

# 微机 C 语言 及其应用

李桂青 罗持久 编著



高等教育出版社

2

11

# 微机C语言及其应用

李桂青 编著  
罗持久

东南出版社

## 内 容 简 介

本书全面地、系统地阐述微型计算机上的C语言,深入地、具体地介绍微机C语言的特殊性以及相关的计算机硬件知识。内容新颖、深广,例题丰富多样,通俗易懂,便于自学。

全书共11章,内容包括标准C语言,扩展的C语言,诊断型专家系统的原理与设计方法。书中配有大量例题,例题内容涉及符号运算、数值运算、窗口、图形、绘图机绘图、软件加密、通讯程序设计等。

本书可供计算机科学、计算力学、人工智能、机械、土建等专业的大学生、研究生、工程技术人员、科学研究工作者和有关教师参考。

微 机 C 语 言 及 其 应 用

李 桂 青

罗 持 久 编 著

\* \* \*

气象出版社出版

(北京西郊白石桥路46号)

武汉工学院印刷厂印刷

气象出版社发行 全国各地新华书店经售

\* \* \*

开本:787×1092 1/16 印张:17.375 字数:444.8千字

1989年10月第一版 1989年10月第一次印刷

印数:1—10000 定价:7.50元

ISBN 7—5029—0278—3/TP·0014

# 前 言

本书主要是针对Turbo C语言来阐述微机C语言,但本书的应用并不限于Turbo C,对于Microsoft C也是基本适用的,仅系统函数略有差别。

本书是在作者的C语言讲义基础上写成的,包括作者近几年来在研究专家系统的过程中对Turbo C语言进行消化、改进、扩充得到的成果。

近30年来计算机语言有很大发展,至今已逾一千种,而为人们广泛接受的亦达数十种。在这众多的计算机语言中,作者深感C语言的确是一种非常精美的语言,尽管它还有一些不成熟的方面。C语言既是一种高级程序设计语言,又几乎可以代替汇编语言进行一切与计算机硬件有关的系统程序设计,它最成功的例子是UNIX操作系统。C语言也很适合作为开发专家系统、数据库系统的语言工具。C语言具有简洁的表达方式、极强的I/O功能、灵活的数据构造能力、丰富的数据类型、强有力的控制流程以及数值运算和符号处理能力、运行效率高、可移植性好等优点。C语言愈来愈受到广泛的重视,它将成为最流行的计算机语言。

Turbo C语言在各种C版本中属于较好的一种,它不仅具有较强的文本操作、串处理、窗口和图形功能,而且操作方便,可以交互式工作,为用户提供在线帮助,程序调试极为简单。

目前C语言的书籍主要限于UNIX系统,关于微机C语言的书籍,尚不多见,但微机C语言却有很多特殊性,本书正是基于这一目的而写的。

本书共11章。1~2章是C语言概述和Turbo C系统介绍,以便读者学完这两章之后即可上机操作;第3~6章介绍C语言的基本语法、基本程序设计方法;第7~9章介绍I/O操作系统函数和图形函数;第10章专门介绍C语言与其它高级语言的接口问题;第11章是C语言应用的总结,介绍了诊断型专家系统的基本原理和设计方法,包括知识库的建造和推理机设计等,并用C语言设计了一个专家系统的实例。每章均附有大量例题,例题包括数值运算、串处理、磁盘操作、软件加密、窗口、图形操作和通讯程序设计等内容。

本书的基本特点是内容新颖、深广、例题丰富多样。在写法上,力求概念清晰,通俗易懂,而且自成体系。阅读本书时一般不要求另找资料,特别便于自学。如读者在学习过程中能配合上机练习,特别是能把书中的例题调试一遍,则必将收到事半功倍的效果。如果读者需要本书介绍的Turbo C系统可来函与武汉工业大学抗震所罗持久同志联系。

本书可供计算机科学,计算力学,人工智能、机械、土建等专业的大学生、研究生、工程技术人员、科学研究工作者和教学工作者参考。

本书由耿南平同志负责誊写和绘图,谨致谢意。

限于作者水平,错误和不妥之处在所难免,恳请读者批评指正。

李桂青 罗持久  
武汉工业大学  
1989年10月

# 目 录

## 前言

|                              |        |
|------------------------------|--------|
| <b>第一章 C 语言概述</b> .....      | ( 1 )  |
| § 1—1 C 语言发展简介 .....         | ( 1 )  |
| § 1—2 C 语言的基本特点 .....        | ( 2 )  |
| § 1—3 C 语言简介 .....           | ( 3 )  |
| <b>第二章 C 语言编译系统</b> .....    | ( 10 ) |
| § 2—1 Turbo C 系统配置 .....     | ( 10 ) |
| § 2—2 系统的安装 .....            | ( 11 ) |
| § 2—3 C 语言的编译模式 .....        | ( 13 ) |
| § 2—4 C 程序的编译、连接和运行 .....    | ( 15 ) |
| § 2—5 Turbo C 综合开发环境 .....   | ( 20 ) |
| § 2—6 库管理程序 .....            | ( 28 ) |
| <b>第三章 基本数据类型和表达式</b> .....  | ( 30 ) |
| § 3—1 基本数据类型 .....           | ( 30 ) |
| § 3—2 类型的自动转换及强制转换 .....     | ( 36 ) |
| § 3—3 运算符和表达式 .....          | ( 37 ) |
| § 3—4 优先级和结合性 .....          | ( 44 ) |
| <b>第四章 语句</b> .....          | ( 46 ) |
| § 4—1 条件语句 .....             | ( 46 ) |
| § 4—2 循环语句 .....             | ( 50 ) |
| § 4—3 间断、接续和转向语句 .....       | ( 53 ) |
| § 4—4 开关语句 .....             | ( 54 ) |
| § 4—5 返回语句、退出语句及函数调用语句 ..... | ( 57 ) |
| <b>第五章 函数</b> .....          | ( 61 ) |
| § 5—1 函数及类型说明 .....          | ( 61 ) |
| § 5—2 变量说明与初始化 .....         | ( 65 ) |
| § 5—3 C 程序的构造 .....          | ( 68 ) |
| § 5—4 预处理和头文件 .....          | ( 71 ) |
| § 5—5 基本数值运算函数 .....         | ( 75 ) |
| <b>第六章 构造类型</b> .....        | ( 81 ) |
| § 6—1 数组和枚举类型 .....          | ( 81 ) |
| § 6—2 指针 .....               | ( 84 ) |

|             |                                |                |
|-------------|--------------------------------|----------------|
| § 6—3       | 结构.....                        | ( 92 )         |
| § 6—4       | 结构的应用.....                     | ( 100 )        |
| § 6—5       | 联合.....                        | ( 105 )        |
| § 6—6       | 自定义数据类型.....                   | ( 107 )        |
| <b>第七章</b>  | <b>输入/输出及字符串函数.....</b>        | <b>( 109 )</b> |
| § 7—1       | 标准输入/输出函数.....                 | ( 109 )        |
| § 7—2       | 流文件I/O函数.....                  | ( 112 )        |
| § 7—3       | 句柄文件I/O函数.....                 | ( 122 )        |
| § 7—4       | 文本及窗口处理函数.....                 | ( 130 )        |
| § 7—5       | 字符串处理函数.....                   | ( 136 )        |
| <b>第八章</b>  | <b>系统函数.....</b>               | <b>( 143 )</b> |
| § 8—1       | 内存管理函数.....                    | ( 143 )        |
| § 8—2       | 直接端口及中断处理函数.....               | ( 146 )        |
| § 8—3       | 磁盘管理系统函数.....                  | ( 159 )        |
| § 8—4       | 进程管理函数.....                    | ( 170 )        |
| § 8—5       | 时间处理函数.....                    | ( 174 )        |
| <b>第九章</b>  | <b>图形函数.....</b>               | <b>( 175 )</b> |
| § 9—1       | 图形工作方式的控制.....                 | ( 175 )        |
| § 9—2       | 基本图形函数.....                    | ( 183 )        |
| § 9—3       | 图形方式下的文本操作函数.....              | ( 190 )        |
| § 9—4       | 绘图机绘图.....                     | ( 195 )        |
| <b>第十章</b>  | <b>Turbo C与其它语言的接口.....</b>    | <b>( 204 )</b> |
| § 10—1      | 内部汇编.....                      | ( 204 )        |
| § 10—2      | C语言与汇编语言的接口.....               | ( 207 )        |
| § 10—3      | Turbo C与Turbo prolog语言的接口..... | ( 217 )        |
| § 10—4      | 协处理器的使用.....                   | ( 223 )        |
| <b>第十一章</b> | <b>C语言在专家系统中的应用.....</b>       | <b>( 225 )</b> |
| § 11—1      | 概述.....                        | ( 225 )        |
| § 11—2      | 专家系统基本原理.....                  | ( 226 )        |
| § 11—3      | 知识库的建造.....                    | ( 233 )        |
| § 11—4      | 推理机的设计.....                    | ( 239 )        |
| § 11—5      | 实例.....                        | ( 248 )        |
| <b>附录A:</b> | <b>键盘输入码.....</b>              | <b>( 256 )</b> |
| <b>附录B:</b> | <b>Turbo C函数索引表.....</b>       | <b>( 257 )</b> |
| <b>附录C:</b> | <b>BIOS及DOS中断表.....</b>        | <b>( 264 )</b> |
| <b>附录D:</b> | <b>CRT显示输出码.....</b>           | <b>( 272 )</b> |

# 第一章 C 语言概述

自1949年世界上第一台电子计算机(ENIAC)问世以来,作为人与计算机交流工具的程序设计语言,经过人们不断地开发,现在已有一千余种。不过,真正为人们所广泛接受和有实用价值的语言也只有数十种。其中有影响和有代表性的语言有以下几种类型:

ALGOL60和FORTRAN是两种广泛用于科学和工程计算的语言,常称为算法语言;COBOL是一种主要用于商业数据处理的语言,常称为数据处理语言;BASIC是一种主要用于人机交互的语言,常称为交互式语言;PASCAL侧重于描述编译系统等,常称为系统程序设计语言。

在所有的高级语言中,C语言是近年来最为人们重视的一种系统程序设计语言。本章简要介绍其发展历史、基本特点等。

## § 1—1 C 语言发展简介

C语言的产生与UNIX系统的发展有密切关系。UNIX系统是一个通用、复杂的计算机管理系统。UNIX第一版是在GE653机上产生的,通过纸带把可执行代码送到PDP—7上运行。在UNIX上实现了汇编语言后,UNIX系统又以汇编语言来编写。汇编语言的主要优点是能充分体现计算机硬件指令级的特性,而且形成的代码有较高的质量。但它的可读性、可移植性以及描述问题的性能远不如当时已经广泛采用的高级语言。能否创造一种既具有汇编语言的基本特性而又能克服其缺点的新语言来进行系统软件的设计呢?英国剑桥大学的M.Richards以及美国Bell实验室的D.M.Ritchie和K.Thompson在这方面进行了大量研究。M.Richards基于CPL语言(Combined Programming Language)于1969年发表了BCPL语言。K.Thompson于1970年在PDP—11/20上实现了B语言,并用B语言编写了UNIX操作系统和绝大部分实用程序。B语言是在BCPL语言的基础上发展起来的,它和BCPL语言一样,存在以下缺点:(1)不能适应PDP—11机用字节编址进行存取的要求;(2)缺乏具有一定表达能力的数据类型;(3)B语言的编译程序产生的是解释执行的代码,机器进行速度较慢。为此,D.M.Ritchie在B语言的基础上加进了构造类型等必要功能并作了归整,形成了一种新的语言——C语言,于1972年正式投入使用。1973年UNIX系统用C语言改写了一遍,加进了多道程序设计的功能,称为UNIX第五版,把UNIX系统推进到一个新阶段。以后的UNIX第六版、第七版,以及System III和System V都是在UNIX第五版的基础上发展起来的。

C语言的出现和崛起与PASCAL语言形成了一种对抗的形势。PASCAL语言作为结构化程序语言的代表,特别适用于教学和描述算法,但其适用性比C语言差。已经有不少PASCAL语言的爱好者转向用C语言作为软件开发工具了。追溯其发展历史,C语言与PASC-

AL语言有同一个祖先，都属于ALGOL语言族系（图1.1）。

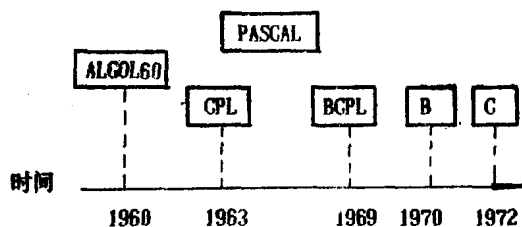


图 1.1

应当指出，C语言虽然在其发展的初期是附属于UNIX系统并在PDP—11上实现的，它的产生主要是为了更好地描述UNIX系统，但时至今日，C语言已独立于UNIX系统，独立于PDP—11系列而蓬勃发展。C语言是近十年来在计算机程序设计实践中作出了重大贡献的一种语言，已经成为微、小、超小、大和巨型机上共同使用的一种语言。K. Thompson和D. M. Ritchie也因其对C语言和UNIX系统方面的贡献获得了很高的荣誉。1982年，他们获得《美国电子学杂志》颁发的成就奖，成为该奖自颁发以来首次因软件工程成就而获奖的获奖者。1983年，他们又获得了计算机界的最高荣誉奖——图灵奖。

## § 1—2 C语言的基本特点

C语言的基本特点是简洁、灵活，表达能力强，产生的目标代码质量高，可读性强，可移植性好。说详细一点，则是：

1. C语言程序紧凑、简洁、规整。使用一些简单、规整的方法可以构成相当复杂的数据类型、语句和程序结构。

2. 表达方式简练、灵活、实用。C语言有多种运算符、多种表述问题的途径，并有多种求表达式值的方法，使程序设计者有很大的主动性，并能提高程序的可读性和编译效率以及目标代码质量。

3. 具有与汇编语言很相近的功能和描述方法，如地址运算、二进制数位运算以及使用寄存器存放变量，对硬件端口直接操作，充分使用计算机系统资源（如BIOS中断和DOS中断）等。

4. 具有丰富的数据类型。

在系统软件中，特别是操作系统中，对计算机的所有软、硬件资源要实施管理和调度，就必须有相应的数据结构作为操作基础。C语言具有四种基本数据类型：char（字符型）、int（整型）、float（浮点型）、double（双精度）和多种构造类型（数组和结构）以及复杂的导出类型。C语言还提供了与地址密切相关的指针以及有关运算符，指针可以指向各种类型的简单变量、函数、数组和结构等。C语言还允许用户自己定义数据结构。

5. 具有丰富的运算符。

丰富的数据类型与丰富的运算符相结合，使C语言具有表达灵活和高效率的优点。

6. C语言是一种结构化语言，适合于大型程序的模块化设计。

C语言具有编写结构化程序所需要的基本控制流结构语句；C语言程序以函数集合构成，函数具有独立的功能，并作为模块化设计的基本单位；它所包含的源文件，可以分割为多个源文件，分别对各个源文件进行编译，然后连接起来形成可执行的目标程序文件；C语言还提供了多种存储属性，使得数据可以按需要放在相应的模块中。

7. C语言为字符、字符串、集合和表的处理提供了良好的基础，它能够表示和识别各种可直接显示的以及起控制作用的字符，能够区分单个字符和字符串等。

8. 具有预处理程序和预处理语句，从而提高了程序的能力、可移植性，给大型程序调试提供了方便。

9. C语言的标准程序库具有使用方便、功能极强的多种函数，从而使C语言的优点更加突出。例如，输入/输出（I/O）就是依靠调用I/O函数来实施的。

10. 移植性好。用C语言编写的程序之所以具有良好的移植性，一方面是由于其语言适用范围广，另一方面是处理系统很紧凑。

在C语言里，仅仅定义了最基本的控制语句，而输入、输出和字符处理等基本语句没有定义，它们作为外部函数，链接到所编写的程序上来使用的。这也是C语言处理系统小、移植性好的原因之一。

## § 1—3 C语言简介

上节已对C语言的基本特点作了一些简介，本节将对C语言开发程序的基本过程以及C语言程序作些初步介绍，使读者尽快对C语言建立概念。

### 一、开发C语言程序的过程简介

开发C语言程序的基本过程可以划分为以下步骤：

#### 1. 编辑

用C语言编写的程序称为C语言源程序，简称C源程序或源程序，或C程序。开发C语言程序的第一步是利用系统提供的编辑程序（例如UNIX系统的文本行编辑程序ed，全屏幕编辑程序vi或用DOS的行编辑程序EDLIN）把源程序以文件形式存入文件系统。文件可以任意取名，但为方便记忆和使用，文件名最好能反映程序的功能特点。所有C源程序文件名最后两个字符必须为.C（即文件扩展名），例如，example.c。这一步叫做编辑。

在MSDOS下，标准文件名由三个部分组成：

路径+文件名+扩展名

“路径”指明文件所在驱动器和子目录名，目录用“\”+字符串组成，文件名是不大于8个的字母数字组成的字符串，扩展名是不大于3个的字母数字组成，文件名与扩展名之间用“.”分隔，它们都必须以字母开头，如A:\C\example.c是A驱动器子目录\C下的文件，文件名为example，扩展名为.c。

#### 2. 编译和汇编

将源程序以文件形式存放在文件系统后，即可调用C语言编译程序对它进行编译。在微机中根据所用C编译版本有：Turbo C语言的编译程序TCC，Lattice C的LC，Microsoft C的MC等。命令TCC example.c，即要求系统对文件example.c中的C源程序进行编译。

编译过程的下一步是通过TCC命令将C语言翻译成由机器指令形成的程序，这种程序称为目标程序，并存放在目标文件中。目标文件名与源程序文件名被系统自动的假定为同名，仅文件的扩展名不同；目标程序文件的扩展名为OBJ。例如，对C源程序example.c进行编译和汇编后形成的目标码文件名为example.obj。

系统在编译过程中，如发现源程序在词法、语法、语义等方面有错误，它将向用户报告这些错误并根据错误的级别和数量适时结束编译过程。然后用户再使用编辑程序修改源程序中的错误，并重新启动编译过程。如此反复直至完全排除源程序中的错误。

### 3. 连接

编译过程的第三步为连接，其任务是通过TLINK命令形成在计算机上可安装并可执行的目标程序。连接的对象除刚获得的目标程序外，还可以包括以前用其它语言所写的编译好的目标程序或目标码形成的系统库程序。连接程序构成的可执行目标程序存放在另一个文件中，该文件名可在调用TCC或TLINK命令时指定，详见第二章，如在使用TCC命令时没有指定，则将它取为缺省名\*.exe，例如：

TLINK c0s example,example.exe,<CR>

其中“<CR>”符号表示回车键，可执行程序名为example.exe。如果使用TCC编译example，则编译后自动连接 example，形成example.exe。

### 4. 执行

以上几步完成后即可执行该程序。

执行时在DOS提示符下，直接敲入文件名（不含扩展名）即可，DOS会自动地寻找带有.exe扩展名的该文件，例如：

example <CR>

系统将把 example.exe 文件加载到内存，并分配相应的代码区、数据区和堆栈区。

如程序在执行时未达预期目标，或者说所取得的结果不正确，则应检查程序的逻辑，包括使用的算法、数据结构以及程序的组织等。这种检查过程叫做排错。在进行排错处理时，通常需要修改源程序，因而又需重复上述步骤直至达到预期目标。整个开发过程可用图 1.2 表示。

上面只是对开发C语言的基本过程作了一些简介，第二章将还要详细讨论这个问题。

## 二、C语言的基本词汇

C语言的基本词汇包括数字（0，1，…，9）、英文字母（大写A，B，…，Z，小写a，b，…，z）、下划线\_以及若干特殊字符。

C语言以这些基本词汇构成基本语法单位，包括标识符、关键字、分隔符、运算符和常量。

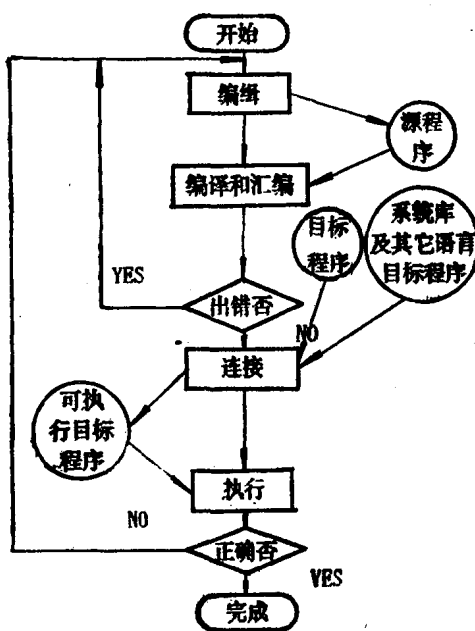


图 1.2

### 1. 标识符

标识符是给变量、函数等取名的符号,是以英文字母或下划线打头的英文字母、数字和下划线组成的序列。例如:

example, \_sum 1, sum\_1

都可作为标识符,不过用户最好不要使用下划线开头,因为所有标识符在编译时,编译程序会自动加上\_,以下序列不是标识符:

lexample (不是以英文字母打头)

ex-A (出现上划线,而不是下划线)

priac\$ (出现有英文字母、数字及下划线以外的符号)

在C语言里,大写与小写英文字母是有区别的,例如:Test, test, TESt, TEST, TEST是不同的标识符。在C语言中常用小写表示标识符,用大写表示常量。

标识符的有效位数在某些系统中长达32个字符,但在大多数系统里有效位数为7个或8个,有效标识符的长度在微机上一般都可达32个,参见表1.1。

表 1.1 标 识 符 长 度

| 计 算 机 型 号               | 标识符有效位数     |
|-------------------------|-------------|
| 3 B系列                   | 8 个字符,大小写不同 |
| PDP—11系列机               | 7 个字符,大小写不同 |
| VAX —11系列机              | 7 个字符,大小写不同 |
| 以M68000为CPU的机器          | >8个字符,大小写不同 |
| 以Z8000为CPU的机器           | >8个字符,大小写不同 |
| 以Intel 8086/8088为CPU的机器 | >8个字符,大小写不同 |
| Honeywell 6000          | 6 个字符,大小写相同 |
| IBM370                  | 7 个字符,大小写相同 |

对于只有8个有效标识符的系统,例如3B系列机,则标识符的第9个及其以后的字符是不起作用的,如下列标识符

example 1 example12 example123在系统看来是完全相同的,因为前8个字符相同。不过在使用IBM PC系列及兼容机时,则可使用32以内的标识符长度。

### 2. 关键词

在标识符中有一些留作特殊用途,在C语言中称为关键词。应当指出,关键词是与系统有关的,应参考机器手册。表1.2所列关键词在IBM PC系列机上都可使用。除注明为TC的关键词只可在Turbo C中使用外,所有关键词都可在任何C编译中使用。

### 3. 分隔符

C语言中的分隔符,主要指空白符、制表符、新行符(换行符)及注释。新行符以\n表示,注释符以/\*打头,并以\*/结尾。

运算符也是一种分隔符。

制表符、运算符和常量等,在有关章节中介绍。

表1.2 关 键 字

| 数据类型说明      | 存储类型说明       | 语 句 说 明  | Turbo C伪变量 | 其 它     |
|-------------|--------------|----------|------------|---------|
| char        | auto         | if       | TC _CS     | sizeof  |
| int         | extern       | else     | TC _DS     | typedef |
| long        | register     | switch   | TC _ES     |         |
| short       | static       | case     | TC _SS     |         |
| TC signed   | TC far       | default  | TC _AH     |         |
| unsigned    | TC near      | while    | TC _AL     |         |
| AN void     | TC huge      | do       | TC _AX     |         |
| float       | TC cdecl     | for      | TC _BH     |         |
| double      | TC interrupt | return   | TC _DH     |         |
|             |              |          | TC _DX     |         |
| struct      | TC pascal    | continue | TC _DL     |         |
| union       | TC asm       | break    | TC _BL     |         |
| AN enum     |              | goto     | TC _DI     |         |
| AN const    |              |          | TC _SI     |         |
| AN volatile |              |          | TC _SP     |         |
|             |              |          | TC _BX     |         |
|             |              |          | TC _CH     |         |
|             |              |          | TC _CL     |         |
|             |              |          | TC _CX     |         |

注：①AN→表示该关键词是ANSI标准（美国国家标准协会）。

②TC→Turbo C中才使用的关键词，Turbo C中的大部分关键词，在MS-C,Lattice-C中都可直接应用。

③伪变量可以直接作为标识符使用，不必重新定义，它们分别对应8086的寄存器，见§2—3。

### 三、C程序简介

#### 1. C程序的格式

C程序一般由一个或多个函数组成，这些函数可集中放在一个文件或几个文件中。为表示这些文件都是C语言源程序，各文件名必须以.C结尾，如example.c就是一个C程序文件。下面用一个简单例子说明C程序的基本格式。

##### 例1.1

A>type sum1.c

```
main(    )    /*This program is the sum of a and b*/
{
int a,b,sum;
```

0

```

a = 43; b = 56;
sum = a + b;
printf( "sum is %d\n", sum );
}

```

这是一个完整的C程序。下面对它作些说明：

(1) C程序无行号。

(2) A>type命令把在A驱动器上的sum1.c文件列出来。

(3) sum1.c是文件名，后面两个字符.c表示C源文件。

(4) main是函数名，函数名一定要紧跟圆括号( )，括号内可以包含若干个参数，但也可不带参数。例如，本例main( )就是不带任何参数的函数。

在C程序中可以有若干个函数，但一定要有一个函数main( )，它可称为主函数。不管它放在什么地方，C程序总是从它开始执行。

(5) /\*与\*/之间的内容是注释，不被编译，它对计算机代码不起任何作用，只对用户起说明、备忘和帮助阅读的作用，是可有可无的，但一般还是以加上注释为好。

(6) 花括号{ }相当于ALGOL语言中的begin(开始)和end(结束)，它是函数体的界限符。

(7) int a,b,sum; 是一个说明语句，它说明变量a,b及sum为整型变量。int是integer(整数)的缩写。

(8) 程序中的=是赋值运算符，表示把赋值运算符右边的数值或运算结果赋给其左边的变量。

(9) printf( )是有格式标准输出函数，其主要的功能是在终端屏幕上按格式说明符所指定的格式输出。

%d是printf( )函数的最简单的输出格式，常称转换控制，它把其后的参数(本例为sum)按十进制整型格式输出，前后不留空格。

(10) \n是新行符，新行符后的任何字符将显示在终端的下一行上。

(11) 语句后面要加;，表示该语句结束。带有;号的语句叫做简单语句，它是执行单位。在C程序中，printf( )语句和赋值语句都是简单语句。若干简单语句用{ }括起来就构成一个复合语句，花括号前一个语句末的分号不可少，而在}后是否有分号要考虑语句类型，这将在后面介绍，通常说明语句要加“;”，而可执行语句不加“;”。

(12) 这个程序执行的结果是

```

A>sum 1 <CR>
sum is 99

```

## 2. 最小C程序

最小C程序是：

```

min( ) { }

```

min是函数名，这里之所以没有用main，是使它更具有一般性。因{ }内为空，故该函数一旦被调用，什么也不执行就立即返回到调用它的函数。在程序开发时，设置这种形式的函数是为将来要设置的函数先安排一个位置。待所需设置的函数设计好后，命一个新名取而代之。

### 3. 函数的定义

C程序是由函数组成的。函数的定义形式为：

函数存储类说明+函数数据类型说明+函数名(参数表)+参数类型说明

```
{  
    变量说明  
    语句部分  
}
```

函数定义中必不可少的部分是：

函数名( ) { }

这就是前面所讲的最小C程序。

上述函数定义形式是由两部分组成的，即由函数说明和函数体组成的。

#### (1) 函数说明

函数存储类说明指出函数是内部函数(或称静态的)或外部函数，分别用关键词static和extern表示。内部函数只能被函数所在文件内的函数调用，而外部函数则可以由其它文件内的函数调用。

函数数据类型说明定义函数返回值的数据类型，C程序所使用的基本数据类型有int(整型)、char(字符型)、float(浮点型)及double(双精度)。无类型说明的函数编译程序自动地定义为整型。

函数名是函数的名字，用标识符表示。参数表在函数名后的圆括号( )内。这里的参数是形式参数，简称形参(亦称哑元)。当函数被调用时，形参要被实在参数(简称实参或实元)所替换。

参数类型说明指定参数的数据类型。

#### (2) 函数体

函数体从花括号“{”开始，直到与之对应的“}”为止，是函数的主体部分。

变量说明一般接在“{”后面。变量也是用标识符表示的，其基本类型也是int, float, double, 和char。

接在变量说明后面的是语句部分。这一部分是函数的主体，描述函数的功能，也是C程序的核心内容，是我们今后讨论的重点。

## 四、微机C语言的特点

微机C语言与UNIX系统下的C语言有着许多相同之处和差别，首先作为一种C语言它们都遵循K&R关于C语言的基本定义，许多微机C版本都是标准C的子集，例如CC86。其次作为微机C和UNIX C，由于它们所依附的操作系统的重要差别则有许多功能完全不同的函数，特别在使用系统资源方面几乎没有共同之处。然而C语言是特别适合于系统开发的工具，因此使用系统资源是C语言的重要内容。如果仅仅为了满足数值运算的要求，也许使用FORTRAN就可以了。

作为本书将要重点引用的Turbo C语言是一种功能强大的微机C语言，具有以下特点：

1. 使用的操作系统是MS-DOS(或PC-DOS)和CP/M，有个别功能函数的调用需使用3.0以上版本的操作系统。

2. Turbo C支持K&R关于C的全部定义,并且还扩展了许多功能。
3. Turbo C除了标准C的功能外,在使用系统资源方面设计了许多函数可供用户直接使用,例如键盘交互,视频输出,磁盘管理,打印机和串行口输入输出函数。
4. Turbo C适用于以Intel \* 86系列芯片(如8086, 80286)为CPU的计算机。这意味着它能在目前广为使用的主流机种IBM PC, 长城系列及兼容机上可靠运行。
5. Turbo C具有功能齐全的图形功能,文本窗口处理功能,中文信息处理功能较弱。
6. 经过作者的研究和二次开发, Turbo C具有直接绘图机驱动功能,完善的汉字信息处理和图形方式下的多窗口处理功能。

Turbo C是微机C版本中功能较好的一种,使用亦特别方便。应用水平可深可浅。Turbo C的综合环境对于初学者特别适宜。

计算机语言是一门实践性很强的学科。虽然C语言素以难学著称,但若能安静地坐在计算机前边学习边实践,用不了多少时间必然会有很大收获。

为了便于读者学习,从第三章开始介绍C语言时,先介绍标准语法,然后补充Turbo C的扩展内容。所谓标准语法是指满足K&R的定义以及所使用的语法规则\*,这样便于读者使用除Turbo C以外的其它C编译版本,使用标准语法编写的C程序移植性好。但是如果读者有Turbo C版本就可以使用扩充内容。应该指出,许多扩充内容都可在Microsoft C(简称MS-C)中使用,这在学习中应该注意的。

\* 即指Brian W. Kernighan & Dennis M. Ritchie所著《C程序设计语言》一书。

## 第二章 C语言编译系统

本章介绍的C语言编译系统,指的是微机C语言编译器。为了配合后面章节的学习,我们选择Turbo C编译系统作为工具,详细介绍该系统的安装、使用、编译和连接操作。读者可以通过上机实习来逐步熟悉Turbo C,以便今后边学习边上机。语言是一门实践性很强的学科,如果读者能把本书的例题上机调试,就一定能收到事半功倍的效果。

### § 2—1 Turbo C系统配置

Turbo C系统共有五张软盘,这些软盘包含了所有的程序文件以及进行C程序编辑、编译、连接和运行C程序所需要的库。此外还有一些辅助工具,如库操作程序和便于用户学习编程技术的程序实例。

#### 一、Turbo C 硬件环境

Turbo C以及Microsoft C和Lattice C均可在IBM PC系列及兼容机和国产长城、浪潮系列微机上运行,包括XT和AT型机。要求MS DOS 2.0或更高版本的支持,内存至少384KB RAM,单色或彩色监视器,至少一个软盘驱动器。但是如果有硬盘的话,使用起来将更为方便。

如果机器配有浮点运算器8087或80287,用户可通过命令来决定是否使用。系统配有浮点运算库。如果机器没有安装8087/80287,则可启动系统的仿真浮点运算库,运算精度不受影响,但运算速度将降低。因此若用户程序包含大量浮点运算时,无浮点运算器的微机将大大增加运算时间。

#### 二、Turbo C系统软件

Turbo C系统软件主要分成综合开发环境、命令行环境、实用程序和源程序实例四个部分。下面是各盘上的主要文件:

1\*盘: 综合开发环境

TC.EXE

Turbo C综合环境编译器

INSTALL.BAT

} Turbo C硬盘安装程序。如果要在硬盘上安装Turbo C  
} 可用此程序。

INSTALLH.BAT

TCINST.EXE

TC.EXE环境设置程序

CPASDEMO.PAS

Pascal与Turbo C程序接口演示的Pascal程序

CPASDEMO.C

Pascal与Turbo C程序接口演示的C程序

|                    |                                     |
|--------------------|-------------------------------------|
| CTOPAS.TC          | 为使用TC.EXE建立C程序并同Pascal程序正确连接的环境配置文件 |
| PBAR.PRO           | Prolog与C程序接口演示的Prolog程序             |
| BAR.C              | Prolog与C程序接口演示的C程序                  |
| 2*盘: 命令行编译系统与实用程序  |                                     |
| TCC.EXE            | Turbo C命令行编译器                       |
| TLINK.EXE          | Turbo C连接程序                         |
| BGIOBJ.EXE         | 转换设备驱动程序,BGI为.OBJ文件的实用程序            |
| CPP.EXE            | Turbo C预处理程序                        |
| MAKE.EXE           | 通用目标程序(.EXE)生成器                     |
| TCCONFIG.EXE       | 在TCCONFIG.TC和TURBOC.CFG之间转换的程序      |
| TLIB.EXE           | Turbo C库管理程序                        |
| C0.ASM             | C主程序main入口的汇编源程序                    |
| 3*盘、4*盘: 头文件及库     |                                     |
| *.H                | Turbo C头文件(扩展名为.H)                  |
| C0?.OBJ            | 对应于各编译模式的启动函数                       |
| C?.LIB             | 对应于各编译模式的C语言库                       |
| MATH?.LIB          | 对应于各编译模式的数学库                        |
| EMU.LIB            | 8087仿真浮点运算库                         |
| FP87.LIB           | 8087浮点运算库                           |
| GRAPHICS.LIB       | 图形函数库(不分编译模式)                       |
| PLOTTER.LIB        | 绘图机驱动程序库                            |
| 5*盘: 设备驱动程序, 字库及实例 |                                     |
| *.C                | 实例的C源程序(扩展名为.C)                     |
| CPINIT.OBJ         | 当连接C与Prolog时的初始化模块                  |
| ATT.BGI            | ATT400图卡驱动程序                        |
| CGA.BGI            | CGA驱动程序                             |
| EGAVGA.BGI         | EGA/VGA驱动程序                         |
| HERC.BGI           | Hercules图形卡驱动程序                     |
| IBM8514.BGI        | IBM8514图形卡驱动程序                      |
| PC3270.BGI         | PC3270图形卡驱动程序                       |
| GOTH.CHR           | 粗体字库(仅限于ASCII字符集)                   |
| LITT.CHR           | 小体字库(仅限ASCII字符集)                    |
| SANS.CHR           | Sans字库(仅限于ASCII字符集)                 |
| TRIP.CHR           | 三重字体字库(仅限于ASCII字符)                  |

## § 2—2 系统的安装

Turbo C系统的安装根据硬件的配置, 有不同的安装方法。