# C程序设计教程

C CHENGXU SHEJI JIAOCHENG

主编 朱友红
主审 杨 威

山西科学技术出版社

# C程序设计教程

主编　朱友红
主审　杨　威

委员(按姓氏笔划为序)
丁慧娴　史春秀
朱友红　苏　钦
原海梅　贾智智



山西科学技术出版社

**图书在版编目(CIP)数据**

C程序设计教程/朱友红主编,杨威主审 .—太原:山西科学技术出版社,2002.7

ISBN 7-5377-2026-6

Ⅰ.C… Ⅱ.朱… Ⅲ.C语言—程序设计—教材

Ⅳ.TP312

中国版本图书馆CIP数据核字(2002)第044697号

**C程序设计教程**

朱友红 主编

杨 威 主审

*

山西科学技术出版社出版发行(太原建设南路15号)

山西新华印业有限公司人民印刷分公司印刷

*

开本:787×1092 1/16 印张:24.5 字数:614千字

2002年7月第1版 2002年7月山西第1次印刷

印数:1—5 000册

*

ISBN 7-5377-2026-6

T·333 定价:29.80元

# 前　言

C语言自推出以来，以其功能丰富、表达能力强、使用方便灵活、目标程序效率高、可移植性好等特点得到了迅速地推广使用，C程序设计已成为计算机专业一门重要的专业基础课。随着我国计算机应用的进一步普及和教育水平的提高，大多数非计算机专业也开始开设C语言课程。但随着计算机等学科教学计划的修改，大多数高等院校都取消了以往的入门语言，而直接用C语言开设第一门程序设计课程。由于C语言牵涉到的概念比较复杂，规则繁多，使用灵活，容易出错，给学生的学习造成了一定的学习困难。根据这一教学特点，我们编写了这本教材，其内容选取和结构编排就是针对从未学过任何高级程序设计语言的初学者。本书具有以下特点：

1. 不要求读者必须具有一门计算机高级语言的基础，本书的体系结构是针对初学者特点精心安排的。难点分散，循序渐进，通俗易懂，减小了学习难度。

2. 在各章中都附有适量习题，题型多样，覆盖面广，有利于学生巩固所学知识，提高分析程序、编写程序的能力。

3. 在教材中增加了等级考试辅导内容，使学生参照等级考试水平，对自己的学习成果有个客观评价。

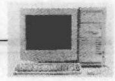
本书的内容丰富、细节很多，介绍的只是C语言的基本部分。读者在学完本书后，对以后编程中出现的复杂问题，还需参考其它的高级编程书籍。

本书共分十四章，其中的第1、3、4章由丁慧娴编写，第5、6、7章由史春秀编写，第8章由原海波编写，第11章由贾睿智编写，第10、13章由朱友红编写，第2、9、12、14章由苏钦编写。各位编者在共同讨论提纲的基础上，分别收集材料并进行编写，最后由朱友红、苏钦统稿，杨威副教授审阅了全稿。书中还引用了一些专家、学者的成果，在此表示衷心感谢。

衷心地希望本书的出版对读者有所帮助，由于编者水平有限，书中难免存在疏漏之处，诚请广大读者不吝指正。

编　者

# 目　录



# 第一章　C 语言概述

## §1.1　计算机语言概述

计算机是信息处理的有力工具，但目前的计算机尚不能直接理解人类的自然语言。人们要指挥计算机运行，不得不使用特定语言——计算机语言，这样才能与之打交道。使用计算机进行信息处理时，通常要为待处理的问题预先编排好明确的工作步骤，把预定的方案用特定的语言表示出来，即编写程序，然后输入计算机，在计算机软、硬件系统的支持下，才能按程序的规定自动地进行信息处理工作。这种用计算机系统所能接受的语言编写程序的过程称为程序设计。

程序设计语言按其发展程度和应用级别可以分为机器语言、汇编语言、高级语言。

### 1.1.1　机器语言

机器语言是计算机系统所能识别的、直接供机器使用的程序设计语言。机器语言中的每一条语句实际上就是由二进制编码表示的基本操作。例如，10001011 代表加一个数，10010001 代表存数等等，这种计算机能够直接识别与执行的基本操作，如取数、送数、加、减等，是由计算机硬件在设计和制造时所确定的。每台计算机在设计时，对于能执行什么样的指令、能执行多少条指令、怎样表示这些指令都有自己的规定，这些指令的总和称为该机器的指令系统。由二进制代码按照一定规则组成的指令系统就是该机器的机器语言。用机器语言编写的程序称为机器语言程序。

使用机器语言，计算机可以直接识别，占用内存少，执行效率高。但它存在两方面严重的缺点：其一是机器语言难懂、难记，编程工作十分繁琐，易于出错，程序直观性差，不便修改；其二是机器语言不能移植，没有通用性。由于不同的机型有不同的指令系统，因此，用某一种计算机语言编写的程序，不能在另一种计算机上运行。

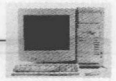
机器语言的缺点极大地限制了计算机的推广应用，目前，除在某些单板机上使用外，大容量的计算机系统都不直接使用机器语言。

### 1.1.2　汇编语言

汇编语言是符号化的、面向机器的低级程序设计语言。

为了克服使用机器语言的困难，在 50 年代初出现了汇编语言。在汇编语言中，用助记符代表机器指令中的操作码。助记符直接反映指令的功能和主要特征，便于理解和记忆。例如，用 ADD 表示加法，用 SUB 表示减法等。

汇编语言与机器语言之间的关系基本上是一对一的。它在格式和内容上均类似于机器语言，因此它仍是面向机器的语言，即每种类型的计算机有自己特有的汇编语言。要想用某种汇编语言编写程序，必须有足够的专业训练，熟悉具体机器指令系统。反过来，也正因为汇编语



言依赖于机器，使用它可以编写出结合机器特点的高质量的程序来。

汇编语言虽然便于理解和记忆，但是，计算机硬件只能识别机器指令，执行机器指令，对于助记符表示的汇编指令是不能执行的。用汇编语言编写的程序要执行的话，必须用一个由专业人员编写的翻译程序将汇编语言程序翻译成机器语言程序，用于翻译的程序称为汇编程序（汇编系统）。

汇编程序将用符号表示的汇编指令码翻译成为与之对应的机器语言指令码。用汇编语言编写的程序称为源程序，变换后得到的机器语言程序称为目标程序。

### 1.1.3　高级语言

从20世纪50年代中期开始，到70年代陆续产生了许多“高级算法语言”。这些高级算法语言中的数据用10进制来表示，语句用较为接近自然语言英文的字来表示。例如，A = B + 7表示把B + 7的值赋给A，PRINT A表示打印输出A的值等等。它们比较接近于人们习惯用的自然语言和数学表达式。因此，称为高级语言。高级语言具有较大的通用性，尤其是有些标准版本的高级算法语言，在国际上都是通用的，用高级语言编写的程序能使用在不同的计算机系统上。

对于高级语言编写的程序，计算机同样是不能识别和执行的。要执行高级语言编写的程序，首先要将它翻译成计算机能够识别的二进制指令，然后供计算机执行。

一般将用高级语言编写的程序称为源程序，把由源程序翻译成的机器语言程序或汇编语言程序称为目标程序。计算机将源程序翻译成机器指令时，通常分为两种翻译方式：一种为编译方式；另一种为解释方式。所谓编译方式，首先是把源程序翻译成等价的目标程序，然后再执行此目标程序。解释方式是把源程序逐句翻译，翻译一句执行一句，边翻译边执行，解释程序不产生将被执行的目标程序，而是借助于解释程序直接执行源程序本身。

常用的高级语言有：

●FORTRAN语言在1954年提出，1956年实现，适用于科学和工程计算，目前应用面还较广。

●PASCAL语言是结构化程序设计语言，适用于教学、科学计算、数据处理等，目前逐渐被C语言所取代。

●C语言程序简练、功能强，适用于系统软件、数值计算、数据处理等，目前成为高级语言中使用得最多的语言之一。现在较常用的C语言，是Visual C++面向对象的程序设计语言。

●BASIC语言是初学者语言，简单易学，人机对话功能强。至今BASIC语言已有许多高级版本，尤其Visual Basic是面向对象的程序设计语言，给非计算机专业的广大用户在Windons环境下开发软件带来了福音。

●Java语言是一种新型的跨平台分布式程序设计语言。Java以它简单、安全、可移植、面向对象、多线程处理等特性引起世界范围的广泛关注。Java语言是基于C++的，其最大特色在于“一次编写，处处运行”。但Java语言编写的程序要依靠一个虚拟机VM（Virtual Machine）才能运行。

## §1.2　C语言的发展历史

60年代，随着计算机科学的迅速发展，高级程序设计语言得到了广泛的应用。然而还没有一种可以用于书写操作系统等系统软件的高级语言。因为一般的高级语言不能实现对硬件

的直接操作。人们不得不用汇编语言(或机器语言)来书写,程序的可读性和可移植性都较差。为此,人们设想能否找到一种既具有一般高级语言特性,又具有低级语言特性的语言,能够集它们的优点于一身。于是在70年代初产生了一种能够用来研制各种系统程序的高级语言——C语言。

C语言的出现是与UNIX操作系统紧密联系在一起的,它起源于1968年发表的CPL语言。它的许多重要思想来源于Martin Richards在1969年研制的BCPL语言,以及以BCPL语言为基础,由Ken Thompson在1970年研制成的B语言。Ken Thompson用B语言写了第一个UNIX操作系统,用在PDP—7计算机上,D.M.Ritchie1972年在B语言的基础上研制出C语言,并用C语言写了第一个在PDP—11计算机上实现的UNIX操作系统。以后的C语言经过多次改进,但主要还是在贝尔实验室内部使用,直到1975年UNIX第六版公布后,C语言的突出优点引起人们的普遍重视。1977年出现了不依赖于具体机器的C语言编译文本,使C语言移植到其它机种时所要做的工作大大简化。同时,UNIX的广泛使用,也推动了C语言的普及。现在,适用于不同操作系统和不同机种的C语言编译程序有几十种之多,如C、Turbo C、Quick C等。不同版本略有差异,但C语言编译系统的基本部分是相同的,可以圆满地解决C语言程序在不同系统间的移植问题。本书叙述基本上以ANSI C(美国国家标准协会制订的新标准C)为基础,读者在具体使用时还应参考所用计算机系统的C编译的特点和规定。

C语言本身也存在着某些局限。例如,C语言对类型检查相对较弱,C语言本身几乎没有支持代码重用的语言结构,也不适合开发大型程序等。为解决这些问题,并保持C语言特点,1980年贝尔实验室的Bjarne Stroustrup博士等人对C语言进行了改进和扩充,称之为"带类的C",1983年取名为C++,以后又经过不断完善和发展,成为目前各种版本的C++。

## §1.3 C语言的特点

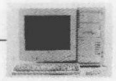
一种语言之所以能够存在和发展,并具有生命力,总是有其不同于(或优于)其它语言的特点。C语言的主要特点如下:

1. 语言简洁、紧凑,使用方便、灵活。C语言一共只有32个关键字(见附录),9种控制语句,程序书写形式自由,主要用小写字母表示,压缩了一切不必要的成分。下面将C与PASCAL语言作一些比较:

| C语言 | PASCAL语言 | 含 义 |
|---|---|---|
| { } | BEGIN……END | 复合语句 |
| if(e) S; | IF(e) THEN S | 条件语句 |
| int i; | VAR i:INTEGER | 定义i为整型变量 |
| int a[10] | VAR a:ARRAY[1…10]OF INTEGER | 定义a为整型一维数组 |
| int *p; | VAR P:↑ INTEGER | 定义p为指向整型变量的指针变量 |

学过PASCAL的读者自然会有所体会,没有学过PASCAL的读者从形式上也可以看到:C程序比PASCAL简练,源程序短。因此,输入程序时工作量小。

2. 运算符丰富。C的运算符包含的范围很广泛,共有34种运算符(见附录),从而使C的表达式类型多样化,灵活使用各种运算符可以实现在其它高级语言中难以实现的运算。



3. 数据类型丰富。C语言不仅提供了字符型、几种尺寸的整型和单、双精度的浮点型等基本数据类型和数组外，还允许程序员定义结构体、共用体等高级数据类型，用以描述比较复杂的数据对象。特别是C语言的指针类型，功能强大，应用也极为灵活，恰当使用可以简化程序结构，提高运行效率。

4. 控制结构简明清晰。C语言的控制结构(包括实现分支、循环和模块的语言成分和语句)完全符合结构化程序设计的要求，程序可读性好，便于调试和维护。

5. 高效率的目标代码。C语言具有丰富的运算符和直接控制计算机硬件的能力，在必要时可直接对字节、位甚至内存地址和寄存器操作，可以直接调用汇编语言编写的子程序，还可以在程序中直接嵌入汇编代码。加之种种C编译程序的设计都很注重考虑效率因素，因此，C语言程序编译后生成的目标代码长度短，运行速度快，在各种高级语言中它的效率最高。

6. 可移植性好。用C语言开发的程序可以很容易地从一种计算机、一种操作系统上移植到另一种计算机和另一种操作系统上，扩大了开发软件的应用价值。

上面我们只介绍了C语言的最容易理解的一般特点，至于C语言内部的其它特点将结合后面各章的内容作介绍。对于这些特点，读者也许还不能深刻理解，待学完C语言以后再回顾一下，就会有比较深的体会。也正是这些特点，使得C语言迅速流行起来，不仅大多数专业程序员将C语言列为首选编程语言，就是在非计算机专业的广大工程技术人员和计算机爱好者中间也出现了许多使用C语言的程序设计高手。C语言的应用不再局限于开发系统软件，而是扩大到包括CAD(计算机辅助设计)、数据库与信息处理、科学计算、实时控制、图形图像处理、网络通讯、人工智能等几乎所有的计算机应用领域。

当然，C语言也有一些缺点。例如，为了提高目标代码效率和编程的灵活性，有意取消了一些编译检查功能，特别是缺乏数据类型的一致性检测和不进行数组下标越界检查。这就要求程序员应当仔细检查程序，保证其正确，不要过分依赖C编译程序去查错，这可能会为初学者带来一定困难。然而瑕不掩瑜，从综合指标来看，C语言仍然可以说是目前最好的程序设计语言。只要掌握了C语言编程的基本规律，同时尽可能多地积累C语言的调试经验，就一定能够学好、用好C语言。

## §1.4　C程序的构成规则

下面我们通过两个例题说明C程序的构成规则。

**[例1.1]**

```
/*　程序 Hello.c:在屏幕上显示 Hello,C! */
main(　)
{
  printf("Hello,C! \ n");
}
```

运行结果：　Hello,C!

分析：本程序中的main表示“主函数”，每一个C程序都必须有一个main函数。函数体由{　}括起来。本例中主函数只有一个输出语句，在该语句中调用了库函数printf(　)，用于在屏幕上输出一个字符串。“\ n”是换行符，表示在屏幕上输出到这里时将回车换行。语句最

后有一分号。

程序的第一行是注释。C语言中的注释可以是由"/*"和"*/"括起来的任何文字,可以出现在程序中的任何地方,用来说明程序段的功能、变量的作用,以及程序员认为应该向程序阅读者说明的任何内容。在将C程序翻译成目标代码时所有的注释都会被忽略掉,不参加编译。因此,即使使用了很多注释也不会影响目标代码的长度和效率。恰当地应用注释可以使程序清晰易懂,便于调试,便于程序员之间的交流与协作。因此,编写程序时精心撰写注释是一个良好的编程习惯。

**[例1.2]**

```
/* 求两数中的大数 */
main(    )                    /* 主函数 */
{int a,b,c;                   /* 定义变量 */
scanf("%d,%d",&a,&b);         /* 输入变量 a 和 b 的值 */
c = max(a,b);                 /* 调用 max 函数,得到的函数值赋给 c */
printf("max = %d",c);         /* 输出 c 的值 */
}

int max(x,y)                  /* 定义 max 函数,函数值为整型,x,y 为形式参数 */
int x,y;                      /* 对形参 x,y 作类型定义 */
{ int z;                      /* 定义 max 函数中用到的变量 z */
if (x > y) z = x;             /* 用 if 判断 x,y 的大小,将大值赋值给 z */
else z = y;
return (z);                   /* 将 z 的值带回到调用处 */
}
```

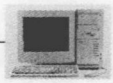
运行情况:8,5↙(用户输入8和5给a和b)

max = 8(系统输出C的值)

分析:本程序包含两个函数:主函数main和被调用的函数max。max函数的作用是将x和y中较大者的值赋给变量z,return语句将z的值返回给主调函数main,返回值是通过函数名max带回到main函数的调用处。main函数的第一行是变量定义部分,说明a和b是整型变量。第二行中的scanf是C提供的标准输入函数,其作用是将两个数值分别输入到a和b的地址所标志的单元中,即输入给变量a和b。&a和&b前面的"%d"是输入输出的格式字符串,用来指定输入输出时的数据类型和格式。第三行调用max函数,在调用时将实际参数a和b的值分别传送给max函数中的形式参数x和y。经过执行max函数,得到一个返回值(即max函数中变量z的值),把这个值赋给变量c,然后输出c的值。printf函数中双引号内的"max = %d",在输出时,其中"%d"将由c的值取代之,"max = "原样输出。

本例用到的函数调用、实参和形参等概念,在此只作简单解释。读者若不理解,可先不予深究,后面相应章节会作详细讲解。介绍此例,无非是使读者对C程序的组成和形式有一个初步的了解。

通过上面例子,可以了解到如下的C程序构成规则:



1.C程序的基本单位是函数。C语言的函数有三种：

第一种是主函数，名为main(　)，每个程序中只能有一个、也必须有一个主函数。在运行时，程序从主函数的第一条语句开始执行，不论main函数在整个程序中的位置如何(main函数可以放在程序最前头，也可以放在程序最后，可在一些函数之前或在另一些函数之后)。

第二种是用户自己编写的函数，如[例1.2]中的max(　)。程序中用户自定义函数的数目不限，当然也可以没有。实际上一个实用的C语言应用程序通常都包含若干用户自定义函数，而每个程序模块(函数)的功能单一、结构简单，便于编写和逐个调试。

第三种是C语言为用户提供的标准库函数，例如上面例子中的printf(　)和scanf(　)。C语言的开发者为了便于程序员编写各种类型的应用程序，已将许多常用的算法预先编写成函数，并将其编译后生成的目标文件统一存放在函数库中。这样，程序员编程时若需要实现这些功能，就不必再自己编程实现了，只需直接调用相应库函数即可。各种版本的C语言提供的库函数数目各不相同，一般都在数百种以上，用时需查相关资料。

另外，针对某个特殊领域，例如CAD工程绘图，也有由独立软件开发商提供的软件包，其中包含了若干面向具体应用领域的库函数。在安装好软件包之后，就可以像使用标准库函数那样使用软件包中的函数了。

2. 一个函数由两部分组成

(1)函数的说明部分。包括函数名、函数类型、函数属性、函数参数(形参)名、函数参数类型。

例如例1.2中的max函数的说明部分为：

```
int          max        (x,y)
 ↓           ↓           ↓
函数类型     函数名      函数参数
int          x,y;
 ↓           ↓
形参类型     形参
```

一个函数名后面必须跟一对圆括弧，函数参数可以没有，如main( )。

(2)函数体，即函数说明部分下面的大括弧{……}内的部分。如果一个函数内有多个大括弧，则最外层的一对{　}为函数整体的范围。

函数体一般包括：

①变量定义。如例1.2 main函数中的“int a,b,c;”。

②执行部分。由若干个语句组成。

当然在某些情况下，也可以没有变量定义部分，甚至可以既无变量定义，也无执行部分。如：

```
dump ( )
{ }
```

它是一个空函数，什么也不干，但这是合法的。

3.C程序书写格式自由，一行内可以写几个语句，一个语句可以分写在多行上。C程序没有行号，也不像FORTRAN或COBOL那样严格规定书写格式(语句必须从某一列开始书写)。

4. 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。例如：

```
c = a + b;
```

分号不可少。即使是程序中最后一个语句也应包含分号。

5.C语言本身没有输入输出语句。输入和输出的操作是由库函数scanf和printf等函数来

完成的。C对输入输出实行“函数化”,使输入输出操作更加灵活,既可调用系统提供的函数,也可以调用自己编写的输入输出函数。

6. 可以用/*……*/对C程序中的任何部分作注释。一个好的、有使用价值的源程序都应当加上必要的注释,以增加程序的可读性。同时在调试程序时,若出现运行错误,难以确定出错位置时,可把部分程序段用/*…*/括起来再调试,以便快速确定错误所在。

## §1.5 C语言程序的开发过程

绝大多数C语言程序为编译执行。因此,C语言程序的基本开发过程如图1.1所示。

### 1.5.1 编辑

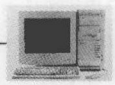
大部分C编译系统带有独立的编辑程序,可用于输入或修改源程序,也可使用操作系统提供的编辑程序,还可以使用功能更强的编辑软件。源程序经编辑程序由键盘输入后,形成源程序文件存在外存中。源程序文件的名字由用户指定,扩展名均为“.c”(系统默认为.c,用户可以不给扩展名)。

### 1.5.2 编译

C源程序一般由ASCII代码构成,计算机不能直接识别和执行,所以要由C编译程序将其翻译成二进制机器命令代码构成的目标程序。目标程序存放的文件名与源程序文件同名,但其扩展名一般为“.OBJ”或“.O”。

刚编写好的程序含有错误是很正常的,C编译程序对C源程序进行编译时如果发现有语法错误,就会输出相应的“出错信息”,提示用户C源程序的第几行出现了什么性质的错误。这时用户应该重新调用C编辑器,修改C源程序,再进行编译,往往要重复若干编译——修改——再编译的过程,直到编译无错通过为止。


图1.1 C语言程序的开发过程



### 1.5.3 连接

编译产生的目标文件虽然是由机器指令代码组成，但它还是不能直接在机器上运行，它是一个浮动的程序模块。也就是说，它是可重定位的，这意味着其中机器码指令的内存地址并未绝对地确定，只有偏移量是确定的。因此，程序需要重新定位在确定的绝对地址上，这由连接程序完成。所有的C语言编译程序都是和标准函数库文件一起提供的，保存在函数库文件的函数也都是可重定位的。当用户的C程序调用了一个标准库函数(或者别人编写的函数)时，编译程序"记忆"它的名字，随后连接程序(LINK)，将用户程序的目标代码同标准函数库文件中找到的目标代码结合起来，这个过程称为"连接"。连接时，对程序和函数进行重定位，内存偏移量被用来产生实际地址(绝对地址)，最终使程序变成在计算机上可以执行的形式——可执行文件，其文件名与目标文件名同名，扩展名为".exe"。如果连接不成功，则会指出有哪些函数没找到，或其它原因，在找到连接错误的原因并改正以后，一定要重新编译才能再次连接。否则，虽然源程序已经修改，但进行连接的目标程序还是以前有错误的目标程序，再次连接仍会产生同样错误。

### 1.5.4 运行

经过编译和连接后产生的可执行文件，只需在操作系统下打入可执行文件名，程序即可启动运行。

总之，C语言程序一般都要经过编辑、编译、连接之后才能运行。不同的C编译系统对于编译和连接的具体操作不尽相同，有的步骤明显，有的步骤不明显。具体的上机操作步骤请参看具体编译系统的有关资料。

另外，C语言允许分割编译，将大的程序分成若干块，装入若干文件，每一文件可单独编译，当所有文件编译完成后，再进行连接。连接程序将分别编译后产生的所有目标文件和库函数进行连接，形成一个完整的可执行文件。

## §1.6 TC集成环境下的上机步骤

C程序的开发需经过编辑、编译、连接、运行这样一个过程，在不同系统环境中，对这几个步骤都提供了相应的命令支持。本节只讨论C程序在Turbo C环境下的上机步骤。

Turbo C不仅是一个快速、高效的编译程序，同时还带有一个易学、易用的集成开发环境。使用Turbo C无需独立地编辑、编译和连接程序，就能建立并运行C程序。所有这些功能都组合在Turbo C的集成环境内，并且可以通过一个简单、清晰的主屏幕使用这些功能。

### 1.6.1 TC的调用及界面介绍

在DOS提示符下键入TC，并按回车键，就可完成TC程序的调用，进入TC集成环境的启动屏幕，包括主菜单和版本信息。按任意键，版本信息消失，留下如图1.2所示的主屏幕。主屏幕由四部分组成：主菜单、编辑窗、信息窗和快速参考行。

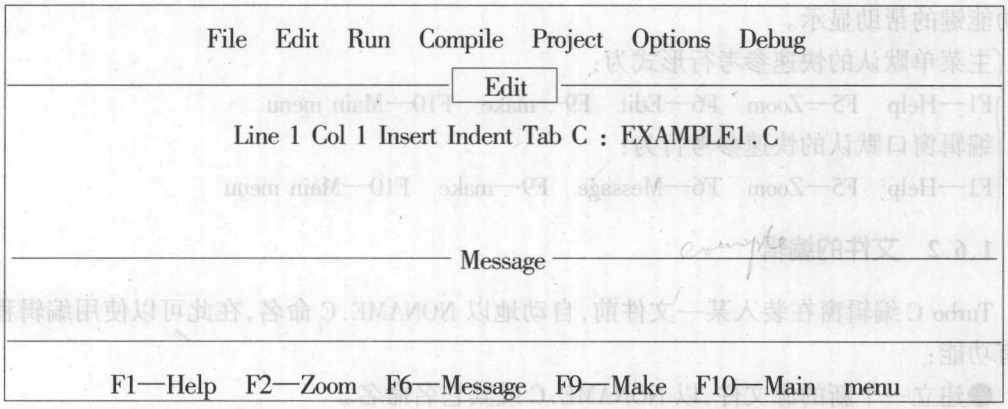



图 1.2 Turbo C 的主屏幕

1、主菜单

在主屏幕顶部是 Turbo C 的主菜单。包括：

File 包括对文件的装入、保存、选取、建立、写入等操作，以及修改目录、退出程序等。

Edit 建立并编辑源程序文件。

Run 自动编译、连接并运行程序。

Compile 将源程序编译并装配为目标文件。

Project 标识组成程序的文件，处理工程。

Options 保存或装入编译选择配置。

Debug 跟踪排错。

2、编辑窗口

编辑窗位于主菜单下，是用户的主要工作区域。只要将主菜单的选择亮条移到 Edit 并按 Enter，或按 Alt－E 都可进入编辑窗的操作。在编辑窗顶部有一编辑状态行给出了正在编辑文件的有关信息，如光标所在行、列号，插入模式设置等。

以下是一些常用编辑命令：

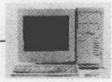
- 正文的滚动用↑ ↓ →←和 Pgup/Pgdn 键
- 删除一行用 CTRL—Y
- 设置块首标记 CTRL—KB
- 设置块尾标记 CTRL—KK
- 移动一块用 CTRL—KV
- 复制一块用 CTRL—KC
- 删去一块用 CTRL—KY
- 取消块标记用 CTRL—KH

3. 信息窗口

信息窗口位于编辑窗下面，在编译和调试源程度时，可以使用信息窗 Message 观察诊断信息。Turbo C 在信息窗列出被编译文件的每一个警告或出错信息，同时在编辑窗以高亮度条指出错误在源程序中的相应位置。

4. 快速参考行

无论工作在窗口或菜单上，屏幕的底部都有一个相应的快速参考行。该行是当前位置可



用功能键的帮助显示。

主菜单默认的快速参考行形式为：

F1—Help　F5—Zoom　F6—Edit　F9—make　F10—Main menu

编辑窗口默认的快速参考行为：

F1—Help　F5—Zoom　F6—Message　F9—make　F10—Main menu

### 1.6.2　文件的编辑

Turbo C 编辑窗在装入某一文件前，自动地以 NONAME.C 命名，在此可以使用编辑程序的所有功能：

●建立一个新的源文件，以 NONAME.C 或其它名命名。

●装入并编辑存在的文件。

●从编辑文件表中挑选一个文件，然后将它装入编辑窗。

●保存编辑窗中的文件。

●将编辑窗口中的文件写入一新文件。

●在编辑窗和信息窗之间转换，查找相应的编译错误，建立或编辑程序而不进行编译，并不需要信息窗。因此，可按 F5 将编辑窗扩展成全屏幕。

1. 建立一个新源文件

建立一个新源文件可用以下方法：

(1)在主菜单上选择 File/New，按 Enter，此时打开编辑窗，且文件名为 NONAME.C。

(2)在主菜单上选择 File/Load 后，打入源文件名。

2. 装入一个已存在的源文件

装入并编辑已存在的文件可用以下方法：

(1)在主菜单下选择 File/Load，打入编辑文件的路径和文件名，或直接输入文件名(可用通配符)选择当前目录下的文件。

(2)在主菜单下选择 File/Pick，可快速选取以前装入过的文件。

3. 保存源文件

(1)在系统的任何地方按下 F2 键。

(2)在主菜单下选择 File/Save。

在保存时，要注意给文件一个名字，否则系统将会以 NONAME 命名该文件，使其它用户进入 TC 时自动打开 NONAME 文件，引起文件的修改错误。

4. 写一个输出文件

可以将编辑窗中的文件写到新文件中，或覆盖一个已存在的文件，要达到这个目的，可在主菜单下选择 File/Write to 完成。

### 1.6.3　文件的编译和连接

在编译菜单下有编译到 OBJ 文件、生成 EXE 文件、连接 EXE、设置初始 C 文件等功能。

1. Compile to OBJ

该命令将源文件编译生成 OBJ 文件。执行该命令会显示要产生的文件名，所显示的文件名来自于下述两个文件名：

●装进编辑窗口的最后一个文件名。