

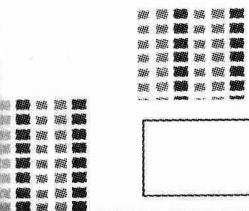
C 语言

程序设计

01 01 01 01 01 01 01 01 01

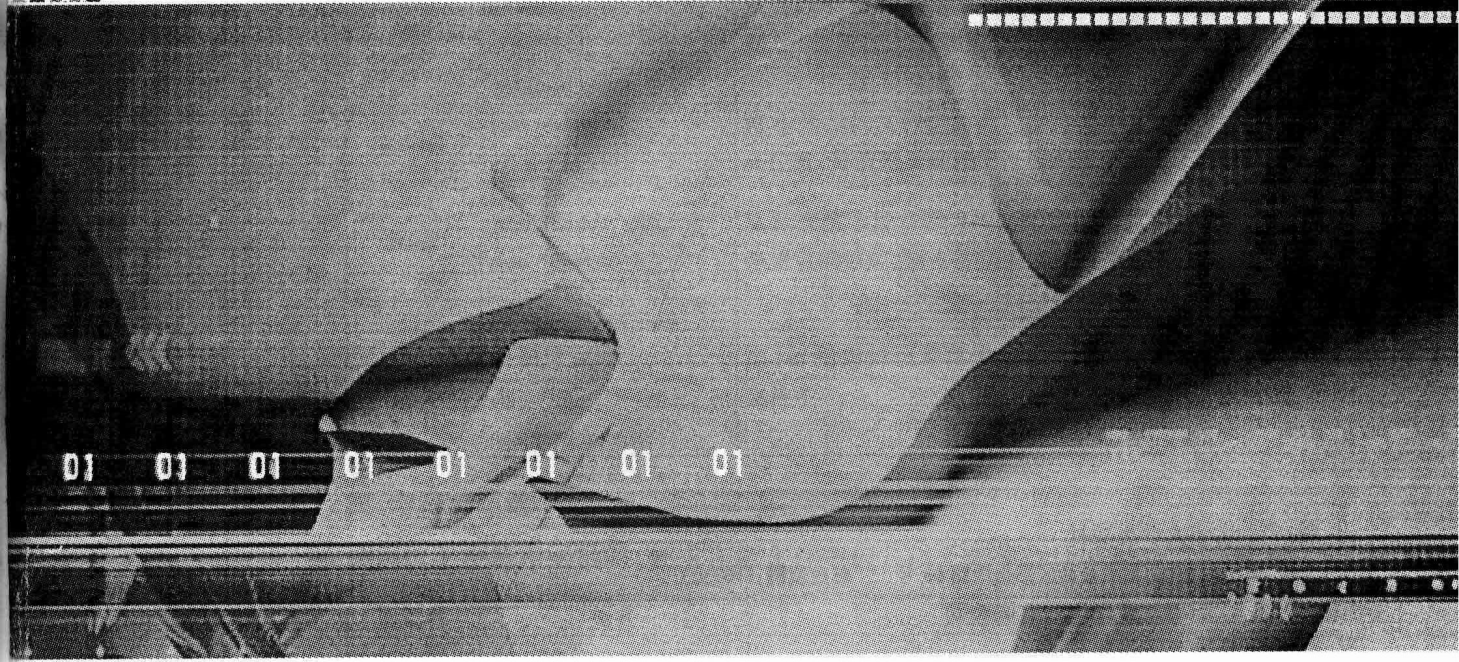
■ 张福祥 牛 莉 主编

辽宁大学出版社



C 语言

程 序 设 计



01 01 01 01 01 01 01 01

■ 主 编 张福祥 牛 莉
副主编 张维梅 黄同成 周红波
王 萌 李国慧

辽宁大学出版社

图书在版编目(CIP)数据

C 语言程序设计/张福祥,牛莉主编. — 沈阳:辽宁大学出版社,2010.1

ISBN 978-7-5610-6005-6

I. ①C… II. ①张…②牛… III. ①C 语言 — 程序设计 — 高等学校—教材 IV. ① TP312

中国版本图书馆 CIP 数据核字 (2010) 第 020385 号

出 版 者: 辽宁大学出版社

(地址:沈阳市皇姑区崇山中路 66 号 邮政编码:110036)

印 刷 者: 沈阳市北陵印刷厂有限公司

发 行 者: 辽宁大学出版社

幅面尺寸: 205mm × 280mm

印 张: 15.75

字 数: 465 千字

出版时间: 2010 年 1 月第 1 版

印刷时间: 2010 年 1 月第 1 次印刷

责任编辑: 董晋骞

封面设计: 韩雪霜

责任校对: 夏忠威

书 号: ISBN 978-7-5610-6005-6

定 价: 32.00 元

联系电话: 024—86864613

邮购热线: 024—86830665

网 址: <http://www.lnupshop.com>

电子邮件: lnupress@vip.163.com

前 言

C 语言在计算机程序设计领域应用广泛,具有功能丰富、语句简洁、使用方便、语法灵活等特点,是高等学校的一门核心基础课程。C 语言作为一种既适合编写系统软件,又适合开发应用软件的高级程序设计语言,已经成为各高校学生学习的主要程序设计语言之一。

本书内容共分 10 章。第 1 章为 C 语言概述,主要介绍程序设计语言的发展历史、C 语言的特点、C 程序的基本结构和 C 程序的上机步骤。第 2 章为数据类型与基本运算,主要介绍 C 语言的基本数据类型、常量、变量、运算符和表达式、基本输入输出函数。第 3 章为选择结构程序设计,主要介绍 C 语言的基本语句分类和选择结构。第 4 章为循环结构程序设计,主要介绍循环结构的几种语句以及与循环有关的控制语句。第 5 章为数组,主要介绍一维数组、二维数组和字符数组。第 6 章为函数与编译预处理,主要介绍函数定义及调用、变量和函数的作用域、编译预处理和条件编译。第 7 章为指针,主要介绍指针的概念、基本运算以及指针与数组和指针与函数的关系。第 8 章为结构体与共用体,主要介绍 C 语言构造类型的定义及应用。第 9 章为位运算,主要介绍 C 语言的各种位操作。第 10 章为文件,主要介绍 C 文件的各种操作。

本书按照 C 语言课程教学大纲的要求,内容安排遵循深入浅出、循序渐进的原则,全面系统地介绍了 C 语言基本语法和组成、程序设计的基本方法,着重培养学生的计算机编程能力。全书结构紧凑、概念准确、重点突出、例题丰富、叙述流畅、阐述问题思路清晰。书中精选了大量典型例题,对常用算法和学习中的重点、难点进行分析归纳,便于读者理解掌握。通过每章后面的习题,学生能随时自我检查学习效果并逐步提高阅读程序和编写程序的能力,为将来自学其他计算机语言和编制大型程序打下良好的基础。

本书采用现代风格定义和声明函数,使程序具有更强的易读性;采用流程图描述算法,所有程序都按结构化程序设计方法编写。本书中例题和习题的源代码程序均在 Turbo C 2.0 环境下通过上机调试。

本书由潍坊学院张福祥、张维梅、王萌、李国慧以及邵阳学院牛莉、黄同成、周红波等具有丰富教学经验的教师编写。张福祥编写了第一章,牛莉编写了第二章,张维梅编写了第三章、王萌编写了第四、五章,李国慧编写了第六、七章,黄同成编写了第八章,周红波编写了第九、十章。张福祥老师拟定编写提纲,统阅并修改了全书。

由于作者的水平有限、编写时间仓促,书中难免存在许多不足之处,恳请广大读者和同行批评指正。

编者
2010 年 1 月

目 录

第 1 章 C 语言概述	1	3.2 C 语言的三种基本结构	43
1.1 C 语言简介	1	3.3 选择结构程序设计	44
1.1.1 程序设计语言的发展	1	3.3.1 if 语句	44
1.1.2 C 语言的发展	2	3.3.2 switch 语句	50
1.1.3 C 语言的特点	2		
1.2 认识 C 程序	2	第 4 章 循环结构程序设计	58
1.3 C 语言程序的执行过程	5	4.1 while 语句	58
1.4 C 程序上机步骤	6	4.2 do-while 语句	59
1.4.1 Turbo C 2.0	6	4.3 for 语句	60
1.4.2 Visual C++6.0	8	4.4 循环的嵌套	62
		4.5 与循环有关的控制语句	63
第 2 章 数据类型与基本运算	12	4.5.1 break 语句	63
2.1 基本数据类型	12	4.5.2 continue 语句	64
2.2 标识符	13	4.6 程序举例	64
2.2.1 系统定义标识符	13	4.6.1 穷举算法	64
2.2.2 用户定义标识符	14	4.6.2 迭代算法	66
2.3 常量和变量	15		
2.3.1 常量	15	第 5 章 数 组	76
2.3.2 变量	19	5.1 一维数组	76
2.4 运算符与表达式	20	5.1.1 一维数组的定义	76
2.4.1 算术运算符和算术表达式	20	5.1.2 一维数组元素的引用	77
2.4.2 赋值运算符和赋值表达式	23	5.1.3 一维数组的初始化	78
2.4.3 关系运算符和关系表达式	24	5.1.4 一维数组应用举例	79
2.4.4 逻辑运算符和逻辑表达式	25	5.2 二维数组	84
2.4.5 条件运算符和条件表达式	27	5.2.1 二维数组的定义	84
2.4.6 逗号运算符和逗号表达式	28	5.2.2 二维数组元素的引用	84
2.4.7 其他运算符	28	5.2.3 二维数组的初始化	85
2.5 数据类型转换	29	5.2.4 二维数组应用举例	86
2.5.1 自动类型转换	29	5.3 字符数组与字符串	88
2.5.2 强制类型转换	30	5.3.1 字符数组	88
2.6 基本输入和输出	31	5.3.2 字符串	90
2.6.1 字符输入输出函数	31	5.3.3 字符串处理函数	92
2.6.2 格式输入输出函数	32	5.3.4 字符数组应用举例	95
2.7 应用举例	36		
第 3 章 选择结构程序设计	42	第 6 章 函数与编译预处理	99
3.1 C 程序的语句	42	6.1 函数定义	99

6.2 函数调用	100	8.3.3 结构体变量作函数参数	177
6.2.1 函数调用的格式	100	8.3.4 结构体指针作函数参数	178
6.2.2 函数调用的方式	101	8.4 动态数据结构——链表	179
6.2.3 函数的参数和返回值	103	8.4.1 链表的概念	179
6.2.4 被调函数的声明	106	8.4.2 链表的建立	179
6.2.5 函数的嵌套调用	107	8.4.3 链表的插入	181
6.2.6 函数的递归调用	108	8.4.4 链表的删除	183
6.3 变量的作用域与存储类别	110	8.5 共用体	185
6.3.1 局部变量及其作用域和存储类型	110	8.5.1 共用体类型的定义	186
6.3.2 全局变量及其作用域和存储类型	114	8.5.2 共用体变量的定义	186
6.4 编译预处理	116	8.5.3 共用体成员的引用	187
6.4.1 宏定义	117	8.6 枚举	189
6.4.2 文件包含	120	8.6.1 枚举类型和枚举型变量的定义	189
6.4.3 条件编译	121	8.6.2 枚举型变量的使用	191
第 7 章 指 针	134	第 9 章 位运算	197
7.1 指针和指针变量	134	9.1 基本位运算	197
7.1.1 指针和指针变量的概念	134	9.1.1 按位与	197
7.1.2 指针变量的定义与初始化	135	9.1.2 按位或	199
7.1.3 指针的基本运算	136	9.1.3 按位异或	200
7.2 指针与数组	139	9.1.4 按位取反	201
7.2.1 指针与一维数组	139	9.1.5 按位左移	202
7.2.2 指针与二维数组	142	9.1.6 按位右移	203
7.2.3 指针数组	144	9.1.7 位运算复合赋值运算符	204
7.3 指针与字符串	146	9.2 位运算应用	204
7.4 指针与函数	150	9.3 位段	204
7.4.1 指针变量作为函数参数	150	第 10 章 文 件	210
7.4.2 指针型函数	151	10.1 文件概述	210
7.4.3 指向函数的指针	153	10.1.1 文件的概念	210
7.5 指向指针的指针	156	10.1.2 文件的分类	210
7.6 指针应用举例	158	10.1.3 文件系统	211
第 8 章 结构体与共用体	167	10.1.4 文件类型指针	211
8.1 结构体	167	10.2 文件的操作	212
8.1.1 结构体类型的定义	167	10.2.1 文件的打开和关闭	212
8.1.2 结构体类型变量的定义和初始化	168	10.2.2 文件的读写	214
8.1.3 结构体成员的引用	170	10.2.3 文件的定位	223
8.2 结构体数组	172	10.2.3 文件的检测	226
8.2.1 结构体数组的定义	172	10.3 常见错误分析	227
8.2.2 结构体数组的初始化	172	10.4 文件应用	228
8.2.3 结构体数组的应用	173	附 录 A 常用字符与 ASCII 代码对照表 ...	240
8.3 结构体与指针	175	附 录 B 运算符优先级和结合性	241
8.3.1 结构体变量与指针	175	附 录 C 库函数	242
8.3.2 结构体数组与指针	176		

第 1 章 C 语言概述

程序设计语言是人与计算机之间交换信息的工具,是为描述计算过程而设计的一种具有语法规则描述的特定符号集合。

C 语言是一种结构化程序设计语言。它具有先进的程序控制结构,有丰富的数据类型和运算,在事务处理、自动控制、科学研究等各领域的应用软件开发中,取得了巨大的成功。

1.1 C 语言简介

1.1.1 程序设计语言的发展

人们利用计算机解决实际问题,一般要编写程序。从世界上第一台计算机诞生到今天,伴随着计算机技术的发展,程序设计语言也在不断地升级换代。

程序设计语言一般分为机器语言、汇编语言和高级语言三类。

1. 机器语言

机器语言是最底层的计算机语言。计算机硬件系统只能执行由 0、1 二进制代码构成的操作指令。在指令代码中一般包括操作码和地址码,其中操作码告诉计算机做何种操作,地址码则指出被操作的对象。例如代码 10000000 表示加法操作,而代码 10010000 表示减法操作。

机器语言是设计计算机硬件时所确定的,对于不同的计算机硬件(主要是 CPU),其机器语言是不同的。由于机器语言程序是直接针对计算机硬件的,因此它的执行效率比较高。

机器语言与人们的自然语言和数学语言有很大的差别,它比较难记、难认、难理解、容易出错,而且程序的直观性比较差,也不容易移植。这样就给计算机的应用推广带来很大的不便。

2. 汇编语言

汇编语言又称符号语言。汇编语言用简单、易记忆的助记符来取代机器语言中的操作码,用十进制或十六进制数取代机器语言中的操作数,对机器指令进行简单的符号化。例如 ADD 表示加法运算操作码, SUB 表示减法运算操作码。用汇编语言编写的程序称为汇编语言源程序。

由于汇编语言采用了助记符,因此,它比机器语言直观,容易理解和记忆。用汇编语言编写的程序也比机器语言程序易读、易检查、易修改。但是,计算机不能直接执行这种源程序,必须经过汇编程序将其汇编形成目标程序,并连接生成可执行文件,才能执行。不同种类的计算机都随机带有不同的汇编程序。

汇编语言与机器语言,都是依 CPU 的不同而异,都称为面向机器的语言。这种语言编程时要依赖于具体的计算机,因此汇编语言也不具有通用性。

3. 高级语言

20 世纪 50 年代后期至 60 年代中后期,出现了高级语言和操作系统。高级语言是一种接近于自然语言和数学语言的计算机语言,如 C 语言、BASIC 语言、FORTRAN 语言等。

高级语言摆脱了对计算机硬件的依赖,编程时不需要考虑机器的内部结构和存储单元的分配。一个语句相当于许多条计算机指令,描述方式也与我们日常处理问题的逻辑思维习惯相接近,因此易于被人们接受和掌握。用高级语言编写程序要比低级语言容易得多,并大大简化了程序的编制和调试,使编程效率得到大幅度地提高。高级语言的显著特点是具有良好的通用性和可移植性。

用高级语言编写的程序叫做高级语言源程序。计算机也不能直接识别和执行这种程序,必须通过翻译,才能将其转换成机器语言程序执行。翻译的方法有两种:一种是编译,另一种是解释。

1.1.2 C 语言的发展

C 语言是在 20 世纪 70 年代初产生的。最初版本的 C 语言是 1972 年由美国贝尔实验室的丹尼斯·里奇设计,在 PDP-11 机器上实现的一种小型语言。后来经过了多次改进,直到 1978 年,贝尔实验室的布莱恩·W.科宁汉和丹尼斯·里奇合著了著名的《C 程序语言》(《The C Programming Language》)一书。从此,C 语言开始作为一种通用的计算机语言得到广泛应用。

1.1.3 C 语言的特点

C 语言是一种通用的结构化语言。它之所以得到迅速普及,并表现出顽强的生命力,是因为具有以下主要特点。

(1) C 语言简洁、使用灵活、方便,共有 32 个关键字,10 种基本语句,压缩了一切不必要的成分,便于阅读和书写。

(2) C 语言具有十分丰富的数据类型,不仅有基本数据类型,还具有构造类型,以及指针类型等丰富的数据类型。可用来实现各种复杂的数据结构。因此 C 语言具有很强的数据处理能力。

(3) C 语言提供了丰富的运算符,共有 24 种运算符。除算术运算、逻辑运算外,还可以实现位运算、指针运算等各种运算和控制操作。因而,C 语言的运算多样化并具有较强的数据处理能力。不仅能完成一般高级语言的运算功能,而且能实现许多汇编语言的功能。

(4) C 语言程序设计模块化、结构化。C 语言是以函数为模块来组成程序的,函数实现了程序的模块化。C 语言提供了顺序、选择和循环控制结构,从而实现了程序的结构化。

(5) C 语言允许直接访问物理地址,能进行位(bit)一级的操作,能实现汇编语言的大部分功能,可以直接对硬件进行编程操作。因此,C 语言既具有高级语言的功能,又具有低级语言的功能,可用来写系统软件。

(6) C 语言生成目标代码质量高,程序执行效率高。在代码效率方面可以和汇编语言相媲美。

1.2 认识 C 程序

本节通过列举几个简单的程序,介绍 C 语言程序的一些基本构成和格式。

【例 1-1】 输出字符串:I want to study English.

【程序源代码】

```
#include "stdio.h"
main( )
{
printf ("I want to study English. \n");
}
```

【程序运行结果】

I want to study English.

【程序分析】

该程序由一个 main 主函数构成,花括号“{ }”内是函数体,一般由若干条语句组成,本函数中只有一条语句,该语句的功能是输出一个字符串。

【例 1-2】 求整数 3 和 1 的和。

【程序源代码】

```
#include "stdio.h"
main( )
{
int x,y,z;    /* 变量 x 和 y 分别存放 3 和 1,变量 z 存放和 */
x=3;
y=1;
z=x+y;
printf( " %d+%d=%d\n",x,y,z);
}
```

【程序运行结果】

3+1=4

【程序分析】

该程序包含一条变量说明语句,4 条执行语句。语句的功能是:定义 3 个变量 x,y,z;给变量 x,y 分别赋值 3,1;求出变量 x,y 的和,赋给变量 z;最后输出结果。

【例 1-3】 从键盘输入任意两个整数,输出两数和。

【程序源代码】

```
#include "stdio.h"
main( )
{
int x,y,z;
scanf("%d,%d",&x,&y); /* 变量 x 和 y 分别存放输入的两个数 */
z=x+y;
printf("%d+%d=%d\n",x,y,z);
}
```

【程序运行结果】

从键盘上输入:10,20

10+20=30

【程序分析】

该程序是将【例 1-2】中的两条赋值语句,换成一条调用标准输入库函数语句,分别从键盘输入两个整数赋值给变量 x 和 y 。

说明:

1.C 程序的组成

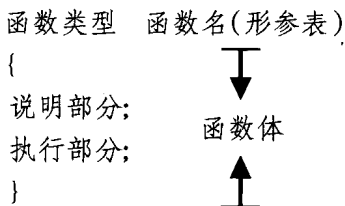
每个 C 语言程序都由一个或多个函数组成,每个函数完成相对独立的功能。以上例题中的程序都是由一个称为 `main` 的函数构成的。`main` 是函数名,系统将通过这个名称调用该函数。每个完整的程序至少包含 1 个 `main` 主函数。

程序总是从主函数 `main` 开始执行。

2.函数及其结构

C 程序的基本单位是函数。一个函数由函数的首部和函数体两部分组成。函数的首部包括函数名、函数类型、函数参数(形参)名、形式参数类型的说明;函数体一般包括说明部分和执行部分。

函数结构如下:



`void` 是 `main` 函数的类型符,该类型符可以省略。`main` 函数的形参表是空的,没有任何参数,但是函数名后的圆括号“`()`”不能省略,因为它是识别函数的重要标志。

函数体以左花括号“`{`”开始,以右花括号“`}`”结束,其间可以有说明部分和执行部分。如【例 1-2】的程序中第 4 行就是程序的说明部分,第 5 行到第 8 行是程序的执行部分。在 C 语言程序中,字母的大小写是有区分的。因此 `main`, `Main`, `MAIN` 等都是不同的名称,作为主函数的函数名的只能是 `main`,不能混同。

3.基本语法规定

(1)C 语言程序在书写格式上是比较自由的,一行中可以书写多个语句,一个语句也可以分布在连续的若干行中。

(2)每条语句后必须以分号“`;`”作为结束标记。

(3)程序注释以“`/*`”开始,以“`*/`”结束。“`/*`”和“`*/`”必须成对出现,“`/`”和“`*`”之间不允许有空格。注释可以用西文,也可以用中文。注释的长度可以是一行或者几行。注释可以出现在程序中任意合适的地方,它对程序的运行不起作用。注释可以用来说明程序段的功能,以便帮助人们阅读程序。因此一个好的程序应该有必要的注释。

(4)C 语言中应严格区分字母的大小写。

4. scanf、printf 函数

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。`scanf`、`printf` 函数是 C 语言的库函数中的标准输入输出函数,不需定义,可直接调用。其中 `printf` 函数中的“`\n`”是 C 语言中规定的一个特殊符号,作为控制代码,作用是“回车换行”。

5.编译预处理命令 #include

程序的第 1 行:`#include "stdio.h"` 是一个编译预处理命令。编译预处理命令必须以“`#`”开头,末尾不能加“`;`”。

该编译预处理命令称为文件包含命令,功能是把一个文本文件的内容包含到该命令处。“`stdio.h`”是

命令的参数,它给出了所包含文件的文件名,即 `stdio.h`。`stdio.h` 是系统提供的头文件名,该文件中包含着有关输入输出函数的信息。

C 语言规定在调用库函数时往往需要用 `#include` 命令将有关的头文件调到所用函数的前面。在【例 1-3】中,由于调用标准输入输出函数 `scanf`、`printf`,所以需要在程序的开始处,将头文件 `stdio.h` 包含进来。

1.3 C 语言程序的执行过程

C 语言程序是一种编译型的高级语言。用 C 语言编写的源程序必须先进行编译和连接,生成可执行程序之后才能执行。执行一个 C 语言程序一般经过如下步骤。

1. 编辑

通过编辑得到的程序称为源程序。源程序以纯文本格式保存在源程序文件(简称为程序文件)中。编辑的工作内容就是输入、修改源程序并将修改好的源程序保存在磁盘文件中。源程序文件的主文件名由编程者自己确定,文件的扩展名必须是“.c”。

2. 编译

编译是对程序文件中的源程序进行语法检查,并翻译成可重定位的二进制目标程序,保存在目标程序文件中。编译生成一个与源文件主文件名相同,但扩展名为“.obj”的目标文件。

3. 连接

经过编译产生的目标文件还不能直接执行,必须使用系统提供的连接程序将编译过的目标文件和程序中用到的标准库函数进行连接,生成一个与源程序文件主文件名相同,扩展名为“.exe”的可执行文件。

4. 运行

在操作系统平台可以直接执行扩展名为“.exe”的可执行文件。

如果运行后没有得到预定的结果,说明程序存在错误,此时必须重复前面的步骤,修改程序、重新编译连接,直到得出正确的运行结果。

由此可见,C 语言程序的开发是一个由编辑开始,经过编译、连接和运行又回到编辑的反复循环的过程。该过程如图 1-1 所示。

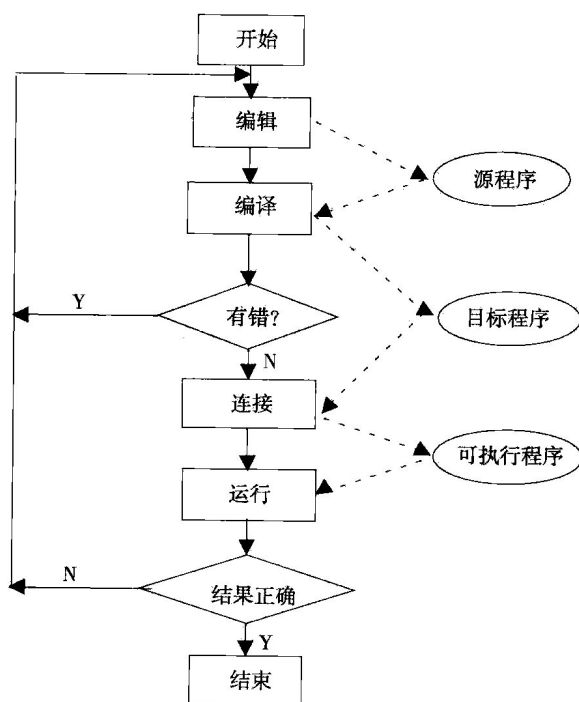


图 1-1 C 程序开发过程示意图

1.4 C 程序上机步骤

1.4.1 Turbo C 2.0

Turbo C 2.0 是 Borland 公司推出的。它为用户创造了编辑、编译、调试、跟踪、存储、调用文件等多功能的集成开发环境(IDE)。

1. Turbo C 2.0 的启动

打开 TC 文件夹,用鼠标双击 TC.EXE 文件,即可启动 Turbo C 系统。进入 Turbo C 运行主界面后,屏幕将出现如图 1-2 的画面。

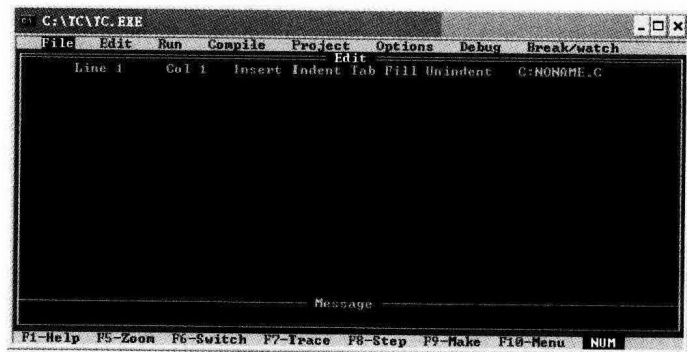


图 1-2 主屏幕

图 1-2 由三部分组成:顶部的主菜单区,中间的编辑区,底部的功能键和热键说明区。

(1)编辑窗口上方的编辑信息行。

Line n 行显示, n 是行数,随光标位置变化而变化。

Col n 列显示, n 是列数,随光标位置变化而变化。

Insert 插入模式开关,用 Ins 键进行模式开关的切换。Insert 表示当前是插入状态,这时所键入的字符将出现在光标之前;Insert 消失表示当前是改写状态,这时键入的字符将替换当前的光标所在处的字符。

Indent 自动缩进开关,用 Ctrl+OI 键(按住 Ctrl 键不放,再键入后面的字母)切换。

Tab 制表开关,用 Ctrl+OT 键切换。

Fill 它与 Indent 和 Tab 的开关(ON/OFF)一起使用。当 Tab 模式为“ON”时,编辑系统将用制表及空格符优化每一行的开始。

(2)功能键提示。

在屏幕底部的一行是功能键提示行,其作用如下:

F1:Help。按 F1 键,将在编辑窗口内打开一个“帮助”窗口,可在此查看各种编辑命令,按 PageDown 键或 PageUp 键可翻页,按 Esc 键退出此窗口。

F5:Zoom。在编辑状态,屏幕上通常有两个窗口,上面是输入源程序的编辑窗口,下面是系统给出信息的信息窗口(此窗口中间标有 Message),当光标或光条出现在哪个窗口,就称该窗口是“活动”窗口。按 F5 键,使活动窗口放大到整个屏幕;再按 F5 键,使屏幕还原。

F6:Switch。按 F6 键,可使光标在编辑窗口和信息窗口之间进行切换。

F7:Trace。按 F7 键,逐行执行程序,跟踪函数调用。

F8:Step。按 F8 键,逐行执行程序,不跟踪函数调用。

F9:Make。按 F9 键,编译并连接编辑窗口内的程序。

F10:Menu。按 F10 键,返回主菜单。

2.文件的编辑、保存和运行

当启动 Turbo C 时,主菜单总是指向 File 菜单项,按(ESC)键就可进入编辑窗口。

(1)新建文件。

按下 Alt+F 键打开 File 菜单,选择 New 命令项,开始编辑一个新的源程序。程序输入完后,按 F2 键保存文件,屏幕上将出现图 1-3 所示的提示信息。这时用户可以修改该文件名,也可以修改保存该文件的路径。

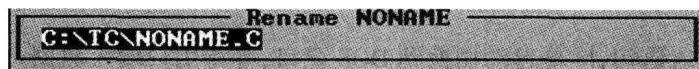


图 1-3 保存文件的提示信息

(2)编辑一个已存在的文件。

选择并执行 File 菜单中的 Load 命令项(或按快捷键 F3),屏幕上出现如图 1-4 所示的画面,用户输入文件名,就可装入该文件。

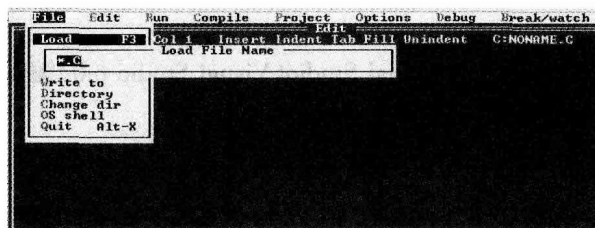


图 1-4 装入文件时的画面

(3)如何保存源程序文件。

无论是新键入的 C 源程序或是进行修改的旧程序,都先不要急于运行。应该先把这个程序以文件的形式保存到磁盘上。养成随时保存自己程序的习惯,以免因为某些预料不到的偶发事件而导致辛苦键入的源程序丢失。

如果一个进行修改的旧的 C 源程序(已有确定的名字),仍用旧的名字保存,按 F2 键即可,或者在 File 的下拉菜单中选 Save。

如果一个修改完的 C 源程序想用另外一个文件名来保存,存盘步骤如下:

- ①按下 Alt+F 键,打开 File 菜单;
- ②选择并执行 Write To 命令项;
- ③在一个标有 New Name 的小框中,键入用户的新文件名。

(4)如何退出 Turbo C。

退出 Turbo C 之前,应检查当前的 C 程序是否已存盘。可用以下两种方法退出 Turbo C:

- ①在 File 下拉菜单中选最后项 Quit,按 Enter 键,立刻退出 Turbo C;
- ②直接按 Alt+X 键,立刻退出 Turbo C。

(5)如何编译程序。

当在编辑窗口内键入程序,存盘后,就可以编译程序,检查是否有语法错误。可以用以下两种方法编译 C 程序。

当文件建立或修改完毕并存盘后,即可进行编译。具体方法有两种:

①选择 Compile 菜单中的 Compile to OBJ 进行编译,得到一个扩展名为“.obj”的目标程序;然后再选择 Compile 菜单中的 Link exe file 选项生成一个扩展名为“.exe”的可执行文件。

②也可以将编译和连接合为一步进行,直接在 Compile 菜单中选 Make exe file 选项,生成扩展名为“.exe”的可执行文件。

在编译过程中,屏幕的中间出现一个方框,用以显示编译处理时的信息。若编译没有错误,方框底部闪现 success 信息,按任意键后返回编辑窗口;若编译出错,在信息窗口中将显示错误信息。按任一键后,光标将停在编辑窗口内的出错行上(注意:出错信息并不完全准确,需要根据上、下文进行分析)。这时应改正错误,保存文件后重新进行编译。

(6)如何运行程序。

确认编译没有错误后,就可运行程序了。方法是直接按 Ctrl+F9 键或在 Run 菜单中选 Run 命令,程序便开始执行。

如果程序中有输入语句,屏幕自动切换到 DOS 窗口,等待用户输入数据。一旦程序运行完毕将返回编辑窗口。若要查看结果,应在 Run 菜单中选择 User screen 或按 Alt+F5。看完结果后按任意键又回到编辑窗口。

若要继续输入新的程序,可选 File 菜单中的 New 选项清除编辑窗口中的原有内容。

1.4.2 Visual C++ 6.0

Visual C++ 6.0 是微软公司推出的 Visual Studio(Visual Studio 6.0)编辑器中的一个。C 语言程序可以在 Visual C++ 6.0 环境下进行编译、连接和运行。

1. Visual C++ 6.0 的启动

从“开始”菜单中选择“程序”,然后选择“Microsoft Visual Studio”中的“Visual C++ 6.0”,即可进入 Visual C++ 6.0,主窗口如图 1-5 所示。

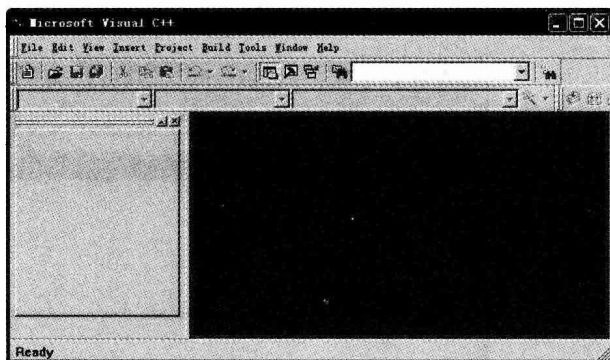


图 1-5 主窗口

2. 新建和编辑源程序

在主窗口的菜单栏中单击 File 菜单,选择 New,将弹出“New”对话框,如图 1-6。

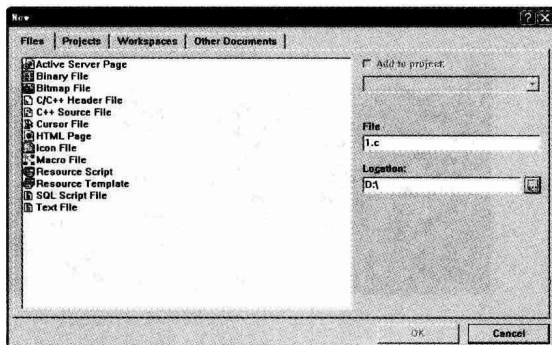


图 1-6 “New”对话框

选择 Files 选项卡。单击 C++ Source File 选项,在 Files 文本框中输入新建的 C 源程序文件名(例如输入 1.c,注意必须加上后缀名.c),在 Location 文本框中输入新建的 C 源程序文件的存取路径(例如输入 D:\),单击 OK 按钮。系统自动返回 Visual C++6.0 主窗口,如图 1-7。

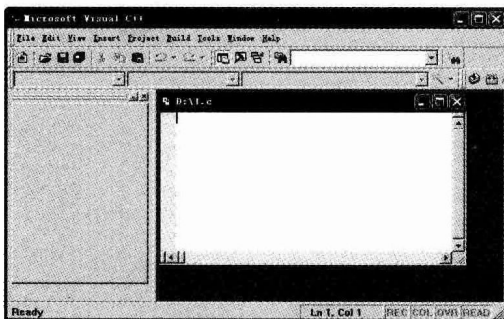


图 1-7 编辑窗口

光标在程序编辑窗口闪烁。在程序编辑窗口,可以输入和编辑源程序。经检查程序无误后,在 Files 菜单中选择 Save,保存文件。

3. 编译和连接

选择主窗口菜单栏中 Build 菜单项;系统弹出下拉菜单,选择 Build 命令可以一次完成编译和连接。编译连接过程中,系统如发现程序有语法错误,则在输出区窗口中显示错误信息,给出错误的性质、出现位置和错误的原因等。如果双击某条错误,编辑区窗口右侧会出现一个箭头,指示出现错误的程序行。用

户据此对源程序进行相应的修改,并重新编译和连接,直到通过为止,如图 1-8。

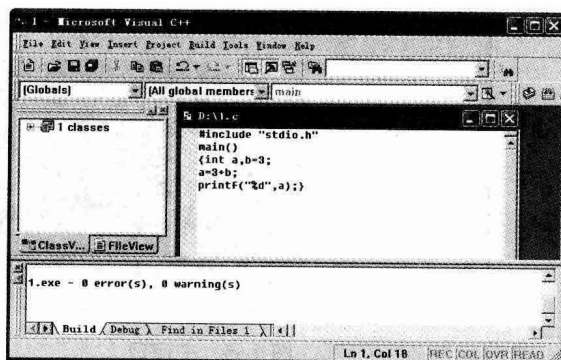


图 1-8 编译连接窗口

4. 运行

选择 Build 菜单中的! Execute 1.exe 命令,就可以执行程序,输出结果窗口如图 1-9。

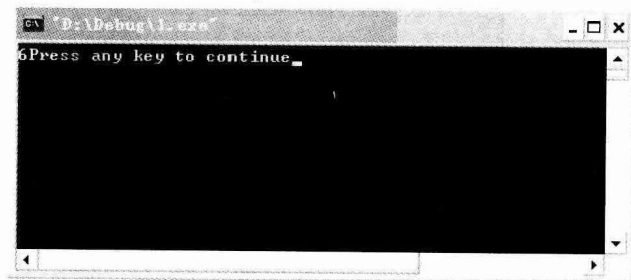


图 1-9 运行结果

在输出结果窗口中,按下任一键后,将返回主窗口。

习 题

一、选择题

- 以下叙述中正确的是()。
 - C 语言比其他语言高级
 - C 语言源程序可以不用编译就能被计算机识别执行
 - C 语言以接近英语国家的自然语言和数学语言作为语言的表达形式
 - C 语言出现的最晚,具有其他语言的一切优点
- 构成 C 语言源程序的基本单位是()。
 - 子程序
 - 过程
 - 文本
 - 函数
- 在一个 C 程序中,main()函数()。
 - 必须出现在所有函数之前

- B) 可以在任何地方出现
C) 必须出现在所有函数之后
D) 以上都不对
4. 下列叙述中正确的是()。
- A) C 程序的执行是从 main() 函数开始, 到本程序的 main() 函数结束
B) 注释语句是必不可少的
C) main() 函数必须放在其他函数之前
D) printf() 是系统提供的输出函数
5. 以下描述正确的是()。
- A) 最简单的 C 源程序可以不包括任何内容
B) 编译程序对 C 源程序编译时不检查语法
C) C 程序中没有输入输出语句
D) 在 C 程序中, 至少必须有一个 main() 函数
6. 下列语句或命令中, 不符合 C 语言程序书写规则的是()。
- A) int m; B) int m,n;
C) m=10; D) #include "stdio.h"
7. 以下叙述中正确的是()。
- A) C 语言的源程序不必通过编译就可以直接运行
B) C 语言中的每条可执行语句最终都将被转换成二进制的机器指令
C) C 源程序经编译形成的二进制代码可以直接运行
D) C 语言中的函数不可以单独进行编译
8. 以下说法正确的是()。
- A) C 程序总是从第一个函数开始执行
B) void 是 main() 函数的类型符
C) C 程序调用的函数在 main() 函数中定义
D) C 程序中的 main() 函数必须放在程序的开始部分

二、填空题

1. C 语言是一种 _____ 化程序设计语言。
2. C 语言源程序文件的后缀是 _____, 经过编译后, 生成文件的后缀是 _____, 经过连接后, 生成文件的后缀是 _____。
3. C 程序中, 语句必须以 _____ 作为结束标记。
4. 一个 C 语言程序是从 _____ 函数开始执行的。

三、问答题

1. C 语言具有哪些主要特点?
2. 简述 C 语言程序的结构特点。
3. 简述开发一个 C 语言程序的步骤。