

四川省高校计算机等级考试中心指导教材

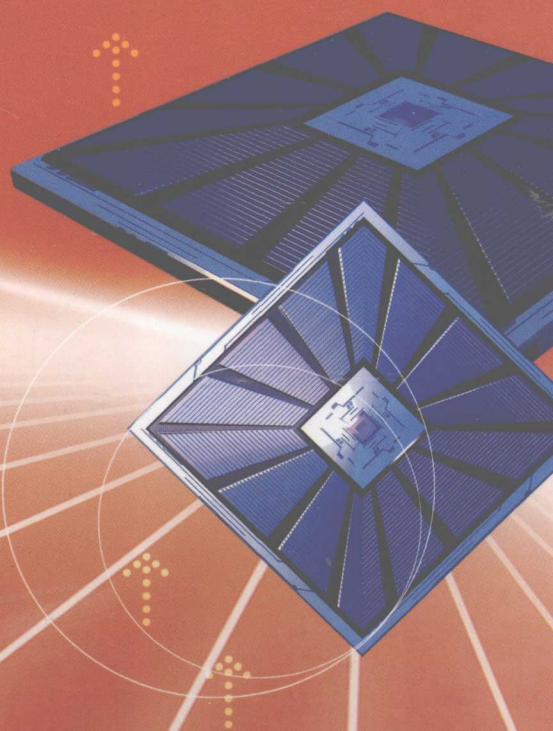
C Chengxu Sheji Jiaocheng

C 程序 设计教程

主 编 孙淑霞 代世雄

主 审 黄迪明

四川出版集团
四川科学技术出版社



○ 本教材是在四川省高校计算机等级考试中心的直接指导下编写 ○

C程序设计教程

C CHENGXU SHEJI JIAOCHENG

主 编：孙淑霞 代世雄

主 审：黄迪明

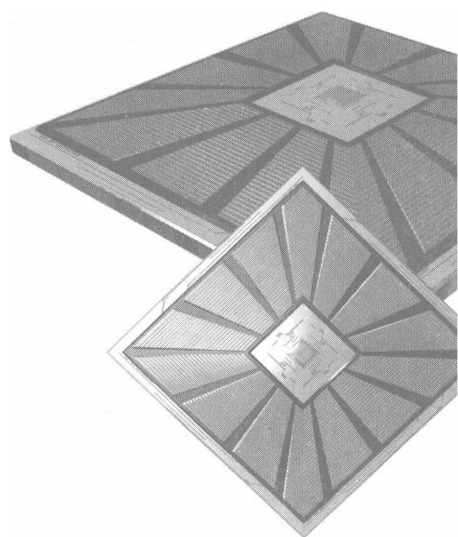
副主编：戴彦群 于 春

编 委：（以姓氏笔画为序）

王邦千 王 超 张 彤 刘 耀 李朝林

李思明 杨 毅 杜 华 查 勇 柳 军

龚 皓 韩 德 温爱红 蒲海波



四川出版集团
四川科学技术出版社

图书在版编目 (C I P) 数据

C 程序设计教程/孙淑霞, 代世雄主编.

—成都: 四川科学技术出版社, 2009. 1

ISBN 978-7-5364-6638-8

I. V… II. ①孙… ②代… III. C 语言—程序设计 IV. TP312

中国版本图书馆 CIP 数据核字 (2008) 第 176686 号

内 容 简 介

本教材根据四川省计算机等级考试大纲编写而成。全书共由 9 章组成, 其主要内容包括: 程序设计概述、C 程序设计基础、程序的控制结构、数组、指针、函数、结构体与共用体、文件访问、变量的作用域与生成期、编译预处理与变量的存储类型等。每章后面都附有适量的练习题, 练习题参考答案和解析在与之配套的《C 实验与考级题训精解》教材中, 这将为学课后自学提供更好的帮助。

本书可作为普通高校、高职高专非计算机专业学生学习用书, 同时可供参加四川省和全国计算机等级考试人员使用及自学者参考。

C 程序设计教程

主 编 孙淑霞 代世雄

主 审 黄迪明

责任编辑 戴 林

特约编辑 李跃丽

封面设计 吴 强

责任出版 邓一羽

出版发行 四川出版集团·四川科学技术出版社
成都市三洞桥路 12 号 邮政编码 610031

成品尺寸 185 mm × 260 mm

印 张 17.5 字 数 399 千

印 刷 郫县犀浦印刷厂

版 次 2009 年 1 月成都第一版

印 次 2009 年 1 月成都第一次印刷

定 价 32.60 元

ISBN 978-7-5364-6638-8

■ 版权所有·翻印必究 ■

■ 本书如有缺页、破损、装订错误, 请寄回印刷厂调换。

地址/成都市三洞桥路 12 号 电话 / (028) 87734035 13018235860

邮编/610031

高等学校计算机基础课规划教材指导委员会

主 任: 兰家隆

副主任: 黄迪明 黎 明

委 员: (以姓氏笔画为序)

马义玲 王 晓 王 超 王秀华

孙淑霞 兰清昭 李 进 伍治林

刘成安 刘益和 杨明广 杨果仁

赵 刚 黄 敏 蒋明礼 蔡 淮

戴彦群

出版说明

以四川省计算机等级考试委员会大部分委员为核心组成的编委会,自2002年以来,编写了既适合计算机基础教学同时又适合高校计算机等级考试的计算机系列丛书。该丛书经多所高校使用,受到好评,取得良好反响。

08年春季,四川省计算机应用知识和能力等级考试大纲再次修订。按照新考纲的要求,需要把计算机科学发展的高新技术融入教材中。为此,编委会于08年7月召开了研讨会,拟对原计算机系列教材重新编写。全川近30所大专院校约60名专家教授、一线骨干教师参会,共商教材建设之事。这是一项教材精品的建设工程。四川省内高校几十位优秀的“建设者”们加入到了这项工程中,拧成一股绳,放弃了节假日休息。从炎炎夏日到数九寒天,从家里、办公室到出差公干的旅途中,时刻都忘不了辛勤笔耕,因为他们最能体会这项工程对社会的意义和对所在学校的重要性。经过大半年的努力,这套计算机系列丛书共8种已顺利出版。

本系列教材有如下特点:

1. 权威性

以四川省计算机等级考试委员会(简称省考委)考试语种总负责人兰家隆教授、省考委一级三级命题组负责人黄迪明教授为代表的省考委委员,以四川省高校计算机基础教育研究会为代表的四川省内知名专家教授起到了重要的指导作用。每种教材均由他们担任主编、主审。从编写构思到字句斟酌,无不映射出其闪光的智慧和丰富的教学经验、编写功力。他们的参与,是本系列教材质量的可靠保证。

2. 广泛适应性

本计算机基础系列教材主要针对普通院校、高职高专。因此须按照教育部高教司有关高校计算机基础课的教学要求以及普通院校、高职高专学生的接受能力编写。在这一点上不同于重点院校。本套系列教材参编老师基本上来自于这类学校,他们最清楚所在学校学生用什么教材能达到最好的教学效果。经过集体充分讨论制定出来的编写大纲,最能反映各校的教学实际,他们的参编无疑使本系列教材具有相当大的广泛性和相当强的适应性。

《C程序设计教程》由孙淑霞、代世雄担任主编,负责统稿统校;黄迪明担任主审,负责审定;戴彦群、于春担任副主编,交叉对书稿进行修改和润色。

本教材参编老师有：李思明（第一章）、蒲海波（第二章）、于春（第三、四章）、孙淑霞（第五、六章、附录）、戴彦群（第七、八、九章）。本教材课件初稿大多由编写相应章节的老师制作，最后由成都理工大学工程技术学院计算机科学与技术系整理补充完成。

本教材编委详细地研讨了编写大纲，提出了很好的建议和意见，并参加了部分编写工作。

我们相信，有高校教学一线广大老师的参与，有众多四川省高校计算机等级考试委员会委员的指导，有四川省高校计算机基础教育研究会的支持，本系列教材能建设得更好，一定受到高校的欢迎！

四川省高校计算机基础教材编委会

13018235860 028-85228526(0)

E-mail: zhonghai_6868@yeah.net

2009 年 1 月

目 录

四川省高校计算机等级考试中心指导教材

第 1 章 C 程序设计概述	1	2.3.3 赋值运算符与赋值表达式	31
1.1 算法与程序设计	1	2.3.4 关系运算符与关系表达式	32
1.1.1 算 法	1	2.3.5 逻辑运算符与逻辑表达式	33
1.1.2 程序设计	2	2.3.6 逗号运算符及逗号表达式	35
1.2 C 语言简介	3	2.3.7 自加、自减运算符及其表达式	36
1.2.1 C 语言的发展	3	2.3.8 位运算符	36
1.2.2 C 语言的特点	3	2.3.9 其他运算符	40
1.2.3 C 程序的构成与格式	4	2.4 C 语句概述	41
1.2.4 C 程序的编写与实现	8	2.5 数据的输入与输出	43
本章小结	16	2.5.1 字符输入/输出函数	43
习题 1	16	2.5.2 格式化输出函数 printf()	45
第 2 章 C 程序设计基础	17	2.5.3 格式化输入函数 scanf()	50
2.1 常量与变量	17	本章小结	53
2.1.1 常 量	17	习题 2	55
2.1.2 变 量	18	第 3 章 程序的控制结构	60
2.1.3 标识符	19	3.1 选择结构	60
2.1.4 变量的定义	19	3.1.1 if 语句	60
2.2 基本数据类型	20	3.1.2 if-else 语句	62
2.2.1 数据类型概述	20	3.1.3 嵌套的 if-else 语句	64
2.2.2 整型数据	20	3.1.4 switch 语句	66
2.2.3 实型数据	22	3.2 循环结构	67
2.2.4 字符型数据	24	3.2.1 while 语句	68
2.2.5 数据类型间的混合运算	27	3.2.2 do-while 语句	69
2.3 运算符与表达式	29	3.2.3 for 语句	70
2.3.1 运算符和表达式概述	29	3.2.4 循环的嵌套	72
2.3.2 算术运算符与算术表达式	30	3.2.5 break 语句和 continue 语句	73

3.3 编译预处理	76	6.2.1 函数的调用	146
3.3.1 宏替换	76	6.2.2 函数的返回	149
3.3.2 文件包含	79	6.3 函数的参数	150
本章小结	81	6.3.1 传值调用	151
习题 3	81	6.3.2 传址调用	152
第 4 章 数组	88	6.4 命令行参数	157
4.1 概述	88	6.5 递归调用	159
4.2 一维数组	88	习题 6	163
4.2.1 一维数组的定义和初始化	88	第 7 章 结构体与共用体	171
4.2.2 一维数组元素的引用	89	7.1 结构体	171
4.3 二维数组	92	7.1.1 结构体类型与结构体变量	171
4.3.1 二维数组的定义和初始化	92	7.1.2 结构体数组	176
4.3.2 二维数组元素的引用	93	7.1.3 结构体指针	179
4.4 字符数组	95	7.1.4 结构体类型数据作函数参数	181
4.4.1 字符串与一维字符数组	95	7.2 共用体	184
4.4.2 二维字符数组	97	7.3 枚举	188
4.4.3 字符串的输入与输出	98	7.4 为已有数据类型创建新类型名	191
4.4.4 字符串处理函数	99	7.5 结构体应用—链表处理	194
本章小结	105	7.5.1 链表简介	194
习题 4	106	7.5.2 建立动态链表	196
第 5 章 指针	113	7.5.3 链表的插入	201
5.1 指针和地址	113	7.5.4 链表的删除	204
5.2 指针变量的定义和引用	115	本章小结	208
5.3 指针运算	118	习题 7	209
5.4 指针与数组	120	第 8 章 文件访问	218
5.4.1 指向一维数组的指针	120	8.1 文件访问概述	218
5.4.2 指向二维数组的指针	123	8.1.1 文件及其分类	218
5.5 指针与字符串	128	8.1.2 C 语言对文件的处理方式	220
5.6 指向指针的指针	130	8.1.3 文件访问基础	221
5.7 用于动态内存分配的函数	132	8.2 文件打开与关闭	223
习题 5	136	8.2.1 打开文件	223
第 6 章 函数	143	8.2.2 关闭文件	225
6.1 函数的定义和声明	143	8.3 文件读写	226
6.1.1 函数的引入	143	8.3.1 单个字符读写函数	226
6.1.2 函数的定义	144	8.3.2 字符串读写函数	230
6.2 函数的调用与返回	146	8.3.3 格式读写函数	232

8.3.4 数据块读写函数	234	9.2.2 寄存器变量	253
8.4 文件定位	237	9.2.3 静态内部变量	255
8.4.1 获取当前读写位置	238	9.3 外部变量	257
8.4.2 文件位置指针的移动	239	9.3.1 外部变量的作用域	257
8.5 文件状态和出错处理	241	9.3.2 通过外部变量在函数之间传递	
本章小结	242	数据	261
习题 8	243	本章小结	261
第 9 章 变量的作用域与生存期	250	习题 9	262
9.1 作用域与存储类别概述	250	附录 I ASCII 码表	266
9.2 内部变量	251	附录 II C 语言的关键字	269
9.2.1 自动变量	252	附录 III 运算符的优先级与结合性	270

第 1 章

C 程序设计概述

程序是为解决某特定问题而用计算机语言编写的语句（指令）序列。程序设计是指设计、编制、调试程序的方法和过程。程序设计往往以某种程序设计语言为工具，给出这种语言下的程序。C 语言是国际上广泛流行的程序设计语言之一。

1.1 算法与程序设计

程序设计涉及程序设计语言、程序设计方法、数据结构和算法等。程序设计语言和环境是编写程序的工具，程序由它们制造；程序设计方法是编程的指导思想，它决定用什么样的方式来组织编写程序；数据结构则是加工的对象；算法是灵魂，是解决问题的方法和步骤。

1.1.1 算法

计算机解题一般可分解成若干操作步骤，通常把完成某一任务的操作步骤称为求解该问题的算法，即算法是解决某一特定类型问题的有限运算序列（指令的有限集合）。例如：有两只杯子 A 和 B，A 杯里装满水，B 杯里装满醋，若要交换 A 和 B 两只杯子里所装液体，需要经过以下步骤来完成：

- ① 找一个空杯 C。
- ② 将 A 杯里的水倒入 C 杯。
- ③ 将 B 杯里的醋倒入 A 杯。
- ④ 将 C 杯里的水倒入 B 杯。

即完成 A 和 B 杯中液体的交换。

对于一个问题，如果可以通过一个计算机程序，在有限的存储空间内运行有限长的时间而得到正确的结果，则称这个问题是算法可解的。但算法不等于程序，也不等于计算方法。程序作为一种算法的描述，由于受到计算机系统运行环境的限制，通常还需要考虑更多的细节问题。一个算法，一般应具有以下几个基本特征。

1) 确定性：算法的确定性是指算法中的每一条规则、每一个操作步骤都应当是确定的，不允许存在多义性和模棱两可的解释。

2) 有穷性: 算法的有穷性是指任意一个算法必须在执行有限步骤后结束。也就是说, 任何算法都必须在有限的时间内完成。同时, 有穷性还隐含了算法的执行时间应该合理, 如果一个算法要执行几十年才结束, 该算法也就失去了它的实用价值。如数学中的无穷级数, 在实际计算时只能根据精度要求取有限项。

3) 拥有足够的情报: 一个算法是否有效, 还取决于为算法所提供的情报是否足够。通常, 一个算法有 0 个、1 个或多个输入。一个算法执行的结果总是与输入的初始数据有关, 不同的输入将会有不同的输出结果。当输入不够或输入错误时, 会导致算法无法执行或执行错误。

4) 可行性: 算法的每一步操作都应该是可执行的。例如, 当 $B=0$ 时, A/B 就无法执行, 不符合可执行性的要求。

1.1.2 程序设计

当确定好一个算法之后, 怎么让计算机来完成相应的操作呢? 有了算法之后, 还需要告诉计算机要“做什么”和“怎么做”, 这个过程就是程序设计。比如说, 计算一个矩形的面积大体需要以下计算步骤:

- ① 说明在计算过程中要用到的变量。
- ② 给定矩形的长和宽。
- ③ 计算矩形的面积。
- ④ 将计算结果(面积)输出到屏幕或打印机上。

在确定了各个步骤之后, 怎么让计算机理解人的意图, 并按照给定的步骤完成计算, 这就有一个人和计算机之间的沟通问题。计算机并不懂得人类的语言, 它只能识别二进制的信息。在计算机产生的初期, 人们为了让计算机工作, 必须编写出由 0 和 1 所组成的一系列的指令, 通过它指挥计算机工作。这种全由 0 和 1 组成的指令, 称为机器语言, 它是第一代程序设计语言。机器语言是紧密依赖于计算机硬件的, 不同型号的计算机的机器语言是不相同的。用机器语言写程序难学、难记、难写、难修改, 而且在不同计算机之间互不通用。为了克服机器语言抽象、难理解、难记忆等缺点, 人们采用了汇编语言, 它是用一些特定“助记符号”代替 0 和 1 来表示指令, 如“ADD A,B”就是一条执行加法的指令。用汇编语言编写程序与用机器语言编写程序的步骤相似, 它们的指令是一一对应的。由于机器语言和汇编语言都依赖于具体机器, 所以被称为“低级语言”。用低级语言编写程序很不直观, 烦琐枯燥, 工作量大, 不具有通用性。

20 世纪 50 年代出现了用于程序设计的“高级语言”, 它比较接近于人们习惯使用的自然语言(英语)和数学语言。因此, 用高级语言编写程序直观易学, 易理解, 易修改, 通用性强。从 1954 年出现第一种高级语言以来, 全世界先后出现了 2500 种以上的高级语言, 其中应用比较广泛的有 100 多种, 影响较大的有: C 语言、BASIC、COBOL、FORTRAN、LISP、PROLOG、C++、Java 等。

有了计算机语言之后, 就可以按照各计算机语言规则, 按照上面计算矩形面积的步骤编写程序。

1.2 C 语言简介

C 语言是国际上应用最广泛的计算机语言之一。它不仅可以用于编写系统软件,如操作系统、编译系统等,还可以用于编写应用软件。著名的 UNIX 操作系统最初就是用 C 语言设计的。

1.2.1 C 语言的发展

C 语言是一种面向过程的高级语言,是集汇编语言和高级语言的优点于一身的程序设计语言;具有高效、灵活、功能丰富、表达力强和移植性好等特点,备受程序员青睐。C 语言是由 UNIX 的研制者丹尼斯·里奇(Dennis Ritchie)和肯·汤普逊(Ken Thompson)于 1970 年研制出的 B 语言的基础上发展和完善起来的。1978 年以后,C 语言先后移植到大、中、小、微型计算机上。C 语言很快风靡全世界,成为世界上应用最广泛的高级程序设计语言之一。

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 语言的发展和扩充,制定了新的标准草案(83 ANSI C),后来于 1987 年又颁布了另一个 C 语言标准草案(87 ANSI C)。1989 年制定出 ANSI C 的标准,又称为 C89。1995 年,经过修订增加了一些库函数,出现了 C++的一些特性,使 C89 成为 C++的子集。最新的 C 语言标准是在 1999 年颁布并在 2000 年 3 月被 ANSI 采用的 C99,但由于未得到主流编译器厂家的支持,并未被广泛使用。本书的叙述基本上以 C89 为基础。

1.2.2 C 语言的特点

C 语言之所以能够广为流传,是因为它有很多不同于其他程序设计语言的特点。其主要特点有:

1) 数据类型丰富。C 语言除了整型、实型、字符型等基本数据类型外,还具有数组、指针、结构体、共同体等高级数据类型,能够用于描述各种复杂的数据结构(如队列、栈、链表等)。指针数据类型的使用,使 C 程序结构更为简化、程序编写更为灵活、程序运行更为高效。

2) 运算符种类丰富。C 语言具有数十种运算符,除了具有一般高级语言具有的运算功能外,还可以实现以二进制位为单位的位运算,直接控制计算机的硬件,还具有自增、自减和各种复合赋值运算符等。C 程序编译后生成的目标代码长度短、运行速度快、效率高。

3) 符合结构化程序设计的要求。C 语言提供的控制结构语句(如 if-else 语句、while 语句、do-while 语句、switch 语句、for 语句)使程序结构清晰,其函数结构使程序模块具有相对独立的功能,便于调试和维护,有利于大型软件的协作开发。

4) 可移植性好。只要是使用 ANSI C 定义的标准函数库来开发,而其中不包含一些中断处理或厂商单独提供的函数库,那么所开发出来的程序将具有相当高的可移植性,几乎不作修改就可用于各种计算机和各种操作系统。

C 语言的这些特点使 C 语言很快应用到了各计算机应用领域中的软件编写, 如信息管理、科学计算、图形图像处理、实时控制等软件。

然而, C 语言也不是十全十美的, 它也有缺点。它的语法限制不太严格, 例如, 缺乏数据类型的一致性检测和不进行数组下标越界检查。正因为 C 语言允许编程者有较大的自由度, 使 C 程序容易通过编译, 但也增加了查找程序运行错误的复杂度。

1.2.3 C 程序的构成与格式

首先看两个简单的 C 语言程序例子, 然后分析 C 语言程序的构成。

【例 1-1】在屏幕上输出一串字符。

```
/* 参考源程序 1-1.C */
#include <stdio.h>                                /* 嵌入文件 stdio.h 到当前源程序中 */
void main()                                       /* main() 表示主函数, void 表示 main() 不返回值 */
{                                                 /* 函数体开始 */
    printf("My first C program. \n");           /* 屏幕上输出一串字符 */
}                                                 /* 函数体结束 */
```

执行该程序将在屏幕上显示如下信息:

My first C program.

该程序由一个主函数 `main()` 组成, `main` 是一个特殊的名字, 它指出程序执行的起始位置。`main` 后的圆括号指明 `main` 是一个函数, `main` 前的 `void` 用于声明该函数无返回值。函数体内 (由大括号 `{}` 括起来的部分) 只有一条语句, 该语句是调用 C 语言提供的库函数 `printf()` (格式输出函数) 输出双引号中的字符串, 其中的 “`\n`” 是转义字符, 表示换行, 即当在屏幕上输出到转义字符 “`\n`” 时, 光标移到下一行的起始位置, 以后再有输出就从该位置开始显示。

程序的第一行 “`#include <stdio.h>`” 的作用是告诉编译系统, 有关标准输入输出 (standard input and output) 函数的定义在 `stdio.h` 文件中, 如本程序中用到的 `printf` 函数在 `stdio.h` 文件中有定义。有关这方面的知识将会在以后章节中学习, 现在只须记住在用标准输入输出函数时, 在程序的第一行加上这个语句即可。

程序中每一行后面都是注释。注释可以由 “`/*`” 和 “`*/`” 括起来的任何文字, 它可以出现在程序的任何地方, 用来说明程序段的功能、语句行的作用、变量的作用等内容, 主要用于向程序阅读者进行说明和交流, 使读者能读懂程序, 便于程序的调试。每一个程序编写者都应该养成在必要的位置上加写注释的良好习惯。

C 语言的每一条语句都以分号 “`;`” 结束。为了清晰地显示程序的结构, 程序的书写应该采用缩进格式, 一行只书写一条语句。

【例 1-2】完成 1.1.2 节提出的求矩形面积的程序, 根据输入的宽和高求矩形的面积。

```
/* 参考源程序 1-2.C */
#include <stdio.h>
```

```

void main()                                /* 求矩形的圆面积 */
{                                           /* 函数体开始 */
    float area;                            /* 定义实型变量, 用于存储矩形的面积 */
    float width,height;                   /* 定义实型变量, 表示矩形的宽和高 */
    printf("请输入矩形的宽和高: ");       /* 屏幕上显示"请输入矩形的宽和高:" */
    scanf("%f%f",&width,&height);        /* 输入矩形的宽和高 */
    area=width*height;                    /* 计算矩形面积 */
    printf("\n area=%f\n",area);          /* 输出矩形面积 */
}                                           /* 函数体结束 */

```

} 声明部分

} 可执行部分

程序运行实例如下:

请输入矩形的宽和高: 5 4✓

area=20.000000

C 语言的函数体主要由两部分组成: 声明部分和可执行部分。声明部分用于定义和说明变量、数组等; 可执行部分由可执行语句和函数调用等语句行组成。其中函数体中的第 4 行和第 5 行是对函数体中要使用的变量进行定义, 称为声明部分; 第 7 行是在屏幕上显示“请输入矩形的宽和高:”, 这种方法常用于输入前对用户的提示; 第 8 行调用输入函数输入矩形的宽和高; 第 9 行是计算矩形的面积, 并将计算结果存储到变量 `area` 中; 第 10 行是输出变量 `area` 的值, 即矩形的面积。如果输入的宽 `width=5`、高 `height=4`, 则屏幕上显示的是:

area=20.000000

由此可见, 上面两个 C 程序的执行都是从 `main()` 函数开始, 依次执行函数体中各条语句, 直到结束。

C 程序的组成特点如下:

- 1) 一个 C 源程序由函数构成, 其中有且仅有一个主函数 `main()`, 它是程序的入口。
- 2) 一个函数由两部分组成: 函数首部和函数体。函数首部即函数的第一行, 包括函数名、函数类型, 如果是带参函数还会有函数参数和参数类型; 函数体即函数首部下面的花括号内的部分, 包括声明部分和可执行部分。
- 3) 分号“;”是 C 语句的一部分, 每一条语句均以分号结束。
- 4) C 程序书写格式自由, 一行内可写多条语句, 一个语句也可写在多行上。
- 5) 程序的注释部分应括在 `/*...*/` 之间, `/` 和 `*` 之间不允许留有空格; 允许注释部分出现在程序的任何位置上。
- 6) C 语言本身没有输入输出语句。输入输出语句的操作由库函数 `scanf` 和 `printf` 等函数来完成, 在 `stdio.h` 文件中定义了 `scanf` 和 `printf` 函数。

在上述例子中还出现了 C 语言程序设计中的标识符、关键字、运算符、常量和变量、函数调用等基本概念。

1. 标识符

例 1-2 中的 `width`、`height` 和 `area`, 在 C 语言中称为标识符。

C 语言的标识符用于给程序中的常量、变量、函数、文件指针和数据类型等命名。用户可以根据需要进行命名, 形成用户标识符。标识符的命名规则如下:

- 1) 标识符由英文字母 (a~z, A~Z) 或下划线开头, 后面可跟字母、数字或下划线。
- 2) 标识符中的大、小写英文字母的表示不同意义, 即代表不同的标识符, 通常 C 程序中的变量用小写字母, 符号常量用大写字母。
- 3) C 编译系统对标识符所用的字符个数有不同的规定, ANSI C 可以识别标识符的前 31 个字符, 但有的 C 编译系统只识别前 8 个字符。
- 4) 取名时, 应当尽量遵循“见名知意”和“简洁明了”的原则。

例如: smart, key_board 为正确的标识符, 而 key.board, 3days 为错误的标识符。

2. 关键字

关键字又称保留字。它是 C 语言中已预先定义, 且具有特定含义, 不能作其他用途使用的标识符。例 1-2 中的 float 就是一个 C 语言的关键字, 它指明 area、width 和 height 的类型为单精度实型。

C 语言共有 32 个关键字, 如表 1-1 所示。所有关键字都用小写英文字母表示, 且这些关键字不允许用作用户标识符。

表 1-1 关键字

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

3. 运算符

运算符是用来表示某种运算的符号, 其中有的由一个字符组成, 如 +、-、*、/ 等; 有的由多个字符组成, 如 <=、<<、&&、!= 等。C 语言的运算符主要有以下几类:

- 1) 算术运算符 (+ - * / %)
- 2) 关系运算符 (> < == >= <= !=)
- 3) 逻辑运算符 (! && ||)
- 4) 位运算符 (<< >> ~ | ^ &)
- 5) 赋值运算符 (= 及其扩展赋值运算符)
- 6) 条件运算符 (? :)
- 7) 逗号运算符 (,)

- 8) 指针运算符 (* 和 &)
- 9) 求字节数运算符 (sizeof)
- 10) 强制类型转换运算符 ((类型))
- 11) 分量运算符 (. ->)
- 12) 下标运算符 ([])

有些运算符具有双重含义, 例如 “%” 可以作为求余运算符; 但是, 当它出现在输入、输出函数中时, 就是 “格式控制符” 了。

4. 常量和变量

在程序运行过程中不发生变化的量, C 语言将其称为常量。C 语言的常量分为整型常量、实型常量和字符型常量。如 3, 8, -4 是整型常量, 5.7, -3.2 是实型常量, 'a', 'b' 为字符型常量。

变量是指在程序运行过程中其值可以改变的量。如例 1.2 中的 width 和 height 就称为变量。每一个变量都有一个名字, 根据变量的不同类型, 系统将为每一个变量分配相应的内存单元。例如, 系统为整型 (int 型) 变量分配 2 个字节的内存单元, 为实型 (float 型) 变量分配 4 个字节的内存单元, 为字符型 (char 型) 的变量分配 1 个字节的内存单元。内存单元中存放的是变量的值。程序执行过程中对其值的读、写是通过变量名找到相应的内存单元来实现的。

5. 函数调用

例 1-2 程序中调用的输出函数 printf() 和输入函数 scanf() 是 C 语言的库函数中为用户提供的。C 语言中的函数分为: 系统提供的库函数和用户自定义函数。对于库函数, 用户可以直接调用; 用户自定义函数是用户用以解决专门问题所定义的函数 (参见第 6 章)。

【例 1-3】从键盘上输入两个整数, 求其中的较大数。

```
/* 参考源程序 1-3.C */
#include<stdio.h>
void main()
{
    int x,y,z;

    printf("输入两个整数: ");      /* 提示输入 */
    scanf("%d%d",&x,&y);           /* 输入两个整数 */
    z=max(x,y);                     /* 调用求最大值函数 */
    printf("max=%d",z);             /* 输出最大值 */
    getch();
}
int max(int x,int y)                /* 自定义求最大值函数 */
{
    return(x>y?x:y);               /* 返回最大值 */
}
```


程序的运行情况如下：

输入两个整数：4 9✓

max=9

上面程序由两个函数组成，其中 `max()` 是自定义函数，用于求两个数中的较大数。通过 `main()` 函数调用 `max()` 函数，由 `max()` 函数返回较大数。程序从 `main()` 函数开始执行，执行到函数调用处，转去执行被调用函数 `max()`；`max()` 函数执行完后，再返回到主函数继续执行函数调用语句后面的语句行。

1.2.4 C 程序的编写与实现

为了编译、连接和运行 C 程序，必须要有相应的 C 编译系统，通过编译系统将 C 源程序转换为可执行的二进制程序文件。目前使用的大多数 C 编译系统都具有集成开发环境，集 C 语言程序的建立、编辑、编译、连接、运行和程序的动态调试于一体。常用的编译系统有 Turbo C 2.0，Visual C++6.0 等。Turbo C 2.0 是用于 DOS 环境的，主要通过键盘操作；Visual C++6.0 是用于 Windows 环境，采用图形界面，操作上更加方便。同时，Visual C++ 集成开发环境还支持 C++ 程序的编译。本节将分别介绍在 Turbo C 和 Visual C++ 环境中进行 C 程序的建立、编辑、编译、连接和运行。

初学编程，最好先把 C 程序编好后，再到计算机上进行调试。当编程能力达到了一定的熟练程度后，就可直接在计算机上写程序了。

1. Turbo C 中实现 C 程序

要实现一个 C 程序，首先得利用编辑器编辑好 C 程序。编辑就是把程序输入到计算机中，并以文本文件的形式保存到磁盘上。用 C 语言编写的程序文件叫源程序文件，其文件的扩展名为 “.C”。

无论新编写一个程序，还是修改一个原有的程序，首先要启动 Turbo C，进入 Turbo C 的集成环境。

(1) Turbo C 的启动及 Turbo C 主屏幕

在命令行键入 TC 并回车或在 Windows 环境中找到并打开 Turbo C (TC) 文件夹，双击 TC 图标，即可启动 Turbo C，屏幕上将出现 Turbo C 的集成界面，如图 1-1 所示。其中最顶上一行为 Turbo C 主菜单，中间窗口为编辑区，接下来是信息窗口，最底下一行为功能键和热键说明区。这四个窗口构成了 Turbo C 的主屏幕，以后的编辑、编译、调试以及运行都将在这个主屏幕中进行。

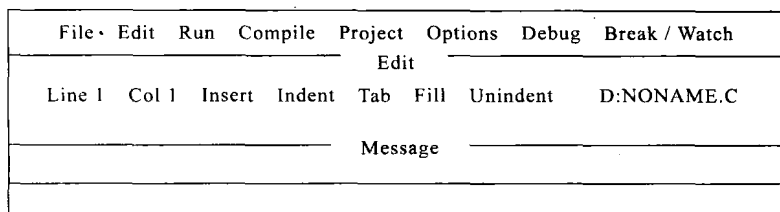


图 1-1 Turbo C 的集成界面