C语言程序设计

C yuyanchengxusheji

刘浩 杜忠友 主编

山东大学出版社

内容提要

本书主要内容包括：C 语言概论、数据类型、运算符与表达式、顺序结构、选择结构、循环结构、函数、数组、指针、字符串、对函数的进一步讨论、结构体和共用体、位运算、文件等。

本书给出了典型的例题和一定数量的习题，便于读者领会、理解和掌握所学的语法规则及程序设计方法与技巧。

本书可作为大专院校计算机专业、非计算机专业和计算机培训班的教材，也可供工程技术人员和自学者使用。

前　言

根据上级有关文件的指示精神，高校各专业教学计划总学时数减少，各门课程的学时数也相应减少。我院对《C 语言程序设计》课程的教学时数作了较大的压缩，目的是要培养学生的自学能力，提高学生的自身素质。为此，我们结合我院当前的实际情况，根据该课程的教学大纲，在以前讲义的基础上，编写了《C 语言程序设计》这本书。在使用本书时可根据实际教学时数对教学内容作适当增减。

这本书的显著特点是内容精炼、提纲挈领、简明扼要，既把问题讲清楚，又不纠缠于细节中，使得学习起来思路清晰，概念清楚，容易理解和掌握。

本书对章节顺序作了合理的安排，做到先易后难，循序渐进。并对各部分做了通俗易懂的讲解，易于教学，易于自学，读者能够在这种通俗易懂中很快进入角色，进而对本书、本课程产生兴趣。

本书给出了典型的例题和一定数量的习题，便于读者领会、理解和掌握所学的语法规则及程序设计方法与技巧。

本书共分 13 章，内容包括：C 语言概述、数据类型、运算符与表达式、顺序结构、选择结构、循环结构、函数、数组、指针、字符串、对函数的进一步讨论、结构体和共用体、位运算、文件等。

本书由山东建筑工程学院计算机系的刘浩、杜忠友副教授任主编。其中，第一章由王晓阆编写，第二章由刘秀婷编写，第三章、第四章由夏传良编写，第五章由崔巍编写，第六章由刘浩编写，第七章由杜忠友编写，第八章由霍涛编写，第九章由赵秀梅编写，第十章、第十二由姜庆娜编写，第十一章由张冬梅编写，第十三章由赵欣编写，附录由陈兆柱编写。山东建筑材料工业学院信息与控制系王孝红教授审阅了全书，并提出了宝贵意见；在本书的编写和出版过程中还得到了各位领导的大力支持，在此，一并表示衷心地感谢！

由于编者水平所限，加之时间仓促，缺点和疏漏之处在所难免。恳请有关专家和广大读者不吝赐教，以使本书更加完美。

编　者

2000 年 9 月

目　录

第1章 C语言概述

为使读者对C语言有一个概括的了解，在详细介绍C语言程序设计之前，本章简单地介绍C语言的产生过程及特点、C语言程序的结构与书写格式以及C语言程序的开发过程。

1.1 C语言的产生过程及特点

1.1.1 C语言的产生过程

60年代，随着计算机科学的迅速发展，高级程序设计语言得到了广泛的应用，然而，还没有一种可以用于书写操作系统和编译程序等系统程序的高级语言，人们不得不用汇编语言（或机器语言）来书写，但汇编语言存在着不可移植、可读性差、研制软件效率不如高级语言等缺点，给编程带来很多不便。为此，人们对能用于系统程序设计的高级语言的开发就变得势在必行了。

1967年，首先由Martin Richards开发出BCPL语言（Basic Combined Programming Language）。作为软件人员开发系统软件的描述语言，BCPL语言的突出特点是：

（1）结构化的程序设计；

（2）直接处理与机器本身数据类型相近的数据；

（3）具有与内存地址对应的指针处理方式。

1970年，Ken Thompson继承并发展了BCPL语言的上述特点，设计并实现了B语言。当时，美国DEC公司的PDP-7小型机UNIX操作系统，就是使用B语言开发的。

1972年，美国Bell实验室在PDP-11小型机的UNIX操作系统开发中，Dennis M. Ritchie和Brian W. Kernighan对B语言做了进一步的充实和完善，正式地推出了C语言。

1.1.2 C语言的特点

C语言把以往高级语言的特征同汇编语言的能力结合起来形成所谓的中级语言，因此，C语言具有许多独到的特点，以下仅从使用者的角度加以讨论：

（1）C语言短小精悍，基本组成部分紧凑、简洁。一共有32个标准的关键字、45个标准的运算符以及9种控制语句，语句的组成精炼、简洁，使用方便、灵活。

（2）C语言运算符丰富，表达能力强。C语言具有高级语言和低级语言的双重特点，其运算符包含的内容广泛，所生成的表达式简练、灵活，有利于提高编译效率和目标代码

的质量。

(3) C 语言数据结构丰富、结构化好。C 语言提供了编写结构化程序所需要的各种数据结构和控制结构，这些丰富的数据结构和控制结构以及以函数调用为主的程序设计风格，保证了利用 C 语言所编写的程序能够具有良好的结构化。

(4) C 语言提供了某些接近汇编语言的功能，为编写系统软件提供了方便条件。如它可以直接访问物理地址，并能进行二进制位运算等。

(5) C 语言程序所生成的目标代码的效率仅比用汇编语言描述同一个问题低 20%左右，因此，用 C 语言编写的程序执行效率高。

(6) C 语言程序可移植性好。在 C 语言所提供的语句中，没有直接依赖于硬件的语句。与硬件有关的操作，如数据的输入、输出等都是通过调用系统提供的库函数来实现的，而这些库函数本身并不是 C 语言的组成部分。因此，用 C 语言编写的程序能够很容易地从一种计算机环境移植到另一种计算机环境中。

当然，C 语言本身也有弱点：一是运算符的优先级较多，不容易记忆；二是由于 C 语言的语法限制不太严格，这在增强了程序设计的灵活性的同时，在一定程度上也降低了某些安全性，这对程序设计人员提出了更高的要求。

1.2 C 语言程序的结构与书写格式

1.2.1 C 语言程序的结构

C 语言程序是由一个或多个函数组成，但任何一个完整的 C 语言程序，都必须包含一个且只有一个名为 main()的函数，它是程序运行时执行调用的第一个函数。

C 语言程序的一般形式如下，其中 f1()到 fn()代表用户自定义的函数：

```
全程变量说明
main()
{
    语句序列;
}
类型 f1(参数)
{
    语句序列;
}
类型 f2(参数)
{
语句序列;
}
   ⋮
类型 fn(参数)
{
    语句序列;
}
```

1.2.2 C语言程序的书写格式

1. 用C语言书写程序时较为自由，既可以一行写多个语句也可以一个语句分几行来写。每个语句用“;”结尾。

2. C语言程序要求关键字都使用小写字母。在C语言中小写字母和大写字母是不同的，例如，小写的while是关键字，大写的WHILE则不是。关键字在C语言中不能用作其他目的，不能用作变量或函数的名字。

3. C语言程序是结构化程序设计语言，为了层次结构分明，书写程序时不同层次结构层次的语句，从不同的起始位置开始，同一层次中的语句缩进对齐。

4. 为了增加可读性，一般一个语句占一行，并适当加一些注释或空行。请看下面C语言程序书写格式示例。

例 1-1 从键盘输入两个整数，并将这两个整数之和显示出来。

```
#include<stdio.h>
mian()
{
  int a,b,sum;
  scanf("%d%d",&a,&b); /* 读入两个整数，存入变量 a 和 b 中 */
  sum=a+b;
  printf("The sum of %d and %d is %d",a,b,sum); /* 输出两数之和 sum */
}
```

此例中，只有一个函数，即主函数main()。a,b,sum都是变量。int a,b,sum语句是说明语句，说明它们都是整数型的变量。后三个语句是执行语句，其中scanf()是输入库函数，它要求从键盘上输入两个整数(以空格为分隔符)，并将它们送到a,b变量中。它要求将a,b用其地址&a和&b表示，双引号中的两个%d是十进制整数格式符，指明对变量a,b的格式要求。printf()是输出库函数，它的作用是进行格式化输出，其参数包括两部分：一是双引号内的“格式控制”部分，用于对输出的数据进行格式说明；二是“输出表列”部分，表示要输出的变量。所有这些语句都放在大括号{}之内，各语句之间用分号“;”隔开。程序的开头的#include语句是编译预处理语句，其作用是将“<>”中的文件stdio.h内容读入该语句的位置处。在使用C语言输入输出库函数时，一般需要使用#include语句将文件stdio.h内容包含到源文件中。“/*”和“*/”之间的是注释，可用英文或中文说明，其作用是增加程序的可读性。

例 1-2 计算 $S=\sum_{i=1}^{m} i$

```
#include<stdio.h>
int sum(int x) /* 自定义函数 sum() 求 1+2+3+…+x */
{ int i,y;
  y=0;
  for (i=1;i<=x;i++)
  y=y+i;
```

```
    return(y);
}
main()
{ int n,s;
  scanf("%d",&n);
  s=sum(n); /* 函数调用,用实参 n 代替形参 x */
  printf("%s %d","sum of 1+2+…n is",s); /* 输出 s */
}
```

在该程序中,自定义了 sum(int x)函数,用于计算 1+2+3+…+x,x 为参数。程序运行时先调用 mian()函数,执行到 scanf()时,从键盘输入 n 值,然后调用 sum(n)函数计算 1+2+3+…+n 并将所求之值 y 返回调用函数 main(),赋予变量 s,最后输出 s 之值。

1.3 C 语言程序的开发过程

1.3.1 C 语言程序的开发过程

C 语言程序需编译执行,它的源代码必须经过编译转化为机器代码——目标文件,并通过连接生成可执行文件然后才能运行。因此,C 语言程序的开发过程需经过以下几个步骤:

1. 编辑源程序

大多数语言编译系统带有独立的编辑程序,可用于输入或修改程序。也可以使用操作系统提供的编辑程序如 MS-DOS 下的 EDIT 等。

源程序经编辑程序由键盘输入后,形成源程序文件以文本文件的形式存储在外存储器(软盘或硬盘)中。源程序文件的名字由用户选定,但扩展名均为 c(即源程序文件均带有后缀.c)。

2. 编译源程序

在程序运行之前,必须用系统提供的编译程序对源程序文件进行编译。编译程序要进行语法检查,若没有发现错误,那么编译后将产生目标文件。目标文件是由"目标代码"组成的,目标代码通常是指可在机器上直接运行的二进制码(机器码)。目标文件带有.obj 后缀。若编译程序发现有错误,则输出错误信息,此时程序员应对程序进行再编辑,改正程序错误后,再进行编译,直到编译正确为止。

3. 连接目标文件及库文件

源程序经编译程序编译后产生的目标文件虽然是由机器指令代码组成,但是它还是不能直接在机器上运行,它是一个浮动的程序模块,也就是说,它是可重定位的。这意味着其中机器码指令的内存地址并未绝对地确定,只有偏移量是确定的。因此,程序要重定位在确定的绝对地址上,这由连接程序完成。并且,所有的 C 语言编译程序都是和标准函数库文件一起提供的,保存在函数库文件中的函数也都是可重定位的。当用户 C 语言程序应调用了一个标准库函数(或别人编写的函数)时,编译程序"记忆"它的名字,随后,连接

程序(LINK)将用户编写的程序产生的目标代码同标准库函数文件中的的目标代码结合起来，这个过程称为“连接”。连接时，对程序和函数进行重新定位，内存偏移量被用来产生实际地址(绝对地址)。经连接重新定位后产生的文件称为可执行文件，可执行文件以.exe作为后缀，它可以在机器上直接运行。

4. 运行程序

经过编译连接后产生的可执行文件(.exe 文件)，只需在操作系统下打入可执行文件名，程序即可启动运行。

综合以上所述，开发一个 C 语言程序的过程如图 1-1 所示。

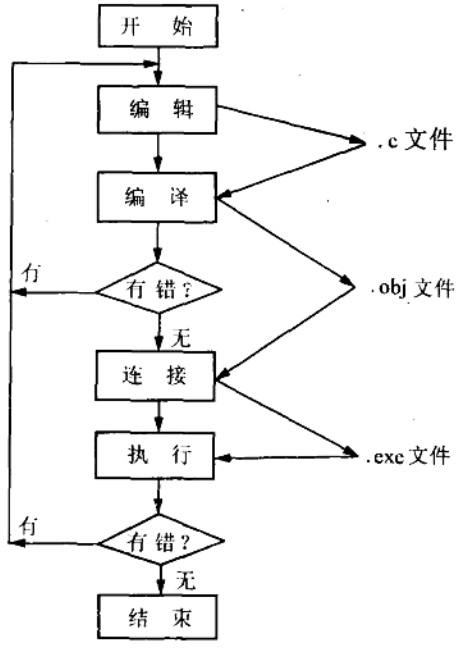


图 1-1　C 语言程序的开发过程

1.3.2　Turbo C 集成开发环境的使用简介

Turbo C 是美国 Borland 公司的产品。它为程序员提供了功能很强且使用方便的程序设计与调试环境，因而得到广泛应用。

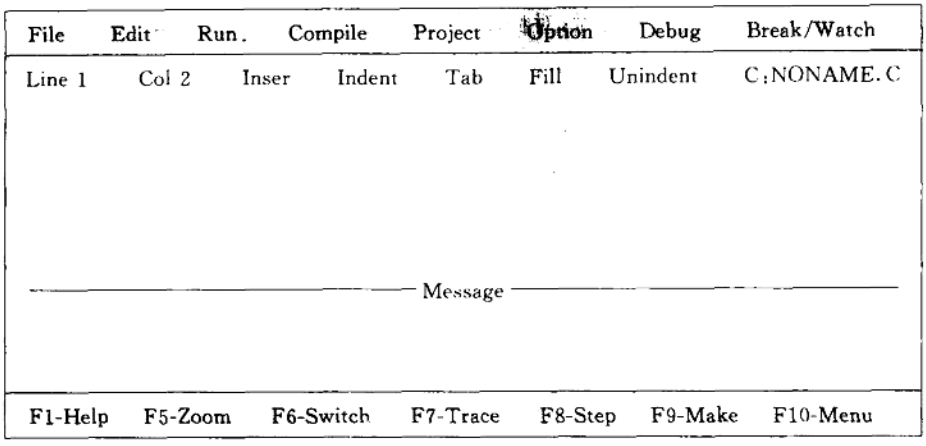
1. Turbo C 的安装

在 Turbo C 的系统盘上，有一个名为 install 的安装程序，只要执行 install 程序并根据提示操作，就可以将软盘上的 Turbo C 程序安装到硬盘上。安装结束后，通常在硬盘上自动建立了“TC”子目录，存放 Turbo C 系统程序的主文件。同时，在“TC”目录下还建立了两个子目录：“INCLUDE”和“LIB”，它们分别存放 Turbo C 系统程序的头文件和库文件。

2. Turbo C 的使用

将 TC 正确安装后，进入 TC 存放目录，键入“TC”便可启动 Turbo C 集成开发环境，

这时，在屏幕上显示如下窗口：



在该窗口最上面一行是主菜单，它提供下面几种功能：

1) File：文件操作

2) Edit：编辑程序

3) Run：运行程序

4) Compile：编译连接程序

5) Project：多文件项目的程序管理

6) Option：设置集成工作环境

7) Debug：在线调试，监视或跟踪程序的运行过程

8) Break/Watch：设置/消除断点，在线监视和跟踪表达式

下面仅就常用的命令：File、Edit、Compile 作一简单介绍：

(1) File 命令

File 主菜单中包含多个子命令，其中常用的有 Load(装入)：从磁盘调入一个文件到当前编辑窗口；Save(保存，快捷键 F2)：保存当前正在编辑的文件，若文件名是 NONAME.C，则询问是否要输入新的文件名；Write(写入)：保存当前正在编辑的文件，每选择一次，都要输入一个文件名；Quit(退出)：退出 Turbo C，返回 DOS 提示符。

(2) Edit 命令

在 Turbo C 中，只要光标位于编辑窗口内，就可以编辑程序。按 F10 键可以使光标在主菜单和编辑窗口之间切换。选择 Edit 命令也可以进入编辑状态。

(3) Compile 命令

Compile 菜单中包含多个子命令，其中常用的有以下几个：

Compile to OBJ(编译生成目标代码)：此命令将源文件(.c)编译成目标文件(.obj)。为了得到(.exe)可执行文件，还需要执行 Link EXE File 命令。

Link EXE File(连接生成可执行文件)：将目标文件(.obj)连接成可执行文件(.exe)。

Make EXE File(生成可执行文件，快捷键 F9)：此命令包括编译和连接两个过程，可以直接将源文件(.c)编译连接成可执行文件(.exe)。

(4) Run 命令(快捷键 ctrl+F9)

经过上述操作得到的.exe 文件,就可以用 Run 命令执行,也可以在 DOS 提示符下直接执行。

3. Turbo C 编译和运行环境设置

为了保证 Turbo C 编译和运行的顺利进行,在 Option 菜单中有几个环境参数选项直接影响 TC 的工作。它们是 Option 菜单下 Directories 选择项,用来设定编译器和连接器的文件查找和存放目录。其中在设置头文件查找目录选项"Directories/Include directories"中应设置:"c:\tc\include";在设置库文件查找目录选项"Directories/Library directories"中应设置:"c:\tc\lib"。除了设置上述目录外,还应在启动文件 autoexec.bat 中加入:"path=c:\tc",这样就可以在任意目录下启动 TC 了。

习 题 一

1.1　简述 C 语言程序结构及书写格式。

1.2　请指出下列 C 语言程序中的错误,并改正之。

```
/* The program is error program!
Main();
{
INT a,b,c,sum
a=1;b=2;c=3;
SUM=A+B+C;
Printf("SUM=%d",sum)
}
```

1.3　参照本章例题,编写一个 C 语言程序,求 x,y 两个整数之差。

1.4　在你的计算机上,用 Turbo C 集成开发环境将例 1-1 和例 1-2 进行编辑、编译、连接和运行。

```
#include <stdio.h>
main()
{
  int a,b,sum;
scanf("%d %d",&a&b);
  sum=a-b
printf("The sum of %d and %d is %d",a,b,sum);
}
```

第2章 数据类型、运算符与表达式

数据与操作是构成程序的两个要素。本章主要介绍c语言描述数据的形式以及对数据的基本操作。

2.1 C语言的数据类型

前面一章已介绍函数体主要由两部分组成，数据说明和语句。为了完成给定的功能，由各个语句描述出具体动作——先做什么，后做什么。而动作的操作对象是数据。在一个程序中，往往要用到很多数据，那么怎样描述数据的性质？比如，它们在什么范围取值才有意义？它们在内存中占多大空间？允许对它们进行何种运算等，这些是由数据类型来决定的。

C语言提供有丰富的数据类型如下所示：

- C数据
  - 基本类型
    - 数值类型
      - 整型
        - 基本类型(int)
        - 短整形(short)
        - 长整形(long)
        - 无符号型
          - 无符号整形(unsigned int)
          - 无符号短整形(unsigned short)
          - 无符号长整形(unsigned long)
      - 实型
        - 单精度(float)
        - 双单精度(double)
    - 字符类型(char)
    - 枚举类型(enum)
  - 构造类型
    - 数组类型 (array)
    - 结构类型 (struct)
    - 共用体类型 (union)
    - 文件类型 (file)
  - 指针类型
  - 空类型(void)

本章主要介绍基本类型(枚举类型除外)，其他类型将在以后章节中逐个介绍。