

高等学校计算机基础教育课程通用教材
高等学校在校学生计算机等级考试用书

C 语言程序设计

(第二版)

孙家启 主编



中国科学技术出版社

高等学校计算机基础教育课程系列教材
高等学校在校学生计算机等级考试用书

C 语 言 程 序 设 计

(第二版)

孙家启 主编

中国科学技术出版社

·北 京·

图书在版编目(CIP)数据

C 语言程序设计/孙家启主编. - 2 版. - 北京: 中国科学技术出版社, 1998

ISBN 7-5046-2490-X

I. C… II. 孙… III. C 语言-程序设计 IV. TP312

中国版本图书馆 CIP 数据核字(98)第 09806 号

中国科学技术出版社出版

北京海淀区白石桥路 32 号 邮政编码: 100081

新华书店北京发行所发行 各地新华书店经售

北京市里顺印刷厂印刷

*

开本: 787 毫米 × 1092 毫米 1/16 印张: 16.25 字数: 390 千字

1998 年 7 月第 2 版 2000 年 7 月第 3 次印刷

印数 4 501-6 500 册 定价: 20.00 元

再版说明

为了适应经济建设、社会发展和科技进步的需要,改革并完善高等学校各学科专业学生的知识和能力结构,提高高校的教学质量,安徽省教育委员会于1994年建立了高等学校计算机基础教育课程教材编审委员会,组织高等学校相关学科的专家、学者,统一编写了高等学校计算机基础教育课程的系列教材:《计算机应用基础》、《FoxBASE⁺程序设计》、《True BASIC 语言程序设计》、《FORTRAN77 结构化程序设计》、《PASCAL 语言程序设计》、《C 语言程序设计》、《计算机硬件技术基础》、《计算机软件技术基础》共八种。

本系列教材经国家教育委员会教材管理部门同意,面向全国高等学校发行。经过四年多高等教育同行的广泛使用,认为该系列教材结构严谨、重点突出、文字简练,适合高等学校的教学需要,体现了国家教育委员会要求面向21世纪开展并建立新的教学内容和课程体系改革计划的精神,是高等教育教学改革的一项重要成果。

本次再版,针对高等学校各学科专业学生开展计算机基础教育三个层次教学的不同要求和需要,以及全国省、市教育委员会联合开展计算机基础教育考试的新形势,修改了部分内容,增加了最新的计算机应用科学技术知识。

本次再版,得到了安徽省教育委员会领导和出版社的大力支持,编审委员会谨此感谢。

安徽高等学校计算机基础教育课程

教材编审委员会

1998年2月8日

再 版 前 言

C语言是近年来国内外得到最迅速推广使用的一种现代编译型程序设计语言,它兼顾了多种高级语言的特点并具备汇编语言的功能。C语言程序处理功能强、运算速度快、目标效率高,具有完善的模块程序结构,可移植性好,而且可以直接实现对系统硬件及外部设备接口的控制,具有较强的系统处理能力。在当今世界技术先进国家中,使用C语言进行程序设计已成为软件开发的一个主流。C++语言是对C语言进行扩充的一种程序设计语言,它包括了C语言的全部功能。1983年在C语言的基础上推出的面向对象的程序设计语言C++,在90年代得到迅猛发展。C++语言今后将成为最流行的一种计算机主流语言。掌握好C/C++语言已成为当今软件工作者的必备条件之一。

为了在我国高等学校更快地推广和普及C语言,编者在多年从事C语言程序设计教学和科研工作的基础上,按照国家教委全国高等工科大学计算机基础课程教学指导委员1993年制订的C语言程序设计教学要求,尽全力编写出内容新颖、逻辑性强、循序渐近、深入浅出,并且适合我国高等学校实际情况的C语言程序设计教材。

本书采用了当前最新的标准C语言——美国国家标准C语言(87 ANSI 标准C),以我国目前高等院校广泛使用的IBM PC系列各档次(80386、80486、80586等)微型机上的C编译系统Turbo C 2.0为实现的版本,同时兼顾Unix操作系统等各种C编译系统内容,全面系统地讨论了C语言及其程序设计方法和技巧。书中对指针概念、函数间数据传递以及结构体和联合体等难点内容,进行了深入的分析和解释。本书最后一章介绍Turbo C简介及错误分析,进一步突出了本书的实用性。为了帮助考生参加全国高等学校计算机等级考试,本书附录中提供C语言程序设计(二级)考试标题和参考答案。

为了便于读者牢固掌握本书知识,并能尽快地把它们应用到实际开发中去,书中给出了大量难易不等的例题,每章之后都有小结并配备一定数量的习题。

本书共九章。第一、第二、第七章及附录由孙家启编写;第三、第八章由张维勇编写;第四、第六章由吴国凤编写;第五、第九章由赵保华、苏仕华编写。主审程慧霞,主编孙家启,副主编赵保华。全书由孙家启最后修改定稿。根据教学需要,这次又进行了修订,力求适合读者的需要。

本书请安徽大学副校长、计算机系教授程慧霞对全书进行审阅、指导,并提出了不少好的修改意见;本书第二版在修改过程中得到了安徽省教委大力支持和帮助;中国科学技术出版社做了大量工作,使本书能尽早与读者见面,在此一并表示感谢。由于编者水平有限,书中难免存在错误和不当之处,恳请读者及专家、同行提出宝贵意见。

编 者
1998年2月

目 录

前言

第一章 C 语言概述	(1)
1 C 语言的发展与特点	(1)
2 C 语言的基本程序结构	(3)
3 C 语言程序的开发过程	(5)
4 C 语言的基本语法单位	(8)
小 结	(9)
习题一	(10)
第二章 C 语言基础	(11)
1 C 语言的数据类型	(11)
2 常量	(11)
3 变量	(15)
4 赋值运算与算术运算	(18)
5 关系运算与逻辑运算	(20)
6 位运算	(22)
7 其他运算	(24)
8 运算符的优先级和结合规则	(25)
9 赋值语句	(27)
10 数据的输入输出	(28)
11 程序设计举例	(33)
小 结	(38)
习题二	(39)
第三章 C 语言程序的基本控制结构	(42)
1 程序的三种基本结构	(42)
2 顺序结构	(43)
3 分支结构	(45)
4 循环结构	(56)
5 程序设计举例	(72)
小 结	(74)
习题三	(74)
第四章 数组	(77)
1 一维数组	(77)

2 多维数组	(81)
3 字符数组	(87)
4 程序设计举例	(92)
小 结	(97)
习题四	(97)
第五章 指针	(99)
1 地址、指针变量和地址运算	(99)
2 指针与函数参数	(104)
3 指针与数组	(105)
4 指针数组和指向指针的指针	(108)
5 命令行参数	(112)
6 程序设计举例	(114)
小 结	(116)
习题五	(117)
第六章 函数	(119)
1 模块化软件与函数的基本结构	(119)
2 函数的定义与调用	(121)
3 变量的存储类型及其作用域	(124)
4 函数间的数据传送	(127)
5 数组与函数参数	(134)
6 指针型函数	(137)
7 递归函数	(138)
8 函数指针	(140)
9 编译预处理	(143)
10 程序设计举例	(147)
小结	(156)
习题六	(156)
第七章 结构、联合、枚举和用户定义类型	(159)
1 结构类型	(159)
2 联合	(177)
3 枚举	(182)
4 用户定义类型	(183)
5 程序设计举例	(185)
小 结	(190)
习题七	(190)
第八章 文件	(191)

1 C 文件的概述	(191)
2 标准设备文件的输入输出	(193)
3 数据文件的输入输出	(195)
4 文件的定位操作	(206)
5 文件的错误检测	(208)
6 程序设计举例	(209)
小 结	(212)
习题八	(213)
第九章 Turbo C 简介和错误分析	(215)
1 Turbo C 简介	(215)
2 常见错误分析	(224)
附录	(232)

第一章 C 语言概述

C 语言是一种通用性程序设计语言,早期的 C 语言被用来书写 Unix 操作系统,它是和 Unix 操作系统相辅相成地发展的。随着 Unix 操作系统在国际上的广泛流行,C 语言已被人们普遍接受。现在,几乎在各种型号的微型机和大、中、小型计算机上都配有 C 语言编译系统。C 语言已广泛用于描述系统软件(操作系统、语言处理、系统实用程序)、数据处理、科学工程数值计算等多个领域,深受专业工作者和广大用户的欢迎。

C++ 语言是对 C 语言进行扩充而成的一种面向编译对象的程序设计语言,它几乎包括了 C 语言的全部功能。要学好 C++ 语言,首先要先学好 C 语言。当然,由于 C++ 语言又引入了一些连同 C 语言在内的其他高级语言所不具备的概念,可以预期,C++ 语言今后将成为最流行的一种计算机语言。

本章将简要介绍 C 语言和 C++ 语言的发展与特点;C 语言的基本程序结构;C 语言程序的开发过程,以及 C 语言的基本语法单位等,以便对 C 语言有一个总的印象,为以后各章学习打下基础。

1 C 语言的发展与特点

1.1 C 语言的发展

在种类繁多的计算机程序语言家族中,70 年代初期,又增添了一名新成员——C 语言。

C 语言是国际上广泛流行的、很有发展前途的一种编译型程序设计语言,它的前身是英国剑桥大学的马丁·理查德(Martin Richards)在 60 年代开发的 BCPL 语言。1970 年,美国贝尔实验室的肯·苏姆普逊(Ken Thompson)在软件开发中,继承和发展了 BCPL 语言,进而提出了“B 语言”,并用 B 语言在美国 DEC 公司的 PDP—7 机上编写了第一个 Unix 操作系统。此后,在美国贝尔实验室进行的更新型的小型机 PDP—11 的 Unix 操作系统的开发工作中,戴尼斯·M·利奇(Dennis M. Ritchie)和布朗·W·卡尼汉(Brian W. Kernighan)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设计语言——C 语言。1978 年, Brian W. Kernighan 和 Dennis M. Ritchie(简称 K&R)合著了具有深远影响的名著《The C Programming Language》,被称为标准 C。1983 年,美国国家标准化协会(ANSI)根据 C 语言问世以来的各种版本对 C 的发展和扩充,制定了新的标准,称为 ANSI C。1988 年 K&R 按照 ANSI C 标准又重新写了他们的名著。目前人们常将 1978 年的标准 C 称为旧标准,将 ANSI C 称为新标准。当前,国内最流行的 IBM-PC 系列微型机配合 MS-DOS 已经装配了很多种版本的 C 语言编译程序,计有 Turbo C 的 TC, Microsoft C 的 MC, Quick C, Computer Innovations C86, Lattice C 的 LC, C-Systems C, Caprock small-C, Desmet C, Intellect Associates C88, Quantum C, Supersoft C 和 Telecon C 等。他们的不同版本又略有差异,因此读者应了解所用的计算机系统的 C 编译的特点和规定(可以参阅有关手册)。

C++ 语言是 80 年代后期研制成功的一种面向对象的程序设计语言 O—OPL(Object Oriented Programming Language),是一种很具特色、功能很强的编程语言,目前已进行的研制和大量实践,已显示 C++ 语言具有很强的生命力和发展前景。C++ 较 C 优越主要是 C++ 支持数据抽

象并具有面向对象的基本特点。

目前,以 AT&T 的 C++ (V2.0)为基础,开发出许多不同名称的 C++ 新版本,如 Zortech 公司根据 AT&T 公司 C++ (V2.0)版本并结合最新的标准化 C 语言(称为 ANSI C)研制成功的新版本 Zortech C++ (V2.0),和 1991 年底,由 Borland 公司推出,完全支持 ANSI C 和 Turbo C++、全局优化、窗口编程功能进一步增强、集成开发环境进一步完善的最新版本 Borland C++ (V3.0)。

事实上,C++ 广泛地用于软件研究领域。例如采用 C++ 开发面向对象的新型 DOS 操作系统;用 Borland C++ (V3.0)支持面向对象窗口的应用与开发;提出用 C++ 库实现数据库的管理;C++ 的用户界面以及 C++ 的专家系统等等。因此,完全可以相信:C++ 有着十分光辉灿烂的未来。

1.2 C 语言的特点

一种语言之所以能存在和发展,并具有生命力,总是有些不同于(或优于)其他语言的特点。C 语言有以下几个基本特点:

(1)C 语言简洁、紧凑,使用方便、灵活 C 语言一共只有 32 个保留字,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,相对其他语言源程序短,因此输入程序时工作量少。

(2)C 语言是处于汇编语言和高级语言之间的一种记述性程序设计语言 他允许直接访问地址,能进行位(Bit)运算,能实现汇编语言大部分功能,可以直接对硬件进行操作。

(3)C 语言是一种结构化程序设计语言 即程序的逻辑结构可以用顺序、分支和循环三种基本结构组成。C 语言具有结构化控制语句(如 if~else、while、do~while、switch、for 等语句),十分便于采用由顶向下、逐步求精的结构化程序设计方法。C 语言程序的函数结构,十分利于把整体程序分割成若干相对独立的功能模块,并且为程序模块间的相互调用以及数据传递提供了便利。因此,用 C 语言编制的程序,具有容易理解、便于维护的优点。

(4)C 语言运算符丰富 它的运算符包含的范围很广泛,共有 34 种运算符。除一般高级语言使用的 +、-、*、/四则运算及与(and)、或(or)、非(not)等逻辑运算功能外,还可以实现以二进制位(bit)为单位的位与(&)、位或(|)、位非(~)、位异或(^)以及移位(>>、<<)等位运算,并且具有如 a++、b--等单项运算和 +=、-=、*=、/=等复合运算功能。

(5)C 语言的数据类型丰富 它的数据类型有:整型、实型、字符型、数组类型、指针类型、结构体类型、联合体类型和枚举类型等,能用来实现各种复杂的数据结构。因此,C 语言具有较强的数据处理能力。

(6)C 语言程序中可以使用如 #define、#include 等编译预处理语句,能进行字符串或特定参数的宏定义,以及实现对外部文本文件的读取和合并。同时还具有 #if、#else 等条件编译预处理语句。这些功能的使用提高了软件开发的工作效率,并为程序的组织和编译提供了便利。

(7)C 语言程序可移植性好 C 语言程序本身并不依存于机器硬件系统,从而便于在硬件结构不同的机种间和各种操作系统实现程序的移植。

由于 C 语言具有上述众多特点,近年来迅速地得到普及和应用。许多大型的软件系统都用 C 语言编写,许多以前只能用汇编语言处理的问题现在可以改用 C 语言来处理。C 语言被称为“高级汇编语言”。

C 语言的优点很多,但和其他程序设计语言一样,它也有弱点,如运算符的优先级较多,有些还与常规约定不同,不便记忆;某些语法部分不易用形式化方法描述;各种 C 语言版本

之间略有差别,缺乏统一的标准;C语言不是强类型的语言,它强调灵活、高效的同时,在一定程度上牺牲了某些安全性,如类型检验太弱,转换比较随便等。因此,C语言对程序设计人员提出了较高的要求,尤其在使用C语言的某些高级手段时更是如此。但是,C语言的优点远远超过了它的弱点,这些优点使C语言具有强大的吸引力。实际经验表明,程序设计人员一旦接触了这种语言,并且有一定程度设计经验后,就会对它爱不释手。

2 C语言的基本程序结构

任何一种计算机程序设计语言,都具有特定的语法规则、语义和一定的表现形式。程序的书写格式和程序的构成规则是程序语言表现形式的一个重要方面。按照规定的书写格式和构成规则书写程序,不仅可以使程序设计人员和使用程序的人容易理解,更重要的是把程序输入给计算机时,计算机能够充分认识,从而能够正确执行它。

C语言程序是由一个或多个具有相对独立功能的程序模块集合而成的,这样的程序模块称为函数。为了更直观地了解C语言的基本程序结构,下面给出两个具体的C语言程序。

例 1-1 C语言的基本程序结构例子 1。程序如下:

```
/* C1-1.C */
main()
{
    int a, b, sum;
    a = 135; b = 246;
    sum = a + b;
    printf("sum is %d\n", sum);
}
```

该程序的作用是求两个整数 a 和 b 之和 sum 。其中,第一行以“/*”开头到“*/”结尾之间的内容是一个注释,它可在一行书写或分多行书写,可写在程序的任何位置。注释的作用是帮助阅读和理介程序,编译时,注释行被忽略掉,即不产生代码行。第二行 `main()` 是主函数,是C语言程序必须有的函数。函数体由大括号 `{}` 括起来的。程序第四行是变量说明,说明 a 、 b 和 sum 为整型(`int`)变量。第五行是两个赋值语句,使 a 和 b 的值分别为 135 和 246。第六行使 sum 的值为 $a + b$,第七行中的“%d”是输入输出的“格式说明”,用来指定输入输出的数据类型和格式,即表示“十进制整数类型”。`printf` 函数中括弧内最右端 sum 是输出的变量,现在它的值是 381(即 $135 + 246$ 之值)因此输出一行信息为:

sum is 381

例 1-2 C语言的基本程序结构例子 2。程序如下:

```
/* C1-2.C */
/* printf string as uppercase */
#include <stdio.h>
#define SIZE 80
main()
{
    char str[SIZE];
    int i;
```

```

    gets(str);
    for(i=0;str[i]!='\0';i++)putu(str[i]);
}

putu(char ch);
{
    char cc;
    cc=(ch>='a'&&ch<='z')?ch+'A'-'a':ch;
    putchar(cc);
}

```

该程序是由名字称为 main 和 putu 的两个函数组成。C 语言程序中,包含一个且只能有一个名字为 main() 的主函数,也可以有一个 main() 主函数和若干个其他函数组成。主函数之外的函数由用户命名,如上例的 putu 函数。一个函数由两部分组成:函数的说明部分(包括函数类型、函数名、函数形参名、形参类型);函数体,即函数说明部分下面大括号{...}内的部分。对于 int 型和无返回值的函数,如上例 putu 函数的函数类型,可以缺省说明。

C 语言程序的执行是从 main() 函数开始的,main() 主函数中所有语句执行完毕,则程序执行结束。如上述程序是从 6 行的 { 开始,执行到 11 行的 } 结束。当程序执行到第 10 行的语句时,程序控制转移到 14 行的函数 putu 中。执行完函数 putu 中的所有语句后,再返回 main() 主函数中继续运行。这种控制转移叫做调用函数 putu。在程序中除了可以调用用户根据需要自己编制设计的函数外,还可以调用由系统提供的库函数。C 的函数库十分丰富,标准 C 提供 100 多种库函数,Turbo C 和 MC 提供 300 多种库函数。

为了避免程序书写的层次混乱不清,便于人们阅读、理解和查错,本书采用了一种使用较多的一种书写格式,这种书写格式并非是计算机要求的,而是为了给人们提供便利。如何正确书写程序,下面再给出一个具体的 C 语言程序。

例 1-3 统计输入文件中,行、单词和字符数量的程序。程序如下:

```

/* C1-3.C */
#include <stdio.h>

main()
{
    int c, n1, nw, nc, inword;
    inword=0;
    n1=nw=nc=0;
    while((c=getchar())!=EOF)
    {
        nc++;
        if(c=='\n')
            n1++;
        if(c==' '||c=='\t'||c=='\n')
            inword=0;
        else if(inword==0)
        {
            inword=1;
            nw++;
        }
    }
}

```

```

    }
}

printf("line = %d word = %d character = %d\n", nl, nw, nc);
}

```

上述程序书写格式要点如下。

(1) C 语言程序习惯上使用小写英文字母。C 语言程序中, 常量的宏定义和其他特殊用途的可用大写字母。

(2) C 语言程序不存在程序行概念。一行中可以有多个语句, 一个语句也可以占用任意多行, 但语句之间必须用“;”分隔。

(3) 不同结构层次的语句, 从不同的起始位置开始, 即在同一结构层次中的语句, 缩进同样的字符数。如程序例中 while 和 if、else 语句, 其结构中的各个语句都缩进相同位置。计算机输入 C 语言源程序时, 一般使用 TAB 键调整各行的起始位置。

(4) 表示结构层次的大括号, 写在该结构化语句第一个字母的下方, 与结构化语句对齐, 并单独占用一行。如 while 下方的“{”和倒数第三行的“}”是表示 while 结构范围的大括号对。同样, else 下方的大括号对也是如此。

(5) C 语言程序为了增强可读性, 可以使用适量的空格和空行。C 语言编译系统重视这样的空格和空行。但是, 变量名、函数名以及 C 语言本身使用的单词(如保留字 if、while、int 等), 不能在其中插入空格。

C 语言程序的函数模块结构和书写格式, 使得程序整体结构分明、层次清楚, 它为模块化软件设计方法提供有力的支持。

3 C 语言程序的开发过程

编写好一个 C 语言程序到完成运行的基本过程, 如图 1-1 所示, 它包含以下几个过程。

3.1 编辑

所谓编辑, 包括以下内容:

(1) 程序开发人员将自己编制的源程序文件输入给计算机内存;

(2) 修改源程序文件;

(3) 将修改好的源程序文件以文本文件形式保存在磁盘文件中, 源程序文件的名字由用户选定, 但要求有一定的后缀“.c”。例如:

f.c

目前用于建立源程序文件的编辑软件种类很多, 如 Unix 系统下的文本行编辑程序 ed, 或屏幕编辑程序 vi, MS-DOS 系统下的行编辑软件 Edlin 和屏幕编辑软件 Edit 或专用编辑软件 wps、Word 等。关于编辑软件的使用方法请参阅有关手册。

3.2 编译

编译的功能就是调用 C 语言编译系统对已编辑好的源程序文件(已存在磁盘文件中)翻译成二进制的目标代码。在对源程序文件进行编译前, 要再次进行检查和确认。在 Unix 系统下用 cat 命令, MS-DOS 系统下用 type 命令可以查看源程序文件的内容。发现问题后, 立即重新进入编辑状态, 对源程序文件进行修改后再重新编辑, 直到通过编译为止。

正确的源程序文件经过编译后得到的二进制代码(即生成目标程序文件)存储于文件系统

中。在 Unix 系统下,目标程序文件名与相应的源程序文件名同名,但将后缀自动改成“.o”,上述源程序文件 f.c 经编译后得到目标程序文件 f.o。在 MS-DOS 系统下,上述源程序文件 f.c 经编译后得到目标程序文件 f.obj。

编译过程是使用系统提供的编译程序进行的。不同操作系统下的各种编译程序的启动命令不同。

3.3 链接

编译后产生的目标程序文件还不能直接用于运行,因为每一个模块往往是单独编译的,必须把经过编译的各个模块的目标文件,以及系统提供的标准函数等连接后才能运行。链接也是使用系统提供的链接程序进行,生成的可执行文件也存储于文件系统中。在 Unix 系统下它以“.out”为后缀(例如, f.out),在 MS-DOS 系统下以“.exe”为后缀(例如, f.exe)。

可执行文件生成后,就可以在操作系统的支持下运行。若执行结果达到预期的目的,则开发工作到此结束,否则,应进一步检查修改源程序文件,再经过编辑—编译—链接的过程,直至取得正确的运行结果为止。

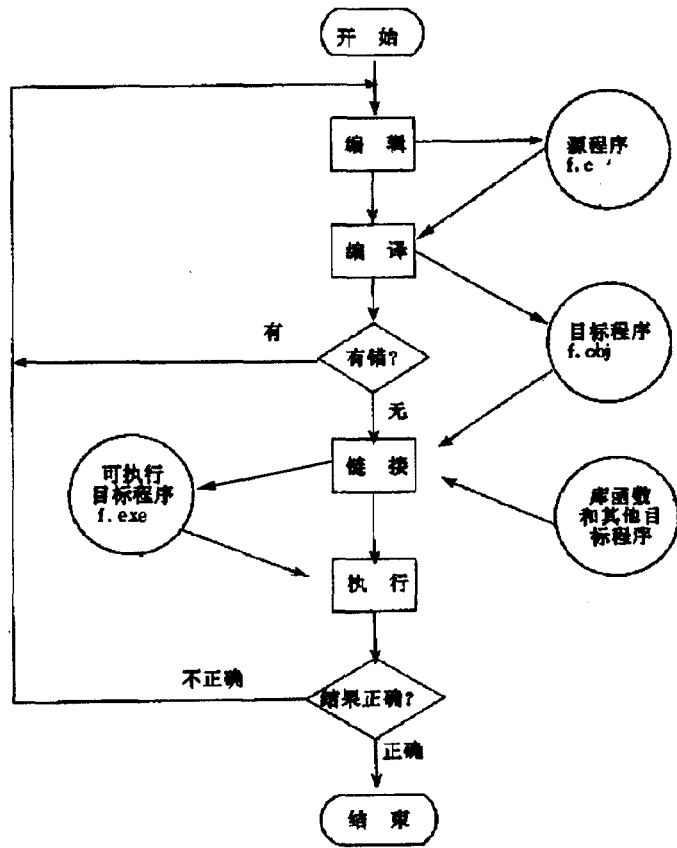


图 1-1 开发 C 语言程序过程流程图

3.4 Turbo C 2.0 的运行

近年来,出现了“集成化”的工具环境,将编辑、编译、链接、调试工具集于一身(例如 Turbo C),用户可以方便地在窗口状态连续进行编辑、编译、链接、调试、运行的全过程,详细内容将

在第九章 Turbo C 简介和错误分析中讨论。下面简要介绍 TC 的运行和操作方法。

3.4.1 启动集成开发环境

假设 Turbo C2.0 版的软件已经安装在 C:\TC 子目录下,在 C:\ 下装有启动 UCDOS 中文系统的批命令文件 UCDOS.BAT,则启用 Turbo C 的操作步骤如下。

(1)首先调入 UCDOS 中文系统,键入命令:

C:\>UCDOS

注:①如果程序不涉及汉字的输出,则可跳过本步骤;

②如果出现内存空间紧张情况,可考虑只调入 UCDOS 的一级字库使用。

(2)进入 TC 子目录,键入命令:

C:\>CD\TC

(3)运行 TC 集成环境,键入命令:

C:\TC>TC

此时出现如图 1-2 所示的菜单示意图。

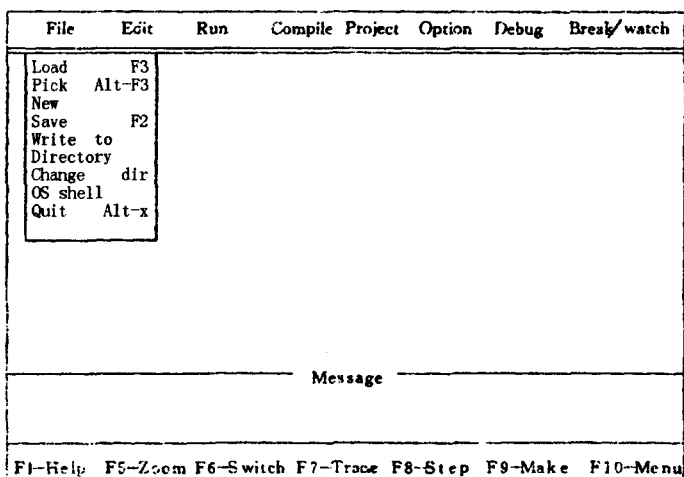


图 1-2 Turbo C 2.0 菜单示意图

3.4.2 用 Turbo C 开发 C 语言程序

(1)编辑 在主菜单下将光标移到 File/load 或 File/New 处,按回车键,屏幕出现一个小窗口,输入要编辑的程序名:

example.c ↵

若盘上无此文件,即建立输入一个新文件;若已有此文件,则将它调入并显示在屏幕上,系统转入编辑(Edit)状态。现经过编辑得到源程序如下:

```
/* example.c */
main()
{ printf("Welcome \n");
}
```

用 File/Save 或按 F2 键存盘。

(2)编译和连接 按 F9 键即进行编译、连接。由信息窗口显示诊断信息。若有错误,按任意键返回编辑窗口,光标停在出错处,修改后再按 F9 编译、连接。直到正确无误,立即生成可

执行程序文件:example.exe。

(3)执行 按 F10 键返回主菜单,将光标移到 Run/Run 处,按回车键,即运行刚编译好的可执行程序,得到结果:

Welcome

(4)退出 Turbo C 按 Alt-x 键,系统退出 Turbo C,返回到 DOS 状态。此时还可在 DOS 下执行上述开发的 C 语言程序:

C>example

立即得到结果:

Welcome

注:Turbo C 为了方便,提供热键 F9 用于编译、连接、执行全过程。可用 Alt + F5 查看程序输出结果(在 UC DOS 环境中,Alt + F5 用于设定五笔输入,此时应按右 Shift 键,屏蔽 UC DOS 组合键的使用)。

4 C 语言的基本语法单位

C 语言作为一种高级语言,除了规定一套严密的语法规则外,还必须规定它的基本语法单位,以便按照语法规则将它们构成语言的各种成分(如常量、变量、表达式、语句、说明等)。

C 语言的基本语法单位有以下几种。

4.1 C 语言字符集

各种程序设计语言都规定了允许使用的字符集,以便处理系统能正确识别它们。

C 字符集由 91 个字符组成,它们是:

小写字母 a b c ...z

大写字母 A B C ...Z

数字 0 1 2 ...9

特殊字符 + = - _ () * & % \$! | < > . , ; : " ' / ? { } ~ [] \

不可印出字符(空白符,包括空格、换行和制表符)。

4.2 标识符

一个标识符是一串由字母、数字和下横线“_”组成的字符串。标识符的第一个字符必须是字母或下横线符。多数 C 语言编译程序区分大写和小写字母。下面是一些标识符例子。

a b1 name _ car line 3 _ buf

下面是一些非法的标识符:

2a 不是以字母开头

char 与保留字同名

up. to 出现既非字母又非数字的字符

last name 空格不能出现在一个标识符中间

标识符是用来为常量、变量、用户定义的函数及类型取名的。用标识符时,一方面要取得有意义,便于阅读;另一方面要注意避免含义上或在书写时引起混淆。比如,字母 o 和数字 0,字母 l 与数字 1,字母 z 与数字 2 等都易于混淆,使用时要小心。

4.3 保留字

保留字是一类特殊的标识符,它们在 C 语言中是具有特定严格含义的一些单词。因此不

允许在 C 语言程序中另作它用。下面是 C 语言中通常的保留字：

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			

其中, signed 是一个运算符,别的主要用作类型标识符和基本控制结构标记,有关类型和基本控制结构在以后章节将详加说明。

下面几个字严格说来不属于保留字,但是建议读者把它们看成保留字而不要在程序中随意使用,以免造成混淆,这些字是:

define undef include ifdef ifndef endif line error elif pragma

它们通常用在 C 语言的预处理程序中。

4.4 运算符和分隔符

C 语言中有一些特殊符号用作运算符,C 语言运算符是比较丰富的,以后将有专门章节介绍。下面是 C 语言中主要的运算符:

() [] . - > ! ~ + + - - (类型) sizeof * / % + - < < > > < = > = =
= ! = < > & ^ | && || ? : = ,

其中,有些运算符在不同情况下有几种含义,这要依赖于上下文。比如“%”,在作为运算符时表示取余数(模),9%5 结果是 4;在作为打印输出语句时,又表示某种格式说明如“printf(“%d”,a)”意味着把变量 a 以十进制数(d)形式打印输出。又如“,”既可作为运算符,也可作一般的标点符号,等等。

从广义上说,C 语言中的运算符也都是分隔符,它们是分隔运算量的。但从狭义上说,C 语言的分隔符主要有:空格符、制表符、换行符和注释,统称空白格。

小 结

C 语言的丰富的数据类型、强有力的控制结构、灵活的表达式、高效率的代码、杰出的可移植性,使得它成为软件开发和文本处理中不可多得的工具。

C 语言程序完全由一系列函数组成,这些函数可以放在一个程序文件中,也可以放在多个程序文件里。这一系列函数必须有一个且只有一个名为 main()函数,这个函数称为主函数,整个程序从它开始执行,并且,一般来说,又在它里面结束。

C 语言程序的书写格式自由度较高,灵活性很强,有较大的随意性。但是,为了程序层次清楚,便于人们阅读、理解和查错,一般都采用有一定格式的习惯书写方法。这并非是计算机要求,而是给人们提供阅读、调试的方便。

C 语言程序的编辑、编译和运行与环境密切相关。本章主要以 Unix 和 MS-DOS 为其基本环境(Turbo C 将在第九章讨论),简要说明了 C 程序的编辑、编译和运行过程,这将为今后的实践打下一个基础。

C 语言的基本语法单位是:字符集、标识符、保留字、运算符和分隔符。

C 语言字符集包括英文字母、数字及一些符号。大多数 C 语言实现所采用的 C 字符集是