高等学校应用型系列规划教材

GAODENGXUEXIAO
YINGYONGXING
XILIEGUIHUAJIAOCAI

C语言程序设计基础

C YUYAN CHENGXU SHEJI JICHU

张鑫　王翠萍　主编

北京理工大学出版社
BEIJING INSTITUTE OF TECHNOLOGY PRESS

C语言程序设计基础

张 鑫 王翠萍 主编
杨 龙 张 林 副主编

北京理工大学出版社
BEIJING INSTITUTE OF TECHNOLOGY PRESS

内 容 简 介

本书针对高职高专教育的特点，以 Turbo C 2.0 为学习环境，突出基础性、先进性、实用性、操作性，注重对学生创新能力、实践能力和自学能力等各种应用能力的培养，由浅入深介绍了 C 语言的基本知识及程序设计的基本方法与编程技巧。

本书共分为 10 章，从 C 语言程序的构成、数据类型、运算符与表达式等基础开始讲解，由浅入深分别介绍了三种结构化程序设计、数组、函数、变量存储类型、指针、结构体和共用体以及文件等内容。全书以程序设计为主线，将实例的设计与分析贯穿始终，着重培养学生编写代码能力，并为每章配有练习题以及综合性的编程实例，有利于学生理解、消化和掌握各章节的学习内容。

参与本书编写的作者长期从事程序设计语言、软件工程等方面的教学与研究工作，具有较高的程序设计与教学水平。作者将多年来的教学经验融入本书，使其内容更易理解、实用性更强。本教材可作为高职院校计算机专业、电子专业以及信息类相关的非计算机专业的本科生的 C 语言程序设计课程的教材，还可作为各类计算机培训的教学用书及计算机工作者和爱好者的参考使用。

图书在版编目(CIP)数据

C 语言程序设计基础 / 张鑫，王翠萍主编. —北京：北京理工大学出版社，2006.8
ISBN 7-5640-0682-X

I.C… II.①张… ②王… III. C 语言-程序设计-高等学校-教材 IV.TP.312

中国版本图书馆 CIP 数据核字(2006)第 101416 号

出版发行 / 北京理工大学出版社
社　址 / 北京市海淀区中关村南大街 5 号
邮　编 / 100081
电　话 / (010)68914775(总编室)　68944990(批销中心)　68911084(读者服务部)
网　址 / http://www.bitpress.com.cn
经　销 / 全国各地新华书店
印　刷 / 北京市业和印务有限公司
开　本 / 787 毫米×1092 毫米　1/16
印　张 / 14.75
字　数 / 350 千字
版　次 / 2006 年 8 月第 1 版　2006 年 8 月第 1 次印刷　　责任校对 / 张　宏
定　价 / 26.00 元　　责任印制 / 母长新

前　言

C 语言是国内外广泛使用的计算机语言，它蕴含丰富的程序设计基础方法，是现代高级程序设计的基础。C 语言既可用来写系统软件，也可用来写应用软件；同时 C 语言也是优秀的结构化语言，且描述能力强，是计算机专业比较理想的教学语言。学生通过学习本课程，可以初步掌握“程序设计”的思想与常见问题的算法，具有阅读、分析和调试 C 语言程序的能力，并可以编写简单问题的程序，为今后的专业学习和深入的程序设计打下基础。

本书紧扣教学大纲，密切结合计算机专业实际，以应用为目的，将传统题材删繁就简，突出重点、内容少而精，加强基本概念和基本分析方法的介绍；叙述采用启发式，逻辑性强、文字流畅、通俗易懂；内容新颖、体系合理,是学习 C 语言的理想教材。

根据高职高专的教学特点，本书增加大量的典型实例，详细介绍了 C 语言应用程序设计的流程、方法、技巧及设计理念。本书特别注重实践环节内容的编写，全书以程序设计为主线，实例的设计与分析贯穿始终，同时，增加了“程序实例”环节，给出详细的程序分析，以及参考程序源码。其目的在于更深入地理解和掌握程序设计教学中的基本概念，提高使用程序设计基本技术和解决实际问题的能力。

本书由张鑫、王翠萍任主编，由杨龙、张林任副主编。参加本书编写的还有：胡菊芬、孔璐、范苗苗、王倩、张静和吴葳葳。

限于编者的水平有限，错误和不妥之处在所难免，敬请广大读者和专家批评指正。

编　者

目　录

第1章　C语言概述

C 语言是国际上广泛流行的计算机高级语言，既可用来编写系统软件，也可用来编写应用软件；同时 C 语言也是理想的结构化语言，且描述能力强，是计算机专业比较理想的教学语言。掌握了 C 语言后，再学 C++、Java、C#就比较容易了。

本章通过介绍 C 语言的发展及特点、C 程序总体结构与书写规则、C 语言的语句和关键字的概念、C 运行环境下 Turbo C 的基本操作，从一个比较低的层次，让读者尽快把握 C 程序设计精髓。

1.1　C 语言的发展简史和特点

1.1.1　C 语言的诞生与发展

在 C 语言诞生以前，系统软件主要是用汇编语言编写的。由于汇编语言程序依赖于计算机硬件，其可读性和可移植性都很差；一般的高级语言又难以实现对计算机硬件的直接操作(这正是汇编语言的优势)，于是人们盼望有一种兼有汇编语言和高级语言特性的新语言。

C 语言是贝尔实验室于 20 世纪 70 年代初研制出来的，后来又被多次改进，并出现了多种版本。80 年代初，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 ANSI C 标准(1989 年再次做了修订)。

最初，设计 C 语言的目的只是为了描述和实现 Unix 操作系统。目前，C 语言已独立于 Unix 系统，先后被移植到大、中、小型计算机及微机上，在微机上广泛使用的 C 语言编译系统有 Microsoft C、Turbo C 、Borland C 等。虽然它们的基本部分都是相同的，但还是有一些差异，所以请大家注意自己所使用的 C 编译系统的特点和规定(参阅相应的手册)。

本书选定的上机环境是 Turbo C。

1.1.2　C 语言的特点

在当前的计算机编程领域中，有大量的高级语言可以选择，如 C、Perl、BASIC、Java 和 C#。这些都是非常卓越的语言，适合完成大部分编程任务。虽然如此，但基于以下几个原因，很多计算机专业人员认为 C 语言是最佳的。

(1) 语言简洁、紧凑，使用方便、灵活。

(2) 运算符极其丰富。

(3) 生成的目标代码质量高，程序执行效率高。

(4) 可移植性好(较之汇编语言)。

(5) 可以直接操纵硬件。

1.2 C语言程序的结构与书写规则

1.2.1 C语言程序的总体结构

一个完整的C语言程序，是由一个main()函数(又称主函数)和若干个其他函数结合而成的，或仅由一个main( )函数构成。

【例1-1】 由main( )函数构成的C语言程序。

```
/*功能:输出文本信息*/
main()
{
printf("This is a C program.\n");
}
```

程序运行结果：

```
This is a C program.
```

【例1-2】 由main( )函数和1个其他函数add()构成的C语言程序。

```
/*功能:求两数之和*/
int  add (int x,int y)
{
  return x+y;
}
main()
{
  int a,b;
  a=10;
  b=20;
  printf("%d\n",add(a,b));
}
```

程序运行结果：

```
30
```

说明：

(1) 函数是C语言程序的基本单位。

main()函数的作用，相当于其他高级语言中的主程序；其他函数的作用，相当于子程序。

(2) C语言程序总是从main( )函数开始执行。

一个C语言程序，总是从main( )函数开始执行，最后回到main( )函数，而不论其在程

序中的位置如何，当主函数执行完毕时，亦即程序执行完毕。

习惯上，将主函数 main( )放在最前头。

1.2.2　函数的一般结构

任何函数(包括主函数 main( ))都是由函数说明和函数体两部分组成的。其一般结构如下：

```
[函数类型]    函数名(函数参数表)                    } 函数说明部分
              {
                  说明语句部分;                     }
                  执行语句部分;                     }  函数体部分
              }
```

1. 使用的语法符号约定

[……]　——方括号表示可选(即可以指定，也可以缺省)。

……　——省略号表示前面的项可以重复。

|　——多(含 2)中选 1。

2. 函数说明

函数说明由函数类型(可缺省)、函数名和函数参数表三部分组成，其中函数参数表的格式为：

数据类型　形参 1 [，数据类型　形参 2……]

例如，**【例 1-2】**中的函数 add( )，其函数说明各部分如图 1.1 所示。

```
函数类型    函数名    数据类型  函数参数  数据类型  函数参数
   ↓          ↓         ↓         ↓         ↓         ↓
  int        add       (int       x,       int        y)
```

图 1.1　函数说明部分结构图

注意：在旧标准中，函数可以缺省参数表。

3. 函数体

函数体是在函数说明部分的下面，大括号(必须配对使用)内的部分。函数体一般由说明语句和可执行语句两部分构成。

(1) 说明语句：由变量定义、自定义类型定义、自定义函数说明、外部变量说明等组成。

(2) 可执行语句：一般由若干条可执行语句构成。图 1.2 是**【例 1-2】**的 main( )函数体的示意图。

```
/*主函数 main()*/
main()
{
    int a,b;
    a=10;
    b=20;
    printf("%d\n",add(a,b));
}
```

变量定义部分

可执行语句部分

函数体

图 1.2 函数体结构示意图

4. 说明

(1) 函数体中的变量定义语句，必须在所有可执行语句之前。下面程序中变量定义语句“int max;”的位置是非法的：

```
main()
{
    int x,y;      /*变量定义语句:定义 2 个整型变量 x、y*/
    x=3;          /*可执行的赋值语句:将 3 赋值给变量 x*/
    y=6;          /*可执行的赋值语句:将 6 赋值给变量 y*/
    int max;      /*变量定义语句:出现在可执行的赋值语句"x=3;"和" y=6;"之后,非法! */
    max=x>y?x:y;
    printf("max=%d\n",max);
}
```

解决办法很简单，请读者自己思考。

(2) 如果函数不需要变量，也可以缺省变量定义语句。

1.2.3 源程序书写格式

(1) 所有语句都必须以分号“;”结束，函数的最后一个语句也不例外。

(2) 程序行的书写格式自由，既允许 1 行内写几条语句，也允许 1 条语句分写在几行上(但尽量避免一行写多条语句)。

(3) 允许使用注释，其格式为：/* …… */。

① /*和*/必须成对使用。

② 注释的位置，可以单占一行，也可以跟在语句的后面。

③ 如果一行写不下，可另起一行继续写。

④ 注释中允许使用汉字。在非中文操作系统下，看到的是一串乱码，但不影响程序运行。

说明：为避免遗漏必须配对使用的符号，例如注释符号、函数体的起止标识符(大括号)、圆括号等等，在输入时，可连续输入这些起止标识符，然后再在其中进行插入来完成内容

的编辑。在起止标识符嵌套以及相距较远时，这样做更有必要。

1.3　C 语言的语句和关键字

1.3.1　C 语言的语句

C 语言是结构化程序设计语言，与其他高级语言一样，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 语言的语句分为五类。

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，又可细分为三种：

(1) 选择结构控制语句有

```
if()~else~, switch()~
```

(2) 循环结构控制语句有

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

(3) 其他控制语句有

```
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：

```
printf("This is a C program! ");
```

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。

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。例如，下面就是一个空语句：

```
;
```

5. 复合语句

复合语句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。

1.3.2 关键字

C语言的关键字共有32个，根据关键字的作用，可分为数据类型关键字、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。

1.4 Turbo C的基本操作

Turbo C是在微机上广泛使用的编译程序，它具有方便、直观、易用的界面和丰富的库函数。它向用户提供一个集成环境，把程序的编辑、编译、连接、运行与调试等操作全部集中在一个界面上进行，使用十分方便。

1.4.1 运行一个C语言程序的一般过程

Turbo C是用菜单驱动的集成软件环境。运行一个C语言程序的一般过程如下：

(1) 启动Turbo C，进入Turbo C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) 退出Turbo C集成环境，结束本次程序运行。

1.4.2　Turbo C 的启动、退出与命令菜单

1. Turbo C 的启动

在安装有 Turbo C 的计算机中启动 Turbo C 有三种方法：

(1) 在 DOS 平台上，进入含有 Turbo C 的子目录，输入命令 TC 回车；

(2) 在 Windows 平台上，打开含有 Turbo C 的子目录后，双击可执行文件 TC；

(3) 在桌面上为 Turbo C 建立一个快捷方式，直接双击该快捷方式。

Turbo C 集成环境如图 1.3 所示。

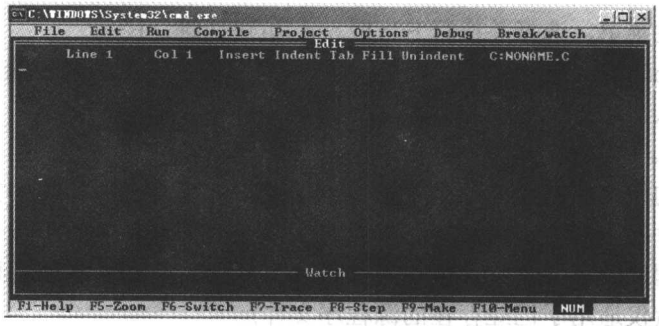

图 1.3　Turbo C 集成环境

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：

- Line 1 和 Col 1：显示光标所在的行号和列号，即光标位置。
- Insert：表示编辑状态处于“插入”。当处于“改写”状态时，此处为空白。
- C: 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“控制键”状态)。

2. 命令菜单的使用

在 Turbo C 集成环境中，可以使用命令菜单完成命令项的选择和控制，其具体操作方法如下：

(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。

3. Turbo C 的退出

退出 Turbo C 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. 编辑一个 C 语言源程序

首先，激活主菜单，选择并执行 File | Load 项(快捷键：F3)。

然后，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。

2. 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。这是一个良好的习惯！

3. 关于在线帮助

- 在任何窗口(或状态)下，按 F1 键都可以激活活动窗口(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。

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

1.4.4　编译、连接——单个源程序文件

选择并执行 Compile | Make EXE File 项(快捷键：F9)，则 Turbo C 将自动完成对当前正在编辑的源程序文件的编译、连接，并生成可执行文件。

(1) 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)。

(2) 按下任意键。此时，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　运行与查看结果

如果程序编译、连接成功，就可以运行并查看运行结果。

1. 运行当前正在编辑的源程序文件

选择并执行 Run | Run 项(快捷键：^F9)，程序运行结束后，仍返回到编辑窗口。

当你认为自己的源程序不会有编译、连接错误时，也可直接运行(即跳过对源程序的编译、连接步骤)。这时，Turbo C 将一次完成从编译、连接到运行的全过程。

2. 查看运行结果

选择并执行 Run | User Screen 项(快捷键：Alt+F5)。查看完毕后，按任一键返回编辑窗口。

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　创建新的源程序

要创建和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”，进入下一步操作；否则，键入“N”(不保存)，跳转到 2。

(1) 系统提示换名：

```
<d:><path>\NONAME.C
```

直接输入给源程序文件起的名字即可。