编程题

编写一个 C 语言程序，输入 a、b、c 三个实数值，计算并输出其平均值  $ave=(a+b+c)/3$ 。