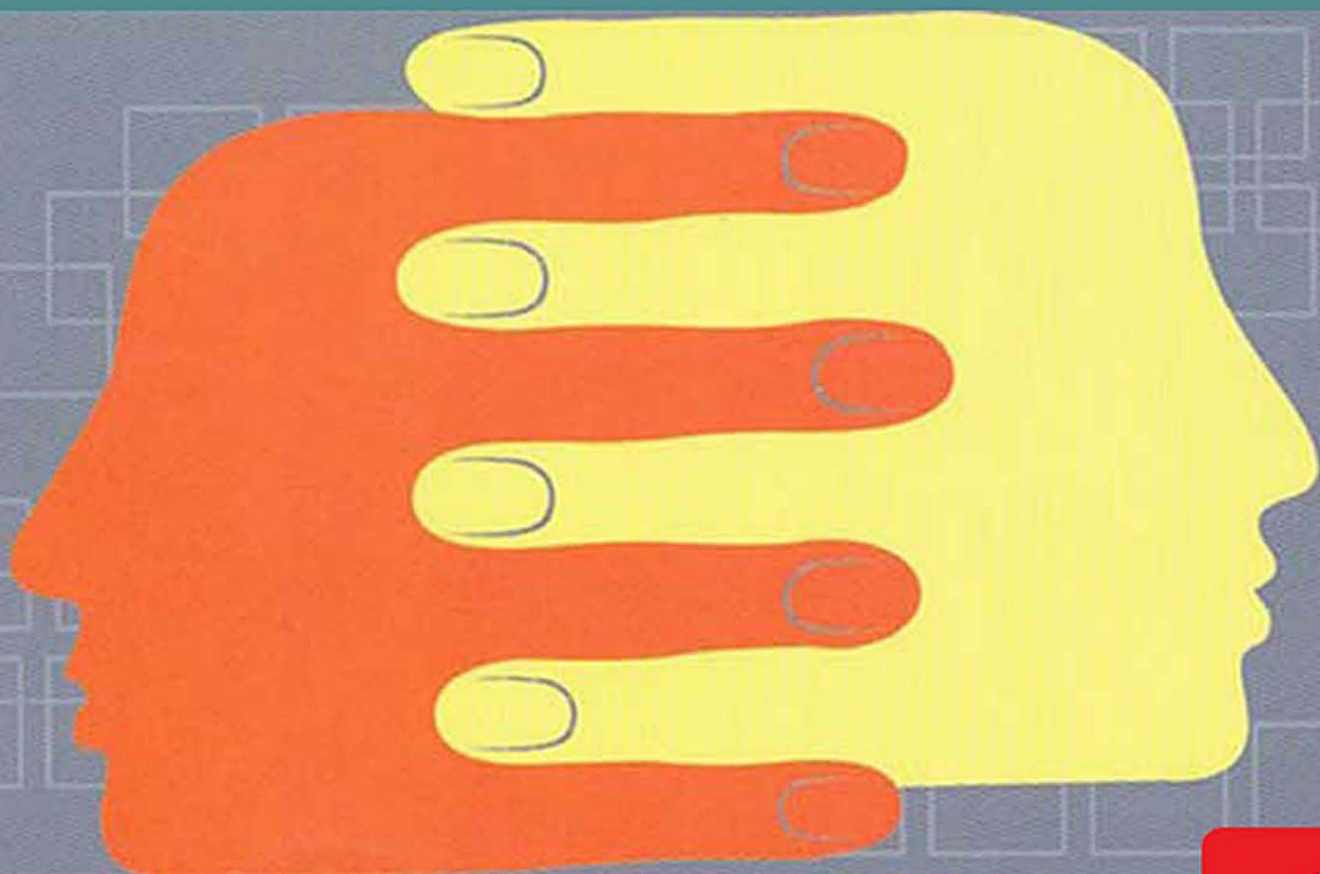


C 语言程序设计与开发

董艳雪 著

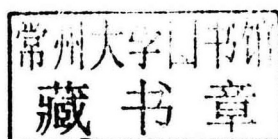


电子科技大学出版社



C 语言程序设计与开发

董艳雪 著



电子科技大学出版社

图书在版编目 (CIP) 数据

C 语言程序设计与开发 / 董艳雪著. —成都: 电子科技大学出版社, 2014.6

ISBN 978-7-5647-2370-5

I. ①C… II. ①董… III. ①C 语言—程序设计 IV. ①TP312

中国版本图书馆 CIP 数据核字 (2014) 第 092535 号

C 语言程序设计与开发

董艳雪 著

出 版: 电子科技大学出版社 (成都市一环路东一段 159 号电子信息产业大厦 邮编: 610051)

策划编辑: 谭炜麟

责任编辑: 谭炜麟

主 页: www.uestcp.com.cn

电子邮箱: uestcp@uestcp.com.cn

发 行: 新华书店经销

印 刷: 北京九州迅驰传媒文化有限公司

成品尺寸: 185mm×260mm 印张 12.5 字数 295.5 千字

版 次: 2014 年 6 月第一版

印 次: 2014 年 6 月第一次印刷

书 号: ISBN 978-7-5647-2370-5

定 价: 68.00 元

■ 版权所有 侵权必究 ■

◆ 本社发行部电话: 028-83202463; 本社邮购电话: 028-83208003。

◆ 本书如有缺页、破损、装订错误, 请寄回印刷厂调换。

前 言

C 语言是一种计算机程序设计语言。它既具有高级语言的特点，又具有汇编语言的特点。它可以作为工作系统设计语言，编写系统应用程序，也可以作为应用程序设计语言。编写不依赖计算机硬件的应用程序。C 语言的应用范围广泛，具备很强的数据处理能力，适于编写系统软件及三维、二维图形和动画程序。具体应用如单片机以及嵌入式系统开发。

本书内容编排以程序设计为主线，注重培养读者程序设计的思维方式和技巧。各章的具体内容分别为：第 1 章对 C 语言和 C 程序设计的基本知识做了简单的介绍，使读者对 C 程序有个初步的了解。第 2 章介绍了 C 语言提供的基本数据类型及数据运算规则。第 3 章介绍了结构化程序设计的三种基本程序结构：顺序结构、选择结构和循环结构。第 4 章介绍了一维数组、二维数组和字符型数组的定义、初始化及基本操作，包括数组元素的引用、赋值、输入和输出。第 5 章介绍了 C 语言函数的定义和调用方法、函数间的数据传递方法、变量和函数的存储类型对函数调用的影响以及递归函数的设计等。第 6 章介绍了指针的概念和运算规则，指针访问变量、一维数组和二维数组，以及用指针处理字符串的方法。

本书可作为高等院校开设的 C 语言程序设计课程的参考书，也可作为成人教育以及在职人员的培训教材，同时也可供参加全国计算机等级考试的读者参考。

编 者
2014 年 1 月

目 录

第1章 C语言概述	1
1.1 C语言的发展及主要特点.....	1
1.2 C程序的基本结构.....	2
1.3 C程序的开发过程.....	6
1.4 C语言集成开发环境的使用.....	7
本章习题	13
第2章 基本数据类型和运算.....	15
2.1 C语言的数据类型.....	15
2.2 常量及其类型	17
2.3 变量及其类型	22
2.4 运算符与表达式	26
2.5 不同数据类型的输出	41
2.6 不同数据类型的输入	46
2.7 常见错误分析（选学）	51
本章习题	53
第3章 结构化程序设计	59
3.1 结构化程序设计的思想及流程图	59
3.2 结构化程序设计的三种基本结构	60
3.3 几种转移控制语句	77
3.4 C程序语句.....	80
3.5 C程序举例.....	81
3.6 易错知识点总结（选学）	84
本章习题	86
第4章 数组及字符串	91
4.1 数组的概念	91
4.2 一维数组	91
4.3 二维数组	96
4.4 字符串与字符函数	100

4.5 数组与字符串的应用举例（选学）	113
本章习题	116
第5章 函数	122
5.1 C程序的模块结构.....	122
5.2 函数的分类	123
5.3 函数的定义和调用	124
5.4 函数间的参数传递	130
5.5 函数的嵌套与递归	136
5.6 系统函数	140
5.7 变量的存储类别、作用域、生存期	146
5.8 程序举例（选学）	152
本章习题	154
第6章 指针	160
6.1 指针的基本概念	160
6.2 指针变量的运算	162
6.3 指向数组元素的指针	167
6.4 指向二维数组的指针变量	171
6.5 指向字符串的指针变量	173
6.6 指针型函数	176
6.7 指针数组	178
6.8 二级指针	179
6.9 程序举例	180
6.10 用指针进行内存动态分配	184
6.11 指针数据类型的总结（选学）	186
本章习题	188

第1章 C语言概述



教学提示

本章对C语言和C程序设计的基本知识做了简单的介绍，后续章节还将对这些内容进行详细的讲解。通过本章的学习，学生将对C程序有个初步的了解，以利于后续章节的学习。



教学目标

熟悉C程序的基本结构和书写风格；掌握C语言关键字和标识符的命名方法，了解C编译系统提供的标题文件的功能，学会用输入/输出函数编制简单的C程序，掌握Turbo C集成开发环境的使用。

1.1 C语言的发展及主要特点

1.1.1 C语言的发展史

C语言是一种面向过程的程序设计语言。其前身是ALGOL60。1963年，英国的剑桥大学和伦敦大学首先将ALGOL60发展成CPL，1967年英国剑桥大学的Martin Richards 将CPL改写成BCPL；1970年美国贝尔实验室的Ken Thompson将BCPL修改成B语言，并用B语言开发了第一个高级语言的UNIX操作系统。1972年Ken Thompson与在开发UNIX系统时的合作者Dennis Ritchie一起将B语言改成了C语言。1978年，Brain W·Kernighan和Dennis M·Ritchie合著了著名的《The C Programming Language》，该书介绍的C语言成为后来广泛使用的C语言版本的基础。

由于自身的优点，C语言在其后的十几年中得到了广泛的使用，适用于不同机种和不同操作系统的C编译系统相继问世。1983年美国国家标准局（ANSI）制定了C语言标准，这个标准不断完善，并从1987年开始实施ANSI的标准C。

C语言发展迅速，而且成为最受欢迎的语言之一。许多著名的系统软件，如UNIX操作系统等都是由C语言编写的。

由于C语言的强大功能和各方面的优点逐渐为人们认识，到了20世纪80年代，C语言开始进入其他操作系统，并很快在各类大、中、小型计算机上得到了广泛的使用，成为当代最优秀的程序设计语言之一。

目前，在计算机上广泛使用的C语言编译系统有Microsoft C，Turbo C等。本书选定的上机环境为Turbo C 2.0。同时，为了满足读者学习和参加计算机等级考试等多方面的需求，附录七对Visual C++ 6.0集成开发环境进行了介绍，读者可以参考学习。

1.1.2 C语言的主要特点

C语言的特点归纳起来主要有以下几个：

（1）C语言简洁、紧凑，使用方便、灵活。C语言一共只有32个关键字，9种控制语

句，而且程序书写形式自由，主要用小写字母表示，压缩了一些不必要的成分。

(2) C语言同时具备了高级语言和低级语言的特征。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。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。换句话说，C语言既具有汇编语言的强大功能，又没有汇编语言的难度，特别适合做底层开发。C语言既可以用来设计芯片，也可以用来编写操作系统。

(3) C语言是结构化语言，具有结构化的控制语句。结构化语言的显著特点是代码及数据的分隔化，即程序的各个部分除了必要的信息交流外，彼此独立。这种结构化方式可使程序层次清晰，便于使用、维护以及调试。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、条件语句，控制程序流向，从而使程序完全结构化。

(4) C语言具有各种各样的数据类型。C语言支持各种高级语言普遍使用的基本数据类型，并允许用基本数据类型构造复杂的数据类型。同时，引入了指针概念，可使程序效率更高。

(5) C语言具有强大的图形功能，支持多种显示器和驱动器；而且计算功能、逻辑判断功能也比较强大，可以实现决策目的。

(6) C语言适用范围大。C语言的一个突出的优点就是适合于多种操作系统，如DOS，UNIX，也适用于多种机型。

(7) C语言生成目标代码质量高，程序执行效率高。

上面只介绍了C语言最容易理解的一般特点，至于C语言内部的其他特点将结合以后各章的内容作介绍。

1.2 C程序的基本结构

C程序的基本结构是指一个C程序的基本组成部分。本节将通过程序实例来说明C程序的基本结构。

1.2.1 C程序的实例

首先通过一个简单C程序的实例来说明C语言源程序结构的特点和书写格式。

【例1.1】一个简单的C程序。

```
void main()
{
    printf("Hello,Human!\n");    /*输出Hello,Human!*/
}
```

这个程序的功能是输出下面一行信息：

 Hello,Human!

上述程序中：

(1) **main**是主函数的函数名，表示这是一个主函数。每一个C语言源程序都必须有且只有一个主函数**main**。

(2) 函数体由花括号{}括起来。上面例子中的函数体只有一个**printf**输出语句。**printf**是C语言中的输出函数，其功能是把要输出的内容送到显示器上去显示。语言中的双引

号用来显示一个字符串，双引号内的字符串将按原样输出：“\n”是换行符，即在输出“Hello,Human!”后回车换行。

(3) /*.....*/表示注释部分，为便于理解，通常用汉字表示注释，当然也可以用英语或汉语拼音作注释。注释只是给人看的，对编译和运行不起作用。注释可以加在程序中的任何位置。

下面看一个相对复杂的C程序。

【例1.2】一个复杂的C程序。

```
#include<math.h>                /*include为文件包含命令*/
#include<stdio.h>
main()                          /*主函数*/
{
    double x,y;                 /*定义变量*/
    printf("input number:");    /*输出字符串“input number:”*/
    scanf("%lf",&x);           /*输入变量x的值*/
    y=cos(x);                   /*求x的余弦值，并把它赋给变量y*/
    printf("cosine of %lf is %lf\n",x,y); /*显示程序的运算结果，即显示x和y的值*/
}
```

这个程序的功能是求一个从键盘上输入的数x的余弦值，然后输出结果。

程序的运行情况为：

```
input number:1.2✓
cosine of 1.200000 is 0.362358
```



说明 本书用✓表示回车。

上述程序中：

(1) 在main()之前的两行语句称为预处理命令，关于预处理在后面的章节中会接触到。预处理命令还有其他几种，这里的include称为文件包含命令，其意义是把尖括号<>或引号""内指定的文件包含到本程序来，成为本程序的一部分。被包含的文件通常是由系统提供的，其扩展名为.h，因此也称为头文件。

(2) 在本例中，使用了三个库函数：输入函数scanf、余弦函数cos和输出函数printf。cos函数是数学函数，其头文件为math.h，因此在程序的主函数前用include命令包含了math.h。scanf和printf函数是标准输入/输出函数，其头文件为stdio.h，在主函数前也用include命令包含了stdio.h。C语言的头文件中包含了各个标准库函数的函数原型。因此，凡是在程序中调用一个库函数时，都必须包含该函数原型所在的头文件。



说明 C语言规定对scanf和printf这两个函数可以省去对其头文件的包含命令。所以在本例中也可以删去第二行的包含命令“#include<stdio.h>”。在例1.1中使用的printf函数就省略了“#include<stdio.h>”。

(3) 在例题中的主函数体又分为两部分：说明部分和执行部分。

① 说明部分完成变量的类型说明。C语言规定，源程序中所有用到的变量都必须先说明后使用，否则将会出错。这一点是编译型高级程序设计语言的一个特点。例1.1中未使用

任何变量，因此无说明部分。说明部分是C语言源程序结构中很重要的组成部分。本例中使用了两个变量x, y, 用来表示输入的自变量和cos函数值。由于cos函数要求这两个量必须是双精度浮点型，故用类型说明符double来说明这两个变量。说明部分后的四行为执行部分或称为执行语句部分，用以完成程序的功能。

② 执行部分的第一行是输出语句，调用printf函数在显示器上输出提示字符串，提示用户输入自变量x的值。第二行为输入语句，调用scanf函数，接受键盘上输入的数并存入变量x中。第三行是调用cos函数并把函数值送到变量y中。第四行是用printf函数输出变量y的值，即x的余弦值。到此程序全部结束。

1.2.2 输入/输出函数的简单介绍

输入是将原始数据通过输入设备送入计算机，输出是将保存在内存中的计算结果送到输出设备上。为完成此操作，C语言编译系统提供了输入/输出函数。其中常用的是格式输出函数printf和格式输入函数scanf。为了以后学习的方便，本节对这两个函数作一简单的介绍，详细的介绍见后面的章节。

1. 格式输出函数printf

格式输出函数printf的功能是按指定的格式输出数据，其一般的调用格式为：

```
printf("格式控制字符串", 参数表);
```

其中，printf是函数名，其后面括号中的内容为该函数的参数：格式控制字符串用双引号括起来，用来规定输出格式，如%f用来输出实数，%c用来输出字符；参数表中包含零个或多个输出项，这些输出项可以是实数、变量或表达式，多个输出项之间用逗号隔开。

例如：语句printf("%d,%d",a,b);用来按十进制整数形式输出变量a和b。

2. 格式输入函数scanf

格式输入函数scanf的功能是按指定的格式输入数据，其一般的调用格式为：

```
scanf("格式控制字符串", 参数表);
```

其中，scanf是函数名，其后面括号中的内容为该函数的参数：格式控制字符串用双引号括起来，用来规定输入格式，其用法和printf函数中规定的相同；参数表中至少包含一个输入项，且必须是变量的地址，多个输入项之间用逗号隔开。

例如：语句scanf("%d%d",&a,&b);用来接受从键盘输入的两个十进制整数，并分别存放在变量a和b中。



说明 变量地址的表示形式是在变量名前加上一个“&”。

1.2.3 C程序的基本结构

函数是C程序的基本结构，一个C程序由一个或多个函数组成，一个C函数由若干条C语句构成，一条C语句由若干基本单词组成。

C函数是完成某个整体功能的最小单位，是相对独立的模块。简单的C程序可能只有一个主函数，而复杂的C程序则可能包含一个主函数和任意多个其他函数。所有C函数的结构都包括三部分：函数名、形式参数和函数体。

图1-1所示为C程序的一般格式。其中的main为主函数，sub1到subn为子函数。在C程序中，主函数名字是固定的，其他的函数名则可以根据标识符的命名方法任意取名。形式参

数是函数调用时进行数据传递的主要途径，当形式参数表中有多个参数时，相互之间用逗号隔开。有的函数可能没有形式参数。花括号{}括起来的部分为函数体，用来描述函数的功能，一般函数体由变量定义和完成本函数功能的语句序列组成。程序在执行时，无论各个函数的书写位置如何，总是先执行main函数，再由main函数调用其他函数，最终终止于main函数。

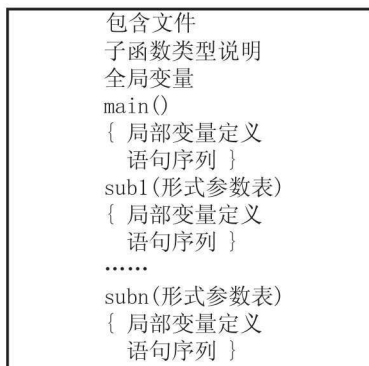


图1-1 C程序的一般格式

1.2.4 C语言的词汇

在C语言中使用的词汇分为六类：标识符、关键字、运算符、分隔符、常量和注释符等。

1. 标识符

标识符用来表示程序中使用的变量名、函数名、标号、数组名、指针名、结构体名、共用体名、枚举常量名以及用户定义的数据类型名等，除库函数的函数名由系统定义外，其余都由用户自定义。C语言规定，标识符是由字母（A~Z，a~z）、数字（0~9）、下画线（_）组成的字符串，并且第一个字符必须是字母或下画线。例如：标识符a，BOOK_1，sum5是合法的。以下标识符是非法的：

- (1) 3s，以数字开头。
- (2) s*T，出现非法字符*。
- (3) !3x，以非法字符!开头。
- (4) bowy-1，出现非法字符-（减号）。



说明 在使用标识符时还必须注意以下几点：

- ① 标准C不限制标识符的长度，但它受各种版本的C语言编译系统限制，同时也受到具体机器的限制。例如在某版本C语言中规定标识符前8位有效，当两个标识符前8位相同时，则被认为是同一个标识符。
- ② 在标识符中，大小写是有区别的。例如：BOOK和book是两个不同的标识符。
- ③ 标识符虽然可由程序员随意定义，但标识符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“顾名思义”。

2. 关键字

关键字是由C语言规定的具有特定意义的字符串，通常也称为保留字。用户定义的标识符不应与关键字相同。C语言的关键字分为以下几类：

- (1) 类型说明符。用于定义和说明变量、函数或其他数据结构的类型。

- (2) 语句定义符。用于表示一个语句的功能。
 - (3) 预处理命令字。用于表示一个预处理命令。
- ANSI C标准规定的关键字有32个，如表1-1所示。

 说明 所有的C语言关键字都必须小写。

表1-1 ANSI C标准规定的关键字

auto	volatile	switch	signed	short	sizeof
default	break	while	typedef	union	unsigned
float	do	case	char	const	continue
register	for	double	else	enum	extern
struct	return	goto	if	int	long
static	void				

3. 运算符

C语言中的运算符由一个或多个字符组成。运算符与变量、函数一起组成表达式，表示各种运算功能。

4. 分隔符

分隔符用来分隔相邻的标识符、关键字和变量等。C语言中常用的分隔符有逗号、空格和制表符等。

5. 常量

C语言中使用的常量可分为数值常量、字符常量、字符串常量、符号常量和转义常量等多种。

6. 注释符

以“/*”开头并以“*/”结尾的字符串为C语言的注释符。注释可出现在程序中的任何位置，用来提示用户或解释程序。程序编译时，不对注释作任何处理。

1.3 C程序的开发过程

C语言是一种编译型的程序设计语言，开发一个C程序要经过编辑、编译、链接和运行四个步骤。

1. 编辑

一般来说，编辑是指C语言源程序的输入和修改。使用文本编辑器来创建源代码的文件，最后以文本文件的形式存放在磁盘上，文件名由用户自行定义，扩展名一般为.c，例如hello.c，b.c等。许多文本编辑器都可以用来编辑源程序，例如Windows记事本、DOS的EDIT等。

2. 编译

编译是编译器把C语言源程序翻译成二进制目标程序。目标程序文件的主文件名与源程序的主文件名相同，扩展名为.obj。如果在编译的过程中出现错误，系统会给出“出错信

息”，此时用户需要回到编辑阶段进行修改，直到编译通过为止。

3. 链接

编译成功后的目标程序仍然不能运行，需要用链接程序将编译过的目标程序和程序中用到的库函数链接装配在一起，形成可执行的文件。可执行文件的主文件名与源程序的主文件名相同，其扩展名为.exe。

4. 运行

链接后生成的可执行文件装入内存后即可运行，并输出运行结果。在很多公用环境，如MS-DOS，UNIX和Linux操作平台中，只要键入相应的可执行文件名即可运行一个程序。在其他环境下，例如Windows环境下可以通过选择菜单中的选项或按下特殊键来编辑并执行C程序。还可以通过双击文件名或图标直接从操作系统运行。

如果在运行程序的过程中得不到预期的结果，就要重复进行编辑、编译、链接和运行四个步骤。C程序的开发过程如图1-2所示。

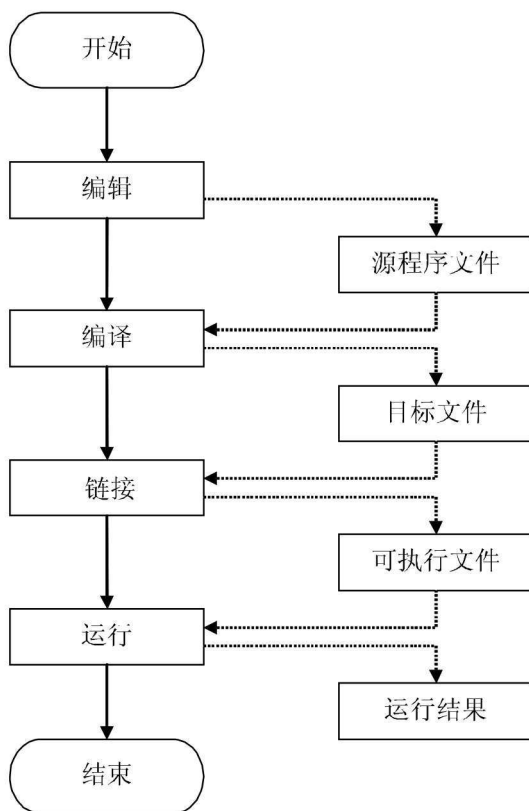


图1-2 C程序的开发过程

1.4 C语言集成开发环境的使用

Turbo C 是美国Borland 公司推出的C语言的集成开发环境，程序的编辑、编译、链接和运行都可以在该环境下完成。

1.4.1 Turbo C 2.0的启动

假定已经将Turbo C 2.0安装到了计算机中，位置在C盘的TC目录下；同时采用Windows XP操作系统（对Windows 98和Windows 2000的用户也同样适用）。启动Turbo C 2.0的步骤如下：

（1）在Windows XP窗口左下角选择【开始】|【所有程序】|【附件】|【命令提示符】命令，打开【命令提示符】窗口。

（2）在提示光标处输入：`cd \`，回车后将回到C盘的根目录，如图1-3所示。

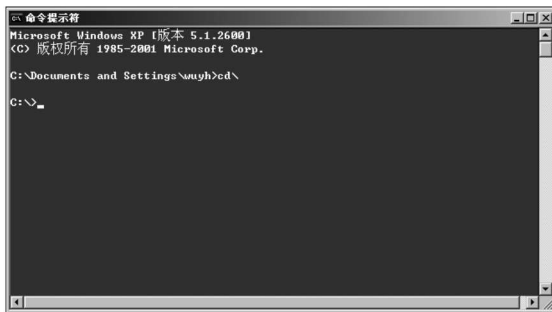


图1-3 【命令提示符】窗口

（3）在窗口中输入：

`C:\>cd tc` ✓

`C:\TC>tc` ✓

打开C程序的开发环境——Turbo C 2.0，如图1-4所示。



图1-4 Turbo C 2.0开始界面

1.4.2 Turbo C的集成开发环境

进入Turbo C 2.0集成开发环境后，屏幕中间显示了一些版本等方面的信息，按键盘上的Esc键可以关闭该窗口。

图1-4所示的集成环境分为4个部分：最顶行为“主菜单”，中间是编辑窗口和编译信息窗口，最底行为功能键提示行。这4个窗口构成了Turbo C 2.0的主屏幕，程序的编辑、编译、调试、链接以及运行都将在这个主屏幕中进行。

通过主菜单可以选择Turbo C集成环境所提供的各项功能。同时，使用键盘上的↑，↓，←，→（上、下、左、右）键可以选择菜单条中所需要的菜单项，被选中的项反向显示。

现在介绍经常使用的几个菜单。

1. 文件(File)菜单

文件菜单提供存取文件的功能，含有9个下拉子菜单，分别如下：

Load	装入文件
Pick	显示最近装入过的最后9个文件的列表
New	编辑新文件，文件名为NONAME.C
Save	将正在编辑的文件存盘
Write to	将正在编辑的文件改名存盘
Directory	显示当前工作目录的内容
Change dir	显示当前工作目录的路径名，允许把它改为另一个名字
OS shell	暂时退回到DOS环境，键入EXIT命令返回Turbo C
Quit	退出Turbo C，返回DOS系统

2. 编辑 (Edit) 菜单

编辑菜单提供输入和编辑程序的功能，不含下拉子菜单。

3. 运行 (Run) 菜单

运行菜单提供运行和调试程序的功能，含有6个下拉子菜单，常用的有：

Run	编译、链接和运行正在编辑的源程序
User screen	查看程序运行时所产生的屏幕输出

4. 编译 (Compile) 菜单

编译菜单提供编译和链接目标程序的功能，含有6个下拉子菜单，常用的有：

Make EXE file	直接将源程序编译和链接成可执行文件
Link EXE file	链接当前的OBJ文件和库文件
Primary C file	指定待编译的主文件，而不是当前编辑的文件
Get info	显示当前对话框的信息

5. 可选项 (Options) 菜单

可选项菜单提供设定集成环境操作方式的功能，包含7个下拉子菜单，常用的有：

Compiler	选择编译程序的存储模式
Environment	更改集成环境的存储模式
Directories	设定标题文件、库文件和可执行文件所在的目录

1.4.3 确立TC工作环境

一般在安装Turbo C时，如果用户不作另外的指定，系统会按照默认的方案建立一个目录C:\TC用来存放Turbo C的系统文件，同时在TC目录下建立一个INCLUDE子目录用来存放“包含文件”，另外建立一个LIB子目录用来存放库函数。

安装Turbo C时，如果用户不采用系统提示的子目录名，而想自己另外指定子目录名，则应当通知Turbo C系统，否则在调试程序时会提示找不到指定的文件。当这种情况发生时，可以采用以下的操作解决这个问题。

(1) 打开Turbo C的主界面后，按键盘上的快捷键Alt+O，在出现的【Options】下拉菜单中，用↑和↓键选择【Directories】项，如图1-5所示。



图1-5 设置Turbo C系统的工作环境（1）

(2) 按回车后弹出如图1-6所示的窗口。



图1-6 设置Turbo C系统的工作环境（2）

现在介绍图1-6所示窗口中的各项含义。

- | | |
|------------------------------------|-----------------------|
| ① Include directories:C\TC\INCLUDE | Install建立的“包含文件”默认目录。 |
| ② Library directories:C\TC\LIB | Install建立的库文件默认目录。 |
| ③ Output directory: | 存放*.exe和*.obj文件的目录。 |
| ④ Turbo C directory:C\TC | Install建立的默认目录。 |

(3) 从图1-6可以看出并未指定某一具体的目录，需要用户来指定。如果不指定，系统就会将它保存在当前工作目录中。此时选中第3行【Output directory:】后，按回车键，将弹出一个“输出文件目录”对话框，在此框内输入“C:\TC\C_OUT”，如图1-7所示，它表示要将.obj文件和.exe文件保存在“C:\TC\C_OUT”子目录中。按回车键后，对话框消失，此时发现“C:\TC\C_OUT”已经加入到第3行中，表示已经进行了该项的设置。



图1-7 保存文件到指定的目录

第4行中的【Turbo C directory:】指定了Turbo C系统文件所在的目录,由于我们的Turbo C就是保存在C盘的TC目录下,所以这一项不作修改。

(4) 进行了以上设置后,需要将这些信息保存起来。具体操作为:在上述设置完毕后按Esc键返回【Options】菜单,然后选择【Save Options】菜单项,并按回车键,在弹出的如图1-8所示的“配置文件”对话框内输入“C:\TC\TCCONFIG.TC”,按回车键后建立配置文件完成。



图1-8 保存所作设置

1.4.4 调试C程序的步骤

以“编写一个程序,用于在屏幕上输出两个数之和”为例,讲解调试程序的步骤。

(1) 编辑源文件。进入Turbo C 2.0集成开发环境后,在【File】菜单下选择【New】选项,系统的光标处于等待输入C源程序状态,输入如下的程序:

```
main()
{
    int x,y,sum;
    printf("input two numbers:\n");
    scanf("%d,%d",&x,&y);
    sum=x+y;
    printf("sum is %d\n",sum);
}
```

输入源程序后的屏幕如图1-9所示。



图1-9 在Turbo C 2.0主窗口中输入源程序

源程序输入后必须进行保存,保存的方法是在【File】菜单下选择【Save】选项,并