



教育部
高等职业教育示范专业规划教材

C语言程序设计 实用教程

主编 陈方



教育部高等职业教育示范专业规划教材

C 语言程序设计实用教程

主 编 陈 方

副主编 吉顺如

参 编 邓 蓓 李 萍

主 审 程龙泉



机械工业出版社

本书是为高职高专电类、机类专业编写的 C 语言程序设计实用教材。

全书按照高职高专教学规律,循序渐进、由浅入深地介绍 C 语言的特点和运行环境、各种数据类型及其运算、控制语句、数组、函数、指针、结构体、文件、位运算及综合应用等。每章都附有小结、习题和上机实训题,便于教师施教和学生学习。

本书可作为高职高专电类、机类专业的教材,也可供其他有兴趣的读者学习和参考。

★为方便教师授课,本书配有参考电子教案,有需要的教师可与责任编辑联系(010-88379758),免费索取。

图书在版编目(CIP)数据

C 语言程序设计实用教程/陈方主编. —北京:机械工业出版社, 2005.6
教育部高等职业教育示范专业规划教材
ISBN 7-111-16742-2

I. C... II. 陈... 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(2005)第 063687 号

机械工业出版社(北京市百万庄大街 22 号 邮政编码 100037)

责任编辑:于 宁

责任印制:杨 曦

济南新华印刷厂印刷·新华书店北京发行所发行

2005 年 7 月第 1 版第 1 次印刷

787mm×1 092mm 1/16·10.75 印张·262 千字

定价:17.00 元

凡购本书,如有缺页、倒页、脱页,由本社发行部调换

本社购书热线电话(010) 68326294

封面无防伪标均为盗版

前 言

C 语言是世界上应用最广泛的几种计算机程序语言之一。目前广泛使用的各种 C 语言编译系统有 Turbo C (简称 TC)、Microsoft C (简称 MSC)、Borland C (简称 BC) 等, 它们的基本部分都是相同的, 本书介绍 Turbo CV2.0。

C 语言是当前软件开发中的主程序语言之一, 它具有适应性强、应用范围广 (基本可以取代汇编语言来编写各种系统软件和应用软件)、语言简洁、使用灵活、表达能力强、程序运行效率高、可移植性好、便于学习和应用等特点。C 语言是一种结构化程序设计语言, 程序逻辑结构可以用顺序、选择和循环三种基本结构组成。C 语言的函数结构便于对程序进行自顶向下逐步求精的分解, 从而实现模块化的结构设计, 便于程序模块化, 符合现代程序设计风格。另外, C 编译系统所占的存储空间很少, 只需 4M 左右。用 C 语言编写各种控制程序可以有效地减少冗余, 节省存储空间。如目前比较流行的嵌入式系统 (用于控制、监视或者辅助操作机器和设备的装置) 软件开发中, 大量采用“汇编语言+C 语言”的方式, 其中 C 语言占 80%~90% 左右。大量嵌入式技术也已经应用于工业控制、数控机床、智能工具、工业机器人、服务机器人等各个行业, 正在逐渐改变着传统的工业生产和服务方式。

本书是按照全国高职高专机电类专业教学计划及教材编写工作会议审定的“C 语言程序设计实用教程”编写大纲编写的, 吸收了各校高职高专电类、机类专业“C 语言”课程教学改革的经验, 将基础性与实用性有机地结合, 减少了课时, 强化了综合应用, 体现了高职特色。

本书由陈方副教授担任主编, 吉顺如副教授担任副主编, 邓蓓副教授、李萍讲师参编, 程龙泉副教授担任主审。其中邓蓓老师编写第 1、2 章; 李萍老师编写第 3、4、8、9 章; 吉顺如老师编写第 5、6、7 章; 陈方老师编写第 10、11 章及 3 个附录, 并负责全书的修改与定稿。

本书可作为高职高专电类、机类等专业的教材, 也可供其他有兴趣的读者学习和参考。

在本书出版之际, 主编代表全体作者, 感谢参加全国高职高专机电类专业教学计划及教材编写工作会议的各兄弟院校专家对教材编写大纲所提出的宝贵建议, 感谢机械工业出版社的有关编辑在本书编写过程中所提出的宝贵意见, 以及为本书的出版所做的一切工作。

由于作者水平有限, 书中难免有不当之处, 恳请专家和广大读者批评指正。

编 者

目 录

前言

第1章 C语言简介	1	本章小结	65
1.1 发展史和特点	1	习题4	66
1.2 程序结构及范例	3	上机题	69
1.3 程序设计	5	第5章 函数	71
1.4 编程环境	7	5.1 函数概念	71
本章小结	10	5.2 函数参数和函数的值	72
习题1	11	5.3 函数的调用	74
上机题	11	5.4 函数的嵌套调用和递归调用	76
第2章 数据类型及其运算	12	5.5 数组作为函数参数	78
2.1 数据类型	12	5.6 局部变量和全局变量	79
2.2 常量与变量	12	5.7 动态存储变量与静态存储变量	82
2.3 用 typedef 定义类型	18	5.8 内部函数和外部函数	86
2.4 赋值语句	18	本章小结	87
2.5 运算符和表达式	19	习题5	87
2.6 数据输出	25	上机题	90
2.7 数据输入	29	第6章 指针	91
本章小结	32	6.1 指针与指针变量	91
习题2	33	6.2 指针运算	93
上机题	33	6.3 指针与数组	96
第3章 控制语句	36	6.4 指针与字符串	101
3.1 分支流程控制语句	36	6.5 指针与函数	103
3.2 循环流程控制语句	40	6.6 指向指针的指针	104
本章小结	47	本章小结	105
习题3	48	习题6	105
上机题	51	上机题	107
第4章 数组	53	第7章 结构体、共用体和枚举	108
4.1 一维数组	53	7.1 结构体	108
4.2 二维数组	56	7.2 共用体	117
4.3 字符数组	59	7.3 枚举	121

目 录

7.4 用 typedef 定义类型名	123	第 10 章 位运算	139
本章小结	123	10.1 位逻辑运算	139
习题 7	124	10.2 位移位运算	143
上机题	127	10.3 位复合赋值运算	144
第 8 章 文件	128	本章小结	144
8.1 文件的打开与关闭	128	习题 10	145
8.2 文件的读与写	129	上机题	145
8.3 文件的定位	132	第 11 章 应用程序举例	146
8.4 检错与处理	133	11.1 全屏幕模拟时钟的 C 源程序	146
本章小结	133	11.2 设计立体按钮的 C 源程序	150
习题 8	134	11.3 一种文件加密技术的 C 源程序	152
上机题	134	本章小结	155
第 9 章 编译预处理	135	上机题	155
9.1 包含指令	135	附录	157
9.2 宏定义指令	136	附录 A Turbo C 2.0 菜单	157
9.3 条件编译指令	137	附录 B 标准 ASCII 码字符编码表	161
本章小结	138	附录 C Turbo C 常用库函数	162
习题 9	138	参考文献	165
上机题	138		

第 1 章 C 语言简介

1.1 发展史和特点

1.1.1 C 语言的发展

C 语言是国际上广泛流行的计算机高级程序设计语言。在 C 语言出现之前，系统软件主要是用汇编语言编写的，程序的可读性和可移植性都比较差。

1967 年，英国剑桥大学的 Martin Richards (马丁·理查德) 在 CPL (Combined Programming Language) 语言的基础上，实现并推出了 BCPL (Basic Combined Programming Language) 语言。BCPL 语言是计算机软件人员在开发系统软件时，作为记述语言使用的一种结构化程序设计语言。它能够直接处理与机器本身数据类型相近的数据，具有与内存地址对应的指针处理方式。

1970 年美国贝尔实验室的 K.Thompson (肯·苏姆普逊) 以 BCPL 语言为基础，又作了进一步简化，设计出了很简单的而且很接近硬件的 B 语言 (取 BCPL 的第一个字母)，并用 B 语言写了第一个 UNIX 操作系统，在 PDP-7 上实现。但 B 语言过于简单，功能有限。此后，在美国的贝尔研究所进行更新的小型机 PDP-11 的 UNIX 操作系统的开发工作中，Dennis M.Ritchie (戴尼斯·M·利奇) 和 Brian W.Kernighan (布朗·W·卡尼汉) 对 B 语言做了进一步的充实和完善，于 1972 年推出了一种新型的程序语言——C 语言 (取 BCPL 的第二个字母)。C 语言既保持了 BCPL 和 B 语言的优点 (精练、接近硬件)，又克服了它们的缺点 (过于简单、数据无类型等)。1973 年，K.Thompson 和 Dennis.M.Ritchie 两人合作把 UNIX 的 90% 以上用 C 语言改写 (即 UNIX 第 5 版，原来的 UNIX 操作系统是 1969 年由美国的贝尔实验室的 K. Thompson 和 Dennis.M.Ritchie 开发成功的，是用汇编语言写的)。

后来，C 语言多次作了改进，但主要还是在贝尔实验室内部使用。直到 1975 年 UNIX 第 6 版公布后，C 语言的突出优点才引起人们普遍注意。1977 年出现了不依赖于具体机器的 C 语言编译文本《可移植 C 语言编译程序》，使 C 语言移植到其他机器时所需做的工作大大简化了，这也推动了 UNIX 操作系统迅速地在各种机器上实现。1978 年以后，C 语言已先后移植到大、中、小型机上，已独立于 UNIX 和 PDP 了。

以 1978 年发表的 UNIX 第 7 版中的 C 编译程序为基础，Brian W. Kernighan 和 Dennis M. Ritchie (合称 K&R) 合著了影响深远的名著《The C Programming Language》，这本书中介绍的 C 语言成为后来广泛使用的 C 语言版本的基础，它被称为标准 C。1983 年，美国国家标准化协会 (ANSI) 根据 C 语言问世以来各种版本对 C 的发展和扩充，制定了新的标准，称为 ANSI C。ANSI C 比原来的标准 C 有了很大的发展。K&R 在 1988 年修改了他们的经典著作《The C Programming Language》，按照 ANSI C 标准重新写了该书。1987 年，ANSI 又公布了新标准——87 ANSI C。目前流行的 C 编译系统都是以它为基础的。

C 语言已风靡全世界，成为世界上应用最广泛的几种计算机语言之一。目前在微型机上广泛使用的各种 C 语言编译系统有 Turbo C (简称 TC)、Microsoft C (简称 MSC)、Borland C

(简称 BC) 等, 它们的基本部分都是相同的, 但也有一些差异, 因此读者应了解所用的计算机系统的 C 语言编译系统的特点和规定。本书介绍 Turbo C V2.0。

1.1.2 C 语言的特点

C 语言成为当前软件开发中的主程序语言, 是由它的特点所决定的。C 语言的主要特点如下:

(1) 适应性强。它能适应从 8 位微型机到巨型机的所有机种。

(2) 应用范围广。它可用于系统软件以及各个领域的应用软件。

(3) 语言本身简洁, 使用灵活, 便于学习和应用。C 语言一共只有 32 个关键字、9 种控制语句, 用于构成 C 语言的全部指令。程序书写形式自由, 区分大小写字母, C 语言的关键字用小写字母表示。在源程序表示方法上, 与其他语言相比, 一般功能上等价的语句, C 语言的书写形式更为直观、精练。

(4) 语言的表达能力强。C 语言是面向结构化的程序设计语言, 通用、直观; 运算符达 30 多种, 涉及范围广、功能强。可直观处理字符, 访问内存物理地址, 进行位操作; 可以直接对计算机硬件进行操作。它反映了计算机的自身性能, 基本可以取代汇编语言来编写各种系统软件和应用软件。

(5) 数据结构系统化。C 语言具有现代化语言的各种数据结构, C 语言的数据类型有: 整型、实型、字符型、数组类型、指针类型、结构体类型、共用体类型等, 便于实现各种复杂的数据结构的运算, 且具有数据类型的构造能力。因此, 能用来实现各种复杂的数据结构(如链表、树、栈等)的运算。

(6) 控制流程结构化。C 语言是一种结构化程序设计语言, 即程序的逻辑结构可由顺序、选择和循环三种基本结构组成。C 语言提供了功能很强的各种控制语句(如 if while for switch 等语句), 并以函数作为主要结构成分, C 语言的函数结构便于对软件进行自顶向下逐步求精的分解, 从而实现模块化的结构设计。便于程序模块化, 符合现代程序设计风格。用 C 语言开发的软件不但设计容易、编码容易, 而且调试和维护都容易, 可以节省人力及系统资源。

(7) 运行质量高, 程序运行效率高。C 语言所生成的目标代码仅比汇编语言生成的目标代码效率低 10%~20%, 但语言编程速度快、程序可读性好, 易于调试、修改, 这也是 C 语言被广泛应用的原因之一。

(8) 可移植性好。统计资料表明, C 语言编译程序 80% 以上的代码是公共的, 因此稍加修改就能移植到各种不同型号的计算机上。

(9) C 语言存在的不足之处是: 运算符和运算优先级过多, 不便记忆; 语法定义不严格, 编程自由度大, 给不熟悉的程序员带来一定困难; C 语言的理论研究及标准化工作也有待推进和完善。

另外, C 语言编译系统所占的存储空间很少, 只需 4M 左右。用 C 语言编写各种控制程序可以有效地减少冗余, 节省存储空间。如目前比较流行的嵌入式系统(用于控制、监视或者辅助操作机器和设备的装置)软件开发中, 大量采用“汇编语言+C 语言”的方式, 其中 C 语言占 80%~90% 左右。各种使用嵌入式技术的电子产品有: MP3、PDA、手机、智能玩具, 网络家电、智能家电、车载电子设备等。在工业和服务领域中, 大量嵌入式技术也已经应用于工业控制、数控机床、智能工具、工业机器人、服务机器人等各个行业, 正在逐渐改变着

传统的工业生产和服务方式。

1.2 程序结构及范例

下面看几个简单的 C 语言程序，然后从中分析 C 程序的特性。

[例 1-1] 仅由 main()函数构成的 C 语言程序。

```
main()  
{  
    printf("This is a C program.");  
}
```

程序运行结果:

This is a C program.

其中，main()表示“主函数”。每一个 C 程序都必须有一个主函数。函数体由一对大括弧{ }括起来。本例中主函数内只有一个输出语句，printf()是 C 语言中的屏幕输出函数，是 C 语言输入/输出库函数之一。双引号" "内的字符串按原样输出。语句最后有一分号，表示这个语句结束。初学 C 语言者常见错误之一，就是漏写这个分号，或在不该写分号的位置写分号。

[例 1-2] 从计算机键盘上输入两个整数，计算它们的和，并在显示器上输出。

程序如下:

```
#include <stdio.h> /* 文件包含命令 */  
main() /* 主函数 */  
{  
    int a, b, c; /* 声明部分，定义变量 */  
    scanf("%d, %d", &a, &b); /* 输入变量 a 和 b 的值 */  
    c = a+b; /* 将 a+b 的和值赋值给变量 c */  
    printf("sum=%d", c); /* 输出 c 的值 */  
}
```

程序运行结果:

17, 39

sum=56

其中，#include <stdio.h> 是文件包含命令，其作用是将存放在 include 子目录下的已有文件 stdio.h 插入到该命令所在位置，取代该命令，从而把文件 stdio.h 与当前的源程序合并连成一个源文件。注意：文件包含命令后面不要分号。/*.....*/表示注释部分，为便于理解，可用汉字表示注释，也可以用英语或汉语拼音作注释；采用注释可提高程序的可读性，对编译和运行不起作用；注释可加在程序中任何位置。第 4 行是声明部分，定义整型变量 a、b 和 c。第 5 行是从键盘上输入 a 和 b 的值（操作过程是：先输入 a，接着输入逗号，再输入 b，回车），scanf()是键盘输入函数，是 C 语言输入/输出库函数之一，其中“%d”表示输入输出“格式字符串”，d 表示“以十进制整数形式输入”。第 6 行是将 a 和 b 的值求和并赋值给变量 c。第 7 行是先在屏幕上输出 sum=，随后再输出 c 的值。

【例 1-3】 由 main() 函数和 1 个自定义函数 max() 构成的 C 语言程序。

程序如下:

```
int max(int x, int y)    /*自定义 max 函数, 函数值为整型, 形参 x、y 为整型*/
{ return( x>y ? x: y ); }    /*返回 x 与 y 相比较后的大数*/
main()                    /*主函数*/
{ int num1, num2;          /*声明部分, 定义变量*/
  printf("Input the first integer number: "); /*输出双引号内的字符串*/
  scanf("%d", &num1);      /*从键盘输入 num1 的值*/
  printf("Input the second integer number: "); /*输出双引号内的字符串*/
  scanf("%d", &num2);      /*从键盘输入 num2 的值*/
  printf("max=%d\n", max(num1, num2)); /*将 num1、num2 的值传递给函数 max 并
                                  输出函数的返回值*/
}
```

程序运行结果:

```
Input the first integer number:6
Input the second integer number:9
max=9
```

其中, 自定义函数 max 的作用是接受传递来的 num1、num2 这两个数后, 找出其中的较大者, 并用 return 语句将大值返回。

通过以上几个例子, 可以看到:

(1) 一个 C 程序都是由若干个函数构成的, 函数是 C 程序的基本单位。在一个 C 程序中, 必须有且只有一个函数名为 main 的主函数, 可以有库函数和自定义函数。自定义函数是用户根据需要自己编制的函数 (如例 1-3 中的 max 函数)。C 语言的函数库十分丰富, ANSI C 建议的标准库函数中包括 100 多个函数, Turbo C 和 MSC4.0 提供 300 多个库函数。本书附录 C 中列举了一些常用的库函数, 要从事 C 语言的研究和开发工作, 应准备一本库函数手册。

(2) 一个函数由两部分组成:

1) 函数的首部, 即函数的第一行。包括函数名、函数类型、函数参数 (形参) 名、参数类型。

例如: 例 1-3 中的 max 函数的首部为

int	max	(int	x,	int	y)
↓	↓	↓	↓	↓	↓
函数类型	函数名	函数参数类型	函数参数名	函数参数类型	函数参数名

一个函数名后必须跟一对圆括弧, 函数参数可以没有, 如 main()。

2) 函数体, 即函数首部下面的大括弧 { } 内的部分。如果一个函数内有多个大括弧, 则最外层的一对 { } 为函数体的范围。

函数体一般包括:

① 说明部分: 对程序中用到的变量或函数等给出声明, 它只在程序的编译阶段起作用, 而在程序的运行中不起作用, 如 “int x, y;”。

② 可执行语句：将在编译时产生可以执行的指令代码。

(3) 一个C程序总是从 `main` 函数开始执行的，而不论 `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，也可以放在程序的最后；或在一些函数之前、在另一些函数之后）。

(4) 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。

(5) 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。分号必不可少，即使是程序中最后一个语句也应包含分号。

(6) 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` 等函数来完成的。

(7) 可以用 `/*.....*/` 对C程序中的任何部分作注释，以增加程序的可读性。这部分内容不受C语言的语法制约，不参加源程序的编译。注释可以使用自然语言书写，其编写要简明扼要，清晰易懂。注释文本也可以放在文件的开始，用来说明文件名称、开发日期、版本、作者以及功能介绍等。注释文本也可以放在程序某一段的前面，用来分隔程序段落并对该段程序代码的功能进行说明。对一行源程序代码的注释则放在该行的后边，用来解释说明本行的运算。`/*` 和 `*/` 必须成对使用，且 `/*` 和 `*/` 以及 `/*` 和 `/*` 之间不能有空格，否则都出错。

1.3 程序设计

1.3.1 C语言的语句

与其他高级语言一样，C语言也是利用函数体中的可执行语句，向计算机系统发出操作命令。按照语句功能或构成的不同，可将C语言的语句分为5类。

1. 控制语句 控制语句完成一定的控制功能。C语言只有9条控制语句，又可细分为3种：

(1) 选择结构控制语句

`if()...else...`, `switch()...`

(2) 循环结构控制语句

`do...while()`, `for()...`, `while()...`, `break`, `continue`

(3) 其他控制语句

`goto`, `return`

2. 函数调用语句 函数调用语句由一次函数调用加一个分号（语句结束标志）构成。

例如：

```
func()  
{ printf("This is a C function statement.");  
}  
main()  
{ func(); /*调用函数 func()*/  
}
```

3. 表达式语句 表达式语句由表达式后加一个分号构成。最典型的表达式语句是在赋

值表达式后加一个分号构成的赋值语句。

例如：“num=5”是一个赋值表达式，而“num=5;”却是一个赋值语句。

表达式能构成语句是 C 语言的一个特色。其实“函数调用语句”也是属于表达式语句，因为函数调用也属于表达式的一种。只是为了便于理解和使用，才把“函数调用语句”和“表达式语句”分开说明。由于 C 程序中大多数语句是表达式语句（包括函数调用语句），所以把 C 语言称作“表达式语言”。

4. 空语句 空语句仅由一个分号构成。显然，空语句什么操作也不执行。

5. 复合语句 复合语句是由大括号括起来的一组（也可以是 1 条）语句构成。例如：

```
main()
{   .....      {.....} /*复合语句。注意：右括号后不需要分号。*/
    .....      }
```

复合语句的性质：

- (1) 在语法上和单一语句相同，即单一语句可以出现的地方，也可以使用复合语句。
- (2) 复合语句可以嵌套，即复合语句中也可出现复合语句。
- (3) 复合语句中最后一个语句中的最后那个分号不能忽略不写。

1.3.2 程序基本结构

近年来广泛采用结构化程序设计方法，使程序结构清晰、易读性强，以提高程序设计的质量和效率。从程序流程的角度来看，程序可以分为三种基本结构，即顺序结构、选择结构、循环结构（见图 1-1、图 1-2、图 1-3）。这三种基本结构可以组成所有的各种复杂程序。C 语言提供了多种语句来实现这些程序结构。本章节介绍这些基本结构，使读者对 C 程序有一个初步的认识，为后面各章的学习打下基础。

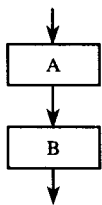


图1-1 顺序结构

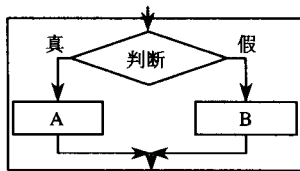
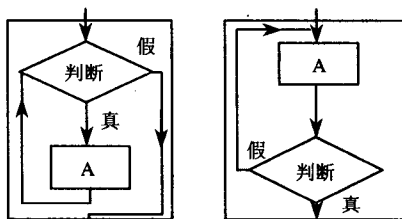


图1-2 选择结构



a) 当型循环结构

b) 直到型循环结构

图1-3 循环结构

1. 顺序结构 就是先执行 A 操作，后执行 B 操作，两者是顺序执行的关系。
2. 选择结构 当判断条件成立（或为“真”）时执行 A 操作，否则执行 B 操作。

注意：只能执行 A 操作或 B 操作其中之一。

3. 循环结构 有两种循环结构：

- (1) 当型循环结构（见图 1-3a），即当判断条件成立（“真”）时反复执行 A 操作，直到判断条件不成立（“假”）时才停止循环。

(2) 直到型循环结构 (见图 1-3b), 先执行 A 操作, 再判断条件是否为“假”, 若为假, 再执行 A 操作, 如此反复, 直到判断为“真”为止。

由选择结构可以派生出另一种基本结构, 即多分支选择结构, 如图 1-4 所示。

根据 k 的值 ($k_1, k_2, \dots, k_i, k_n$) 不同, 而决定执行 $A_1, A_2, \dots, A_i, A_n$ 之一的操作。这就是多分支选择结构。

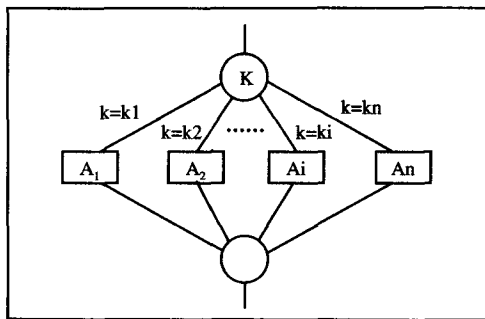


图1-4 多分支选择结构

1.3.3 关键字

C 语言的关键字共有 32 个, 根据关键字的作用, 可分其为数据类型关键字、控制语句关键字、存储类型关键字和其他关键字四类。

(1) 数据类型关键字 (12 个): `char, double, enum, float, int, long, short, signed, struct, union, unsigned, void`。

(2) 控制语句关键字 (12 个): `break, case, continue, default, do, else, for, goto, if, return, switch, while`。

(3) 存储类型关键字 (4 个): `auto, extern, register, static`。

(4) 其他关键字 (4 个): `const, sizeof, typedef, volatile`。

1.3.4 基本字符集

一个 C 程序是 C 语言基本字符构成的一个序列。C 语言的基本字符集包括:

(1) 数字字符: 0、1、2、3、4、5、6、7、8、9。

(2) 字母: A、B、C、……、Z、a、b、c、……、z (字母的大小写是可区分的)。

(3) 运算符: `+, -, *, /, %, =, <, >, <=, >=, !=, ==, <<, >>, &, |, &&, ||`……。

(4) 特殊符号和不可显示字符: `_` (连字符或下划线)、空格、换行、制表符。

1.3.5 标识符

在程序中有许多需要命名的对象, C 语言规定了在程序里描述名字的规则, 这些名字包括: 变量名、常数名、数组名、函数名、文件名、类型名等, 通常统称为“标识符”。

(1) 命名规则: 标识符由字母、数字或下划线“`_`”组成, 它的第一个字符必须是字母或下划线。例如: `c_1`、`_str` 是合法的, 而 `8c3` 是非法的。命名时通常采用英语或汉语拼音, 见名知意; 命名时字母大小写是有区别的, 如 `a`、`A` 是不同的标识符; 标识符不能和关键字相同, 也不要和库函数名相同。

(2) 长度规则: 在 Turbo C V2.0 中, 标识符的有效长度为 1 至 32 个字符。在不同的系统中, 长度规定不同, 无论哪个系统, 至少前 8 个字符有效。

1.4 编程环境

1.4.1 运行一个 C 语言程序的一般过程

Turbo C 是一个集源程序编辑、编译、连接、运行与调试于一体、用菜单驱动的集成软件环境。编辑并运行一个 C 语言程序的一般步骤如下:

(1) 启动 TC, 进入 TC 集成环境。

(2) 编辑 (或修改) 源程序。

(3) 编译。如果编译成功, 则可进行下一步操作; 否则, 返回 (2) 修改源程序, 再重新编译, 直至编译成功。

(4) 连接。如果连接成功, 则可进行下一步操作; 否则, 根据系统的错误提示, 进行相应修改, 再重新连接, 直至连接成功。

(5) 运行。通过观察程序运行结果, 验证程序的正确性。如果出现逻辑错误, 则必须返回 (2) 修改源程序, 再重新编译、连接和运行, 直至程序正确。

(6) 保存源程序。

(7) 退出 TC 集成环境, 结束本次程序运行。

其中, 第 (3) 至第 (5) 步也可以合并进行。

1.4.2 TC 的启动、退出与命令菜单

1. 启动 Turbo C 如果采用 DOS 方式启动, 则先进入 Turbo C 子目录, 输入 TC, 回车; 如果采用 Windows 方式启动, 则先进入 Turbo C 子目录, 双击 TC.EXE 文件。

进入 Turbo C V2.0 集成开发环境中后, 其顶上一行为 Turbo C V2.0 主菜单, 中间窗口为编辑区, 接下来是信息窗口, 最底下一行为参考行。这四个窗口构成了 Turbo C V2.0 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。Turbo C V2.0 主菜单及子菜单的详细介绍见附录 A。

2. 窗口简介 启动 Turbo C 后, 其主菜单条横向排列在屏幕顶端, 并被激活, 其中 File 主项成为当前项。主菜单的下面, 是 Edit (编辑) 窗口和 Message (消息) 窗口。两个窗口中, 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, 其中:

① Line 1 Col 1: 显示光标所在的行号和列号, 即光标位置。

② Insert: 表示编辑状态处于“插入”。当处于“改写”状态时, 此处为空白。

③ d: 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, 表示处于“控制键”状态)。

3. 菜单的使用

(1) 按下功能键 F10, 激活主菜单。如果主菜单已经被激活, 则直接转下一步。

(2) 用左、右方向键移动光带, 定位于需要的主项上, 然后再按回车键, 打开其子菜单 (纵向排列)。

(3) 用上、下方向键移动光带, 定位于需要的子项上, 回车即可。执行完选定的功能后, 系统自动关闭菜单。

注意: 菜单激活后, 又不使用, 可再按 F10 / Esc 键关闭, 返回原来状态。

4. 设置路径 在一台计算机上安装 Turbo C V2.0 后, 因不同的用户安装的路径不同 (如有的安装为 C: \tc, 有的安装为 D: \turboc2 等), 因此, 第一次启动 Turbo C V2.0 后, 首先应正确设置路径并保存, 再编辑和调试源程序。否则如果路径不对, 调试 (编译、连接、运行) 时会提示错误。

以 Turbo C V 2.0 安装在 D: \turboc2 为例。在 turboc2 目录中, 已有两个 Turbo C V 2.0 自带文件夹 Include 和 Lib 及若干其他文件 (包括 tc.exe); 另外用户新建一个文件夹 User, 用于保存自编的 C 程序。设置路径的步骤如下:

(1) 进入 Options (选择菜单) | Directories (路径) | Include directories (包含文件的路径), 输入 D: \turboc2\Include, 回车确定;

(2) 进入 Options (选择菜单) | Directories (路径) | Library directories (库文件路径), 输入 D: \turboc2\Lib, 回车确定;

(3) 进入 Options (选择菜单) | Directories (路径) | Output directory (输出项目文件 (.obj 文件) 和可执行文件 (.exe) 文件的路径), 输入 D: \turboc2\User, 回车确定;

(4) 进入 Options (选择菜单) | Directories (路径) | Turbo C directory (输出源文件 (.c 文件) 的路径), 输入 D: \turboc2\User, 回车确定;

(5) 进入 Options (选择菜单) | Save options (存储配置), 保存所有新的设置, 对弹出的窗口, 依次按回车、Y 即可。其中提示的文件 TCCONFIG.TC 为配置文件。

完成上述步骤后, 就可以编辑和调试源程序了, 以后再启动 Turbo C2.0, 也不用再行设置。

5. 退出 Turbo C 退出 TC 有两种方法:

(1) 菜单法: File | Quit (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)。

1.4.3 编辑并保存一个 C 语言源程序

1. 激活主菜单 选择并执行 File | Load 项 (快捷键: F3)。

2. 输入文件名 在 “Load File Name” 窗口, 输入源程序文件名。

文件名的输入有两种方法: 直接输入和选择输入。

(1) 直接输入。按照文件名的组成字符串, 逐个字符输入即可。如果是已经存在的文件, 系统就在编辑窗口显示该文件的内容, 可供编辑、修改。如果是新文件, 则给出一个空白编辑窗口, 可供输入新的源程序。如果该文件不在当前目录下, 则需要冠以路径名和 (或) 盘符。

(2) 选择文件 (仅适用于已经存在的源程序文件)

① 空回车, 打开当前目录下、后缀为 .c 的所有文件的文件名窗口。

② 用上、下、左、右方向键, 将光带定位于所需的文件名上。

③ 按回车键。

3. 常用编辑操作 在编辑源程序过程中, 随时都可以按 F2 键 (或 File | Save), 将当前编辑的文件存盘, 然后继续编辑, 所编辑的源程序运行成功后, 要保存文件。这是一个良好的习惯!

关于在线帮助: 在任何窗口 (或状态) 下, 按 F1 键激活活动窗口 (或状态) 的在线帮助。

● 下一页 —— PageDown, 返回上一页 —— PageUp。

● 关闭在线帮助、返回原窗口 (或状态) —— Esc。

● 返回前一个在线帮助屏 —— Alt+F1 (无论在线帮助是否被激活)。

● 返回在线帮助索引 —— F1: 激活在线帮助后, 再按 F1, 则返回在线帮助索引, 以便查询其他类别在线帮助信息。

● 查询库函数的在线帮助信息 —— ^F1: 将光标移到需要查询函数名的首字符上, 然后键入 ^F1, 即可获得该库函数的在线帮助信息。

注：为简化描述，用“^”代表“Ctrl”键。^Fn 就是 Ctrl+Fn，下同。

1.4.4 编译、连接、运行单个源程序文件

对编辑好的源程序，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、连接、运行（即 1.4.1 中的第（3）至第（5）步骤合并进行），并生成可执行文件。

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，若按空格键，则屏幕下端的“**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在源代码中的相应位置。

注意：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！

1.4.5 查看结果

程序运行结束后，仍返回到编辑窗口。此时选择并执行 **Run | User Screen** 项（快捷键：Alt+F5），可查看运行结果，按任一键就返回编辑窗口。如果发现逻辑错误，则可在返回编辑窗口后，进行修改；然后再重新编译、连接、运行，直至正确为止。

1.4.6 编辑下一个新的源程序

选择并执行 **File | New** 项即可。

如果屏幕提示如下确认信息：

NONAME.c not saved. Save? (Y/N)

如果需要保存当前正在编辑的源程序，则键入“Y”，输入文件的保存路径和文件名；否则，键入“N”（不保存），系统给出一个空白的编辑窗口，可以开始编辑下一个新的源程序。

1.4.7 利用记事本编写与保存源程序

Turbo C V2.0 是英文环境，源程序中的汉字（如注释或字符串中汉字）显示出来是乱码，可以在 UC DOS、汉化版的 TC 或中文版的 VC 环境中显示和编辑汉字。教学中，为便于学生理解，对源程序的注释多采用汉字。可利用 Windows 附件中的记事本编写、保存、打开、修改 C 源程序。为便于查找用户自编的 C 程序文件，建议在 Turbo C V2.0 的子目录下新建一个文件夹 **user**（或 **work**），保存自编的 C 程序文件。在用记事本保存文件时，保存类型选择“所有文件”，文件名用汉字、字母、数字及其他符号都可以，因 Turbo C V2.0 是英文环境，建议不用汉字，文件扩展名取 C（如 **file.c**），保存到 Turbo C V2.0 的子目录下的 **user**（或 **work**）文件夹中。

本章小结

1. C 语言是一种功能强大的计算机高级语言，它既适合作为系统描述语言，也适合作为通用的程序设计语言。任何计算机语言都有一系列的语言规定和语法规则，C 语言的基本规则是：有自己的基本字符集、标识符、关键字、语句和标准库函数等；C 程序的基本单位是函数，一个完整的 C 程序有且仅有一个主函数 **main**，可以有若干个子函数，也可以没有子函

数。这些子函数有用户自定义的函数，也有C编译系统提供的标准库函数。每个函数都由函数说明和函数体两部分组成，函数体必须用一对花括号括起来。

2. C语言中包含了五种基本语句：控制语句、函数调用语句、表达式语句、空语句、复合语句，它们完成各自特定的操作。C程序每个语句都由分号作为结束标志。

3. C语言采用结构化程序设计方法，使程序结构清晰、易读性强。从程序流程的角度来看，程序可以分为三种基本结构，即顺序结构、分支结构、循环结构。C语言提供了多种语句来实现这些程序结构。

4. 一个C源程序需要经过编辑、编译和链接后才可运行，C源程序(.c)编译后生成目标文件(.obj)，对目标文件和库文件连接后生成可执行文件(.exe)。程序的执行是对可执行文件而言的。

习 题 1

1. C语言有哪些主要特点？
2. C语言的基本单位是什么？
3. C语言程序的3种基本结构是什么？
4. C语言标识符的命名规则是什么？
5. C语言有几种语句？单个语句以什么符号结束？复合语句如何表示？

上 机 题

一、目的和要求

1. 熟悉C语言的开发环境和C程序的上机步骤。
2. 了解Turbo C V2.0编译系统，熟悉各菜单的功能，掌握在该系统上编辑、编译、连接和运行一个C程序的方法。
3. 经过调试、运行简单的C程序，初步了解C语言源程序的特点。

二、练习题

1. 根据正在使用的计算机的情况，在Turbo C目录下设置一个user子目录，用来保存自己编写的C源程序；在Turbo C V2.0集成环境中，在Options（选择菜单）下的Directories（路径）中，正确设置：Include directories、Library directories、Output directory和Turbo C directory的目录，并通过Save options（存储配置），保存所有的选择。
2. 写一个C程序，输出一句话“This is a C program.”。
3. 编写一个C程序，求一个圆的面积。
4. 编写一个C程序，输入a、b、c三个值，输出其中最大者。
5. 编写一个C程序，从键盘上输入两个字符，计算其ASCII码值的相减值和相加值。