



21世纪高等院校新概念系列教材

C 语言程序设计

C YUYAN CHENGXU SHEJI

马秋菊 主编



中国宇航出版社

内容简介

C 语言程序设计

马秋菊 主编

江苏工业学院图书馆
藏书章

中国宇航出版社

内 容 简 介

本教材是由多年对 C/C++ 语言程序设计的教师和工程技术人员编写的,主要根据是 C 语言程序设计二级考试大纲、多年辅导学生参加二级考试的教学经验、在程序编写过程中经常遇到的实际问题、以及根据高职高专教育的特点,以必需、够用为度,加强内容的针对性和实用性。在编写时力求文字叙述简明,结构清晰、条理分明、概念清楚,使读者在学习本书后掌握 C/C++ 语言的基本概念和语法规则、阅读和编写 C/C++ 语言程序的基本方法、基本能力和初步技巧。

本书既可作为高职高专理工科类教学用书外,又可作为成教、夜大、职大、函大等大专层次的教学或自学参考用书,也可作为全国计算机等级考试(二级 C 语言)的参考用书。

图书在版编目(CIP)数据

C 语言程序设计/马秋菊主编. —北京:中国宇航出版社,2004.7

(21 世纪高等院校新概念系列教材)

ISBN 7-80144-834-0

I. C... II. 马... III. C 语言—程序设计—高等学校—教材 IV. TP312

中国版本图书馆 CIP 数据核字(2004)第 060474 号

出版 中国宇航出版社

地址 北京市阜成路 8 号 邮编 100830
(010)68768548

网址 www.caphbook.com / www.caphbook.com.cn

经销 新华书店

发行部 (010)68692341(含传真)

零售店 读者服务部 北京宇航文苑
北京市阜成路 8 号 北京市海淀区海淀大街 31 号
(010)68371105 (010)62579190(传真)

承印 北京京科印刷有限公司

版次 2004 年 7 月第 1 版
2004 年 7 月第 1 次印刷

开本 1 / 16

规格 787 × 1092

印张 17.25

字数 424 千字

书号 ISBN 7-80144-834-0 / TP·281

定价 28.00 元

本书如有印装质量问题可与发行部调换

前 言

C语言是目前世界上最为流行的计算机高级程序设计语言之一，它设计精巧，功能齐全，同时兼具高级语言和低级语言的特点，既适合于编写应用软件，又特别适合于编写系统软件，具有短小精悍、表达能力强、数据结构丰富、结构化好、目标代码质量高、可移植性好、能直接访问硬件等特点，因此更适合于应用在机电、自动化、通信等领域。

目前许多高等学校，不仅在计算机类专业开设了C语言课程，而且在非计算机类专业也开设了此课程。尤其是全国计算机等级考试和全国计算机应用技术证书考试都将C语言列入了考试范围。本书正是根据以上原因和读者的实际情况，力图体现以下编写特色：

1. 概念引入通俗易懂，内容编排合理，知识点循序渐进

每个知识点的引入都是由提出问题，分析问题，解决问题，到最后总结出概念，并推广到一般的编程方法，在讲解中强调通俗性。同时后面的知识点都以前面的知识点为基础，使知识系统化，降低教师讲解和学生学习的难度，注意了难点分散，循序渐进。

2. 概念准确，内容详略得当，重点突出

参与本书编写的人员都是长期从事C/C++语言程序设计教学、富有指导C/C++语言程序设计过级考试经验和使用本语言编写过大型应用程序的教师和科研人员，并且相互之间对教材的编写方法、概念的准确性、内容的详略程度、如何重点突出以及使读者更容易理解和抓住重点等进行了多次的研究和探讨。

3. 强化实践，重视应用

本教材力求使读者学完后不仅能懂C语言的语法、语义，更重要的是具备编程解决实际问题的能力。同时，教材更突出了实践环节，以应用为目的，书中不仅写入了一些经典实例和算法，而且还列举了一些来自实际工作和工程实践的实用案例，以期能够帮助读者掌握编程和程序调试的技巧。

本教材根据高职高专教育的特点，以必需、够用为度，加强内容的针对性和实用性。在编写时力求适用面广，文字叙述简明，结构清晰、条理分明、概念清楚，使读者在学习本书后能够掌握C/C++语言的基本概念和语法规则、阅读和编写C/C++语言程序的基本方法、基本能力和初步技巧。本书既可作为高职高专理工科类教学用书，又可作为成教、夜大、职大、函大等大专层次的教学或自学参考用书，也可作为全国计算机等级考试（二级C语言）的参考用书。

读者在学习C语言时，应注意以下几点：首先，读者要在阅读大量程序的基础上，必须自己亲自动手编写程序，逐步掌握和提高编程能力。其次，要重视上机训练，只有通过大量的上机练习，才能掌握C语言的实际应用能力。最后，C语言结构灵活，容易出错，必须通过大量的练习（包括上机练习）才能掌握C语言的某些特殊及细微之处。对一些似是而非的问题，读者应亲自上机实验。

本教材共14章，全书选择了大量的典型例题，每章后附小结和复习思考题，使读者能通过这些例题和复习思考题掌握C语言的语法特点和应用。书后附有常用附录，包括DOS的使用、ASCII码表、C语言中使用的关键字以及C语言中常用函数。

本书由马秋菊任主编，王学军、王贺艳任副主编，宋汉珍副教授担任主审并对书中内容提出了宝贵的意见。第1章、第2章、第9章、第12章和第14章由马秋菊编写，第3章、第4章由田晓霞编写，第7章、第11章由高丽编写，第5章、第6章由王贺艳编写，第8章、第10章和第13章由王学军编写。全书由马秋菊统稿，由王学军和王贺艳协助进行了认真校对。

尽管各位作者具有教学和科研经验，但是书中难免有错误或不当之处，敬请专家、同仁和广大读者批评指正。

编者

目 录

第1章 C语言基本知识	1	习题	25
1.1 C语言发展简介	1	第3章 顺序程序设计	27
1.1.1 C语言的起源	1	3.1 C语句概述	27
1.1.2 C语言的标准化	1	3.2 程序的三种基本结构	28
1.2 C语言的特点	2	3.2.1 三种基本结构的流程图	28
1.3 C语言程序的基本结构	2	3.2.2 N-S图表示	30
1.3.1 C程序的构成	2	3.3 赋值语句	31
1.3.2 书写规则	4	3.4 数据输入输出的概念及在C语言中 的实现	31
1.4 C语言程序的开发过程	4	3.5 字符数据的输入输出	32
1.4.1 程序开发的一般步骤	4	3.5.1 单个字符的输出——putchar() 函数	32
1.4.2 上机操作	5	3.5.2 单个字符的输入——getchar() 函数	33
1.4.3 Turbo C使用简介	7	3.6 格式输出与输入	33
1.4.4 菜单命令简介	8	3.6.1 格式化输出——printf()函数	33
习题	10	3.6.2 格式化输入——scanf()函数	38
第2章 数据类型、运算符与表达式	12	3.7 顺序结构程序设计举例	42
2.1 基本数据类型	12	习题	44
2.1.1 整型(int)	12	第4章 选择结构	47
2.1.2 浮点型(float)	13	4.1 关系运算符和关系表达式	47
2.1.3 字符型数据(char)	13	4.1.1 关系运算符及其优先次序	47
2.2 常量和变量	13	4.1.2 关系表达式	48
2.2.1 常量和符号常量	13	4.2 逻辑运算符和逻辑表达式	48
2.2.2 变量	16	4.2.1 逻辑运算符及其优先次序	48
2.2.3 变量初始化和赋初值	19	4.2.2 逻辑表达式	49
2.3 运算符和表达式	20	4.3 if语句	51
2.3.1 算术运算符和表达式	20	4.3.1 if语句的三种形式	51
2.3.2 条件运算符“?:”	22	4.3.2 if语句的嵌套	54
2.3.3 逗号运算符和逗号表达式	22	4.3.3 条件运算符	55
2.3.4 sizeof运算符	23	4.4 switch语句	57
2.3.5 联合(组合)操作	23	4.5 应用举例	60
2.3.6 “&”和“*”运算符	23		
2.4 各类数值型数据的混合运算	23		
2.5 Turbo C运算符的优先级	24		

习题	66	7.3 函数的定义及调用	112
第 5 章 循环结构程序设计	69	7.3.1 函数定义的一般形式	112
5.1 概述	69	7.3.2 函数调用	114
5.2 循环结构的三种语句	69	7.3.3 函数的参数和函数的值	115
5.2.1 while 语句	69	7.4 函数与数组	117
5.2.2 do-while 语句	71	7.4.1 数组元素作函数实参	117
5.2.3 for 语句	73	7.4.2 数组名作为函数参数	118
5.2.4 三种循环语句的比较	77	7.5 函数的嵌套与递归调用	121
5.3 循环的嵌套	78	7.5.1 函数的嵌套调用	121
5.4 break 语句与 continue 语句	80	7.5.2 函数的递归调用	123
5.4.1 Break 语句	80	7.6 变量的作用域及存储类型	124
5.4.2 continue 语句	81	7.6.1 局部变量和全局变量	124
5.5 程序举例	82	7.6.2 变量的存储类别	128
习题	86	7.6.3 内部函数和外部函数	133
第 6 章 数组	88	习题	135
6.1 一维数组	88	第 8 章 指针	137
6.1.1 一维数组的定义	88	8.1 指针概述	137
6.1.2 一维数组的初始化	89	8.2 指针变量及运算	137
6.1.3 一维数组元素的引用	89	8.2.1 指针变量	137
6.2 二维数组	91	8.2.2 指针变量的运算	138
6.2.1 二维数组的定义	91	8.3 指针与数组之间的关系	141
6.2.2 二维数组的初始化	92	8.3.1 一维数组的指针	141
6.2.3 二维数组元素的引用	93	8.3.2 多维数组的指针	142
6.3 字符数组	95	8.3.3 字符串指针变量的说明 和使用	144
6.3.1 字符数组的定义	95	8.3.4 指针数组的说明与使用	147
6.3.2 字符数组的初始化	96	8.4 指针与函数之间的关系	148
6.3.3 字符数组的引用	96	8.4.1 函数指针变量	148
6.3.4 字符数组与字符串	97	8.4.2 指针型函数	149
6.3.5 字符数组的输入与输出	98	8.4.3 main 函数的参数	150
6.3.6 字符串处理函数	99	8.5 指向指针的指针变量	151
6.4 程序举例	104	习题	153
习题	106	第 9 章 结构体和共用体	156
第 7 章 函数	109	9.1 结构体类型	156
7.1 概述	109	9.1.1 结构体类型的定义	156
7.1.1 函数的概念	109	9.1.2 定义结构体类型变量的方法	158
7.1.2 函数的分类	109	9.1.3 结构变量的引用与初始化	159
7.2 系统函数	110	9.2 结构体数组	160
7.2.1 数学函数	110	9.2.1 结构体数组的定义	161
7.2.2 其他系统函数	111		

9.2.2 结构体数组的初始化	161	12.1.1 位运算及其运算符	207
9.2.3 结构体数组的引用	162	12.1.2 位运算的复合赋值运算符	210
9.3 指向结构体类型数据的指针	164	12.1.3 应用举例	210
9.3.1 指向结构体变量的指针	164	12.2 位段简介	212
9.3.2 指向结构体数组的指针	165	12.2.1 引入位段的原因	212
9.3.3 用结构体变量和指向结构体 的指针作函数参数	166	12.2.2 位段的概念与定义	212
9.3.4 链表	167	12.2.3 说明	213
9.4 共用体类型	172	习题	214
9.4.1 共用体类型的定义	172	第 13 章 图形	215
9.4.2 共用体变量和数组的定义	172	13.1 C 语言图形概述	215
9.4.3 共用体变量和数组的引用	173	13.2 图形模式和初始化	215
9.5 枚举类型	175	13.3 基本图形函数	219
9.6 用 typedef 定义类型	177	13.3.1 画点函数	219
习题	178	13.3.2 画线函数	219
第 10 章 编译预处理	180	13.3.3 图形的填充	222
10.1 预处理	180	13.3.4 有关图形窗口和图形屏幕 操作函数	224
10.1.1 概述	180	13.3.5 图形模式下的文本输出	224
10.1.2 宏定义	180	习题	227
10.2 文件包含	184	第 14 章 面向对象的程序设计	229
10.3 条件编译	185	14.1 概述	229
习题	187	14.2 C++ 对 C 的非面向对象扩充	230
第 11 章 文件	189	14.3 C++ 语言	240
11.1 文件概述	189	14.3.1 面向对象程序设计的基本 术语	240
11.1.1 磁盘文件名	189	14.3.2 C++ 主要特征	241
11.1.2 磁盘文件的分类	189	习题	253
11.2 文件指针及文件的操作	190	附录 A 标准 ASCII 字符集	255
11.2.1 文件型指针及文件型指针 的定义	190	附录 B C 语言中的关键字	256
11.2.2 文件的打开与关闭函数	191	附录 C 运算符和结合性	257
11.2.3 文件的顺序读写	193	附录 D C 语言的库函数	258
11.2.4 文件的其他函数	202	附录 E 常用 DOS 命令	263
习题	204	参考文献	267
第 12 章 位运算	207		
12.1 位运算	207		

第 1 章 C 语言基本知识

20 世纪 60 年代初期流行的一些高级语言如: ALGOL60、FORTRAN、COBOL 等, 这些语言的数据类型单调, 缺乏一种规范化的设计方法, 因此程序很难编得更大。随着计算机技术的发展, 程序设计语言也在不断更新, 20 世纪 70 年代兴起了结构化程序设计, 强调程序的模块性, C 语言就是这种结构化程序设计语言的代表作之一。

本章简单介绍了 C 语言的发展和 C 语言的特点, 特别是结构化的编程特点、基本词法及 C 语言的结构框架, 并且通过范例, 介绍 C 程序的格式和书写规则。

1.1 C 语言发展简介

1.1.1 C 语言的起源

Turbo C 是美国 Borland 公司的产品, Borland 公司是一家专门从事软件开发、研制的大公司。

在 C 语言出现之前, 包括操作系统在内的一些系统软件, 主要用汇编语言编写。汇编语言是一种低级语言。最接近于硬件, 可以对硬件施加控制和操作, 执行速度最快, 能充分发挥机器的潜能, 汇编语言程序这些特点都是开发系统软件不可缺少的。但是, 汇编语言编程难度大, 调试也较难, 可读性和移植性都较差。与此相反, 高级语言却有着编程容易、调试方便、可读性和移植性好的优点。C 语言就是在这一背景下问世的。

1972 年贝尔实验室的布朗·W·卡尼汉 (Brian.W.Kernighan) 和丹尼斯·M·利奇 (Dennis.M.Ritchie) 两人 (简称 K&R), 在原有语言的基础上, 加进了许多程序设计功能, 提供了更丰富的数据类型, 并以“C”命名该语言。随后, K&R 两人合作, 用 C 改写了 UNIX 操作系统。应该说, 这时的 C 语言还是与 UNIX 操作系统息息相关的, 是作为 UNIX 的程序设计语言而开发的。而后 C 语言又从 UNIX 的程序设计语言发展成为通用的程序设计语言了。

C++是在 C 基础上扩充了面向对象的程序设计语言。C++采取与 C 完全兼容的策略, C 是 C++的子集。目前 C 和 C++已经成为最流行的主流程序设计语言。

1.1.2 C 语言的标准化

1978 年, K&R 出版了《The C Programming Language》一书, 建立了 C 语言标准, 因为微型计算机的普及, 各种 C 编译系统实用版本纷纷出现, 为了使 C 程序之间能够交流。1983 年夏, 美国标准化协会 (ANSI) 成立了一个委员会, 制定了 C 语言标准, 称为 ANSI C, 该标准于 1987 年进行了完善, 目前流行的各种 C 编译系统都是以它为基础的。

目前, 在微型计算机上广泛使用的 C 编译系统有 Turbo C、Microsoft C、Quick C 等, 都

遵从 ANSI C 标准。不同的编译系统、不同版本可能有些差异，因此在具体使用时需参阅有关手册。

本书内容遵从 ANSI C 标准，实际使用的 C 语言编译系统采用 Turbo C 2.0 版。

1.2 C 语言的特点

C 语言的主要特点如下：

(1) C 语言是“中级”程序设计语言。它把高级语言面向用户与低级语言操纵硬件的能力有效地结合起来，兼有两方面的优点。与其他高级语言项比较，C 语言更接近于硬件，能直接访问内存地址，进行字、字节和位操作，目标代码执行速度非常快，因此适合于编写系统软件和实时控制、图形处理、文字编辑等实用程序；同时它又具有一般高级语言的易读、易写、易查错、易维护的特点，因此也能编写科学计算、数据处理等方面的应用程序。

(2) C 语言是一种结构化程序设计语言。结构化程序设计主张程序模块化，一个大程序由若干模块构成，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。

(3) C 语言数据类型丰富，结构化好。C 语言基本类型有字符型、整型、实型，在此基础上，用户还可构造各种类型，如数组、结构和联合等。

(4) C 语言运算符特别丰富，表达能力强。通过运算符构成的 C 表达式灵活、多样、功能非常强。除一般高级语言普遍拥有的四则运算、逻辑运算符外，它还有以二进制位 (bit) 为单位的位运算等。

(5) C 语言移植性好。C 语言程序基本上可以不加修改就能从一种机器移植到另一种机器上。

(6) C 语言适用范围大。C 语言适合于多种操作系统，如 DOS、UNIX，也适用于多种机型。

1.3 C 语言程序的基本结构

本节先介绍几个简单的 C 语言程序，再结合这些程序对其语法进行说明，以便使读者对 C 语言程序的基本结构有一个概括的了解。程序结构是指程序的组织形式，是程序的骨架，程序本身可以改变，但是程序的组织形式不能随意变动。

1.3.1 C 程序的构成

【例 1.1】最简单的 C 语言程序，用于显示字符串“Hello, Beijing 2008!”。

```
#include <stdio.h> /*用预处理命令指定一个包含头文件*/
main() /*主函数*/
{ /*函数体开始标记*/
    printf("Hello, Beijing 2008!\n"); /*调用 printf 库函数，在屏幕输出结果*/
}
```

将【例 1.1】的 C 语言源程序，经过编辑、编译和链接，运行结果是在光标所在处显示如下字符串，并将光标移到下一行行首。

Hello, Beijing 2008!

其中，main 表示“主函数”，每一个 C 程序有且只有一个 main 函数。花括号括起来的部分是 main 函数的函数体。本例中的函数体内只有一个输出语句，printf 是 C 语言编译系统提供的一个标准输出函数，printf 的作用是把圆括号中用双引号“”括起来的字符串原样显示输出。“\n”是换行符，其作用是显示输出字符串后将光标移到下一行的行首位置。每个语句最后有一个分号。第一行#include <stdio.h>为编译预处理命令行，表示把头文件“stdio.h”包含到程序中，这样可以使使用系统提供的 printf 等库函数。

【例 1.2】从键盘输入两个整数，求它们的最大值。

```
#include "stdio.h"
main() /*主函数 main 函数*/
{ int a,b,c; /* 变量定义 */
  scanf("%d,%d",&a,&b);
  c=max(a,b);
  printf("max=%d\n",c); /* 显示输出 */
}
int max(int x, int y) /*定义 max 函数，函数返回值为整型，x、y 为形式参数*/
{ int z; /* max 函数体中用到的变量也要定义*/
  if(x>y) z=x; /* 求 x 和 y 的最大值并赋给 z */
  else z=y;
  return(z); /*将 z 的值返回给主调函数*/
}
```

程序运行情况如下：8, 9<回车>

max=9

在【例 1.2】程序中，每一行尾部/*……*/部分为程序的注释部分，只起注解作用。第三行定义了 a、b、c 三个变量，它们是整型变量，可以存放整数。第四行 scanf 也是 C 语言编译系统提供的输入函数，它能接收用户从键盘输入的两个整数，并分别保存到变量 a 和 b 中。“%d, %d”是格式控制符，指明从键盘输入整数，&a, &b 是输入表列，指明输入对象。

本程序包括两个函数：main 函数和 max 函数，main 函数调用 max 函数，故 main 函数称为主调函数，max 函数称为被调函数。

在 printf 输出语句，双引号内其他字符原样输出，当遇到一个输出格式控制符“%d”时，就要到双引号的后面寻找一个对应的输出对象，把该输出对象以整数形式显示输出。

通过上述一些简单的例子可以看到：

1. C 语言程序是由函数构成的

一个 C 程序一定有且只有一个主函数（main 函数），它可以只由一个 main 函数组成，也可以由一个 main 函数和其他一个或多个函数组成，因此，函数是 C 程序的基本单位。

2. 一个函数由两部分组成

(1) 函数的说明部分包括函数名、函数（返回值的数据）类型、函数参数（形参）名、形参（数据）类型。

(2) 函数体即函数说明部分后面的花括号{……}内的部分。函数体中一般应包括：

① 变量定义。Turbo C 的变量在使用之前必须先定义其数据类型，未经定义的变量不能使用。

② 执行部分。它由若干语句组成，指明在函数中要执行的一些操作，函数的作用和功能主要由函数体中的操作语句决定。函数体甚至可以为空。

3. 每个语句和数据定义的最后必须有一个分号，分号是 C 语句的必要组成部分。

4. 可以用/* */对程序中的任何部分加以注释。

注释的作用是：一是提高程序的可读性，以方便自己或别人阅读程序；二是用于程序调试，当不希望源程序中的某一部分起作用时，可以在该部分的前后分别加上“/*”和“*/”。

1.3.2 书写规则

1. C 语言严格区分大小写英文字母，习惯上用小写字母，而且关键字必须采用小写。但是，某些情况下也可以用大写字母，一般大写字母用来定义符号常量。

2. C 源程序也是由一个个语句组成的。语句用“;”作为结束符。语句后的分号不可少。

3. C 语言中的大括号“{”和“}”必须配对使用。“{”和“}”用来：

(1) 标定一个函数的范围；

(2) 标定一个复合语句的范围。

例如，【例 1.1】中的第 3 行的“{”号和第 7 行的“}”号，两者配对标定了 main（）模块的作用范围。

4. 可读性要强。一个程序，首先保证是正确的，其次是可读性强。建议做到：

(1) 每个语句最好占一个程序行位置，即一行只写一个语句。

(2) 合理地使用注释。

1.4 C 语言程序的开发过程

1.4.1 程序开发的一般步骤

用 C 语言书写的程序又称为 C 的源程序，它是不能直接运行的，必须生成可执行程序，即需要按照图 1-1 所示各个步骤完成开发工作。

1. C 源文件的编辑

选则适当的文本编辑程序或直接在 Turbo C（简称 TC）集成开发环境中的“编辑器”下通过键盘输入源程序，并将源程序取一文件名并以“.C”为扩展名存盘。

2. 编译 C 源文件

编译是由 C 语言编译系统（编译程序）对源文件进行词法和语法检查和翻译成目标文件的过程，并生成默认扩展名为“.OBJ”的目标程序文件。目标文件与源文件的主文件名同名，但扩展名不一样。

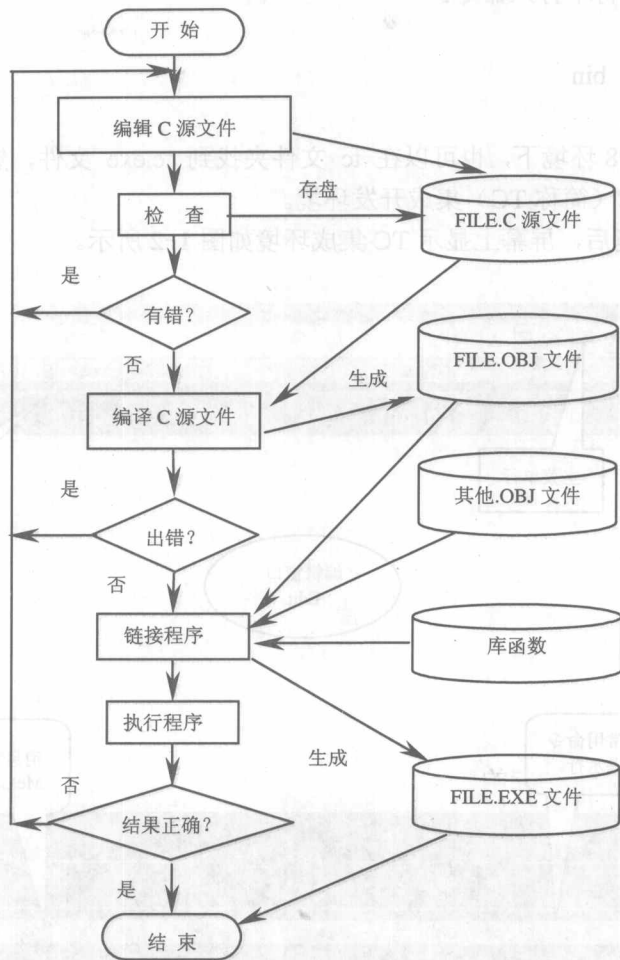


图 1-1 开发和使用 C 语言程序的基本过程

若一个大的 C 程序由多个 C 源文件组成，则各个源文件要分别单独编译。

3. 链接过程

将编译后生成的目标文件（.OBJ 文件）经过链接后生成可执行文件（扩展名为“.EXE”的文件）。可执行文件通常与源文件同名，只是扩展名不同。

4. 执行过程

源程序经过编译、链接后，生成相应可执行的 EXE 文件，这时就可以脱离编译系统而独立存在。在操作系统支持下，键入可执行文件名，程序就可以执行了。

若程序运行结果不正确，通常是属于逻辑性错误。这就要求程序员认真检查程序，再经过“编辑→编译→链接→执行”全过程，直到程序运行结果正确为止。

1.4.2 上机操作

为了使读者尽快掌握 Turbo C 2.0 环境的使用，通过例子将常用的菜单或命令简述如下。

1. 编辑（输入）并保存一个 C 语言源程序

在 DOS 提示符下打入命令：

C: \>CD TC

C: \TC>cd bin

C: \tc\bin> tc

在 Windows 98 环境下，也可以在 tc 文件夹找到 tc.exe 文件，然后用鼠标双击该文件名也可进入 Turbo C（简称 TC）集成开发环境。

进入集成环境后，屏幕上显示 TC 集成环境如图 1-2 所示。

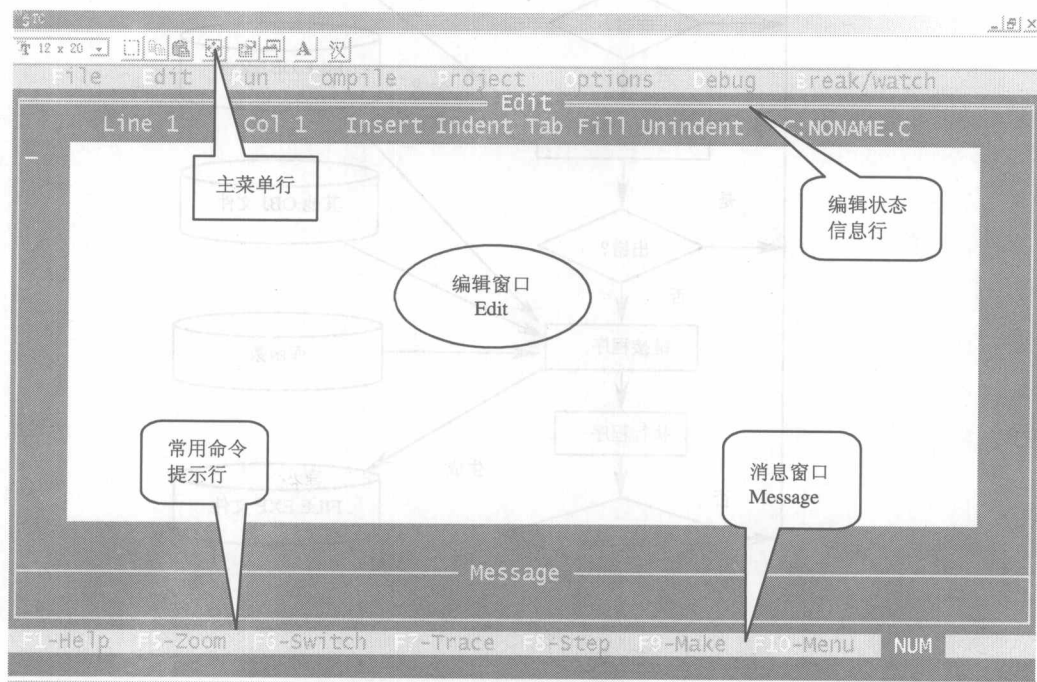


图 1-2 TC 窗口

在编辑窗口输入下面的程序：

【例 1.3】在屏幕指定的位置上输出一个字符串。

```
#include <stdio.h>
```

```
#include <conio.h>
```

```
main()
```

```
{ clrscr();
```

```
gotoxy(15,20);
```

```
printf("*****\n");
```

```
getchar();
```

```
return 0; }
```

输入完之后，用 Alt+F 键激活主菜单中的 File 菜单，用光标↓键将光标下移 Write to（换名存盘）处，按回车，取名 EX1_3.C，存盘。若程序较长，则在输入时随时存盘，中间存盘时可以用 F2 键，也可以用 Alt+F 键，再选 Save（存盘）命令。这是一个良好的习惯！

按 F1 键可激活在线帮助活动窗口获取 TC 环境提供的帮助。

2. 编译、链接——单个源程序文件

选择并执行 **Compile | Make EXE File** 项 (快捷键: F9), 则 TC 将自动完成对当前正在编辑的源程序文件的编译、链接, 并生成可执行文件 EX1_3.EXE。

如果源程序有语法错误, 系统将在屏幕中央的“**Compiling**”(编译)窗口底端提示“**Error: Press any key**”(错误: 按任意键)。

此时, 按空格键, 屏幕下端的“**Message**”(消息)窗口被激活, 显示出错(或警告)信息, 光带停在第一条消息上。这时“**Edit**”(编辑)窗口中也有一条光带, 它总是停在编译错误的源代码中的相应位置。

当用上、下键移动消息窗口中的光带时, 编辑窗口中的光带也随之移动, 始终跟踪源代码中的错误位置。

3. 运行与查看结果

(1) 运行当前正在编辑的源程序文件。选择并执行 **Run | Run** 项 (快捷键: ^F9), 程序运行结束后, 仍返回到编辑窗口。

(2) 查看运行结果。选择并执行 **Run | User Screen** 项 (快捷键: Alt+F5)。

上述程序分析:

- `clrscr()` 是清屏函数, 起清屏作用。
- 光标定位函数 `gotoxy(x, y)`, 该函数将光标定位在屏幕坐标 x, y 位置。
- 格式输出函数 `printf()`, 该函数按指定的“格式”在屏幕上输出数据。
- 从键盘读一个字符函数 `getchar()`, 作用是等待用户从键盘上输入一个字符。

`#include <stdio.h>`和`#include <conio.h>`是二条预处理命令, 是因为上述函数分别包含在头文件“`stdio.h`”和“`conio.h`”中。

运行结果为在屏幕的<15,20>坐标处显示 *****。

1.4.3 Turbo C 使用简介

Turbo C 是 1987 年 Borland 公司推出的广受欢迎的 C 编译系统, 它是 DOS 操作系统支持下的软件, 在 Windows 98 环境下, 可以在 DOS 窗口下运行。它是一个带有编辑器、编译器、链接器、调试器和一些其他实用程序的集成环境软件。

1. Turbo C 的安装

插入安装盘于 A 驱动器, 键入 `A: \>INSTALL <CR>` (<CR> 代表回车键)

此后根据提示, 插入其他盘, 安装成功后, 在 C 盘上自动创建 TC 子目录以及 TC 下的三个子目录 LIB、INCLUDE 和 BIN。LIB 子目录中存放库文件, INCLUDE 子目录中存放所有头文件。

也可以用 COPY 命令直接拷贝 Turbo C 系统到 C 盘上。

2. Turbo C 的启动

按 1.4.2 小节中提供的方法启动 TC 后, 屏幕出现如图 1-2 所示的集成环境窗口。

其中顶上一行为 Turbo C 2.0 主菜单, 包括 File、Edit、Run、Compile、Project、Options、Debug、Break/watch 等项, 中间窗口为编辑窗口, 接下来是信息或消息窗口, 最底下一行为常用命令或热键(hot key)提示行。这四个窗口构成了 Turbo C 2.0 的主屏幕, 以后的编程、编

译、调试以及运行都将在这个主屏幕中进行。其中 Edit(编辑)和 Message(消息)这两个窗口顶端横线为双线显示的,表示该窗口是活动窗口。

编辑窗口的顶端为状态行,它给出了正在编辑文件的信息:Line Col Insert 等,其中:

- Line 1 Col 1: 显示光标所在的行号和列号,即光标位置。
- Insert: 表示编辑状态处于“插入”。当处于“改写”状态时,此处为空白。
- C: NONAME.C: 显示当前正在编辑的文件名。显示为“NONAME.C”时,表示用户尚未给文件命名。

屏幕底端是 7 个功能键的说明,以及 Num Lock 键的状态(显示“NUM”时,表示处于“数字键”状态;空白,表示“控制键”状态)。

在 Edit 环境下通过按 F1 键取得在线帮助,了解这些命令的使用方法,以便加快编辑和修改程序的速度。TC 集成环境的常用编辑命令有:

(1) 光标移动

←: 光标左移一个字符;

→: 光标右移一个字符;

↑: 光标上移一行;

↓: 光标下移一行;

PageUp: 向前翻页;

PageDn: 向后翻页;

Home: 将光标移到所在行的开始;

End: 将光标移到所在行的结尾。

(2) 插入、删除

Insert: 打开/关闭插入功能;

Del: 删除光标所在处的一个字符;

Backspace: 删除光标所在处的前一个字符;

(3) 块命令(对连续的一个文本块进行操作)

Ctrl+Y: 删除光标所在的一行;

Ctrl+QY: 删除至行尾;

Ctrl+T: 删除光标所在处的一个词;

Ctrl+KB: 设置块开始;

Ctrl+KK: 设置块结尾;

Ctrl+KV: 块移动;

Ctrl+KC: 块拷贝;

Ctrl+KY: 块删除;

Ctrl+KR: 读文件;

Ctrl+KW: 存文件;

Ctrl+KP: 块文件打印。

3. 退出 Turbo C

文件编辑完和被修改后要存盘退出。退出 TC 有两种方法:

1) 菜单法: FileQuit(先选择 File 主项,再选择并执行 Quit 子项)。

2) 快捷键法: Alt+X(先按下 Alt 键并保持,再按字母键 X,然后同时放开)。

1.4.4 菜单命令简介

主菜单提供了 8 种菜单项,除 Edit 外,其他菜单项都是下拉式菜单,有的还有多级子菜单。用 F10 键可以从窗口切换到主菜单或相反画面,按回车键为确认选择子菜单,或者用 Alt 键加上某项中第一个字母,也可进入该项的子菜单中。

1. File(文件)菜单

File 菜单的子菜单如图 1-3 所示,共有 9 项,分别叙述如下:

(1) Load(加载): 装入一个文件,该项的热键为 F3。

(2) Pick (选择): 将最近装入编辑窗口的 8 个文件列成一个表让用户选择。其热键为 Alt+F3。

(3) New (新文件): 新建文件。

(4) Save (存盘): 将编辑区中的文件存盘, 其热键为 F2。

(5) Write to (换名存盘): 可由用户给出文件名将编辑区中的文件存盘。

(6) Directory (目录): 显示目录及文件, 并供由用户选择。

(7) Change dir (改目录): 用户可以改变当前默认目录。

(8) Os shell (暂时退出): 暂时退出 TC 到 DOS 提示符下, 此时可以运行 DOS 命令, 若想回到 TC 中, 只要在 DOS 状态下键入 EXIT 即可返回 TC。

(9) Quit (退出): 退出 TC, 返回到 DOS 操作系统中, 其热键为 Alt+X。

说明: 以上各项可用光标键移动色条进行选择, 回车则执行。也可用每一项的第一个大写字母直接选择。若要退到主菜单或从它的下一级菜单列表框退回均可用 Esc 键, TC 所有菜单均采用这种方法进行操作, 以下不再说明。

2. Edit (编辑) 菜单

按 Alt+E 可进入编辑菜单, 若再回车, 则光标出现在编辑窗口, 此时用户可以进行文本编辑。在编辑 (Edit) 状态下, 可用 F1 键获得有关编辑方法的帮助信息。

(1) 与编辑有关的功能键如下:

F1: 获得 Turbo C 2.0 编辑命令的帮助信息;

F5: 扩大编辑窗口到整个屏幕;

F6: 在编辑窗口与信息窗口之间进行切换;

F10: 从编辑窗口转到主菜单。

有关编辑命令已经在上一小节中介绍过, 这里不再重复。

(2) TC 在编辑文件能够自动缩进, 即光标定位和上一个非空字符对齐。在编辑窗口中, Ctrl+OL 为自动缩进开关的控制键。

3. Run (运行) 菜单

按 Alt+R 键可进入 Run 菜单, 如图 1-4 所示, 该菜单有以下各项:

(1) Run (其热键为 Ctrl+F9): 对当前编辑的源文件进行编译、链接和运行, 运行后仍回到原来状态。若想观察输入输出信息, 需按 Alt+F5 键。

(2) Program reset (程序重启): 中止当前的调试, 其热键为 Ctrl+F2。

(3) Go to cursor (执行到光标处): 调试程序时使用, 其热键为 F4。

(4) Trace into (跟踪): 单步执行, 可以跟入下一级子函数, 其热键为 F7。

(5) Step over (步进): 单步执行, 不跟踪进入下一级函数。其热键为 F8。

(6) User screen (用户屏幕): 显示运行时在屏幕上显示的结果。其热键为 Alt+F5。

4. Compile (编译) 菜单

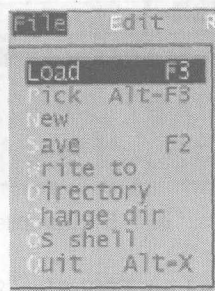


图 1-3 File 子菜单

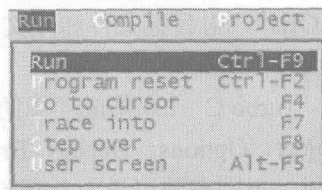


图 1-4 Run 子菜单