



21 世纪高职高专规划教材 计算机系列

实用C语言 程序设计教程

石从刚 孟祥伟 主编

- 数名一线教师多年教学经验集萃
- 以实例的教学模式编写
- 图文并茂、条理清晰、易教易学
- 免费提供 PPT 格式电子教案



中国电力出版社
www.infopower.com.cn



21 世纪高职高专规划教材 · 计算机系列

实用 C 语言 程序设计教程

石从刚 孟祥伟 主 编
孙学农 鲁 红 孙 艳 副主编
宋剑英 陈会秋 李艳玲 刘天庆 陈 健 参 编
王美平 主 审



中国电力出版社

www.infopower.com.cn

内容提要

C 语言是程序设计的入门语言,也是许多高职院校为学生开设的第一门程序设计课程,本书充分考虑到高职高专教学的特色,理论上必需、够用,注重理论联系实际,突出实用性,语言组织上通俗易懂,做到在内容的编排上尽量适应初学者的要求。

全书共分 10 章,主要包括:C 语言概述、C 语言的基本知识、运算符和表达式、顺序和选择结构程序设计、循环结构程序设计、数组、函数、指针、结构体、文件等。本书在体系结构安排上尽可能将概念、知识点与案例结合起来,进行案例教学,在实例选择上从易到难,使得老师好教、学生易学。

本书可作为高职高专相关专业学生学习 C 语言程序设计的教材,同时也可作为广大计算机编程爱好者学习 C 语言的自学教材和参考书。

图书在版编目(CIP)数据

实用 C 语言程序设计教程 / 石从刚等主编. —北京:中国电力出版社, 2006.1

ISBN 7-5083-3882-0

I. 实… II. 石… III. C 语言—程序设计—高等学校; 技术学校—教材 IV. TP312

中国版本图书馆 CIP 数据核字 (2005) 第 140120 号

丛 书 名: 21 世纪高职高专规划教材·计算机系列

书 名: 实用 C 语言程序设计教程

出版发行: 中国电力出版社

地 址: 北京市三里河路 6 号

邮政编码: 100044

电 话: (010) 68362602

传 真: (010) 68316497, 88383619

本书如有印装质量问题, 我社负责退换

服务电话: (010) 88515918 (总机)

传 真: (010) 88518169

E-mail: infopower@cepp.com.cn

印 刷: 汇鑫印务有限公司

开本尺寸: 185×233

印 张: 13.5

字 数: 297 千字

书 号: ISBN 7-5083-3882-0

版 次: 2006 年 2 月北京第 1 版

印 次: 2006 年 2 月第 1 次印刷

印 数: 0001—4000 册

定 价: 19.80 元

版权所有, 翻印必究

前 言

C 语言是一种应用十分广泛的结构化程序设计语言。C 语言功能丰富、表达能力强、使用灵活方便、应用面广、程序执行效率高、可移植性好,既具有高级语言的特点,又具有汇编语言的特点和较强的系统处理能力,因此成为开发系统软件和应用软件的首选语言。UNIX、Linux 操作系统都是使用 C 语言编写的。

现在, C 语言是许多大学都在开设的重要基础课之一, 特别是在高职高专的课程设置中, C 语言被工科各专业作为程序设计语言的必修课程。同时, C 语言也已成为获取计算机专业技术资格的程序员证书和参加计算机等级考试的首选语言。

在编写本书的过程中, 作者充分考虑高职院校把 C 语言作为程序设计的入门语言的特点, 同时结合自己多年从事 C 语言教学的实践, 理论联系实际, 内容由浅入深, 力求通俗易懂。每章开头都列出了该章的学习重点, 让学生明白该章的学习任务和要掌握的知识重点, 同时每章末给出了该章小结, 通过小结可检查读者对该章主要内容的掌握情况。本书在体系结构安排上尽可能地将概念、知识点与案例结合起来, 进行案例教学, 在实例选择上从易到难, 读者通过对实例的认真思考和总结, 定能举一反三, 提高 C 语言的编程能力。

全书共十章, 第 1 章是 C 语言概述, 第 2 章介绍了 C 语言的基本知识, 第 3 章介绍了运算符和表达式, 第 4 章和第 5 章介绍了结构化程序设计的三种基本结构, 即顺序结构、选择结构、循环结构, 第 6 章介绍了数组, 第 7 章介绍了函数, 第 8 章介绍了指针, 第 9 章介绍了结构体和共用体, 第 10 章介绍了文件操作。

作为本书配套教材, 我们将同时出版本教材的习题解答和实验指导书, 供读者学习时借鉴和参考。

本书由石从刚和孟祥伟主编, 王美平副教授主审, 孙学农、鲁红、孙艳副主编, 宋剑英、陈会秋、李艳玲、刘天庆、陈健参编。全书由石从刚统稿, 王美平在百忙中审阅了全书, 并提出了宝贵的建议和意见。在此作者对关心和参与本书编写的所有人员表示感谢。

由于作者水平有限, 书中缺点和错误在所难免, 恳请广大读者予以批评指正。

作 者
2005 年 9 月

目 录

前 言

第 1 章 C 语言概述.....	1
1.1 C 语言发展概述和主要特点	1
1.2 简单的 C 程序介绍	2
1.3 C 程序的调试	4
本章小结	7
习题	7
第 2 章 C 语言的基本知识	8
2.1 C 语言的数据类型	8
2.2 字符集和标识符	9
2.3 变量与常量	10
2.4 整型数据	12
2.5 实型数据	17
2.6 字符型数据	19
本章小结	22
习题	22
第 3 章 基本运算符和表达式	24
3.1 表达式	24
3.2 算术运算符和算术表达式	24
3.3 赋值运算符	26
3.4 不同数据类型间的混合运算	28
3.5 关系运算符和关系表达式	29
3.6 逻辑运算符及逻辑表达式	31
3.7 自增运算符和自减运算符	32
3.8 逗号运算符及逗号表达式	34
3.9 位逻辑运算符	34
3.10 求字节数运算符	36

本章小结.....	36
习题.....	37
第 4 章 顺序和选择结构程序设计.....	40
4.1 语句概述.....	40
4.2 结构化程序设计的三种基本结构.....	41
4.3 顺序结构程序设计.....	42
4.4 选择结构程序设计.....	44
4.5 switch 语句.....	49
4.6 条件运算符.....	51
4.7 程序举例.....	52
本章小结.....	55
习题.....	56
第 5 章 循环结构程序.....	58
5.1 概述.....	58
5.2 while 语句.....	58
5.3 do-while 语句.....	60
5.4 for 语句.....	62
5.5 循环的嵌套.....	65
5.6 break 语句和 continue 语句.....	66
5.7 程序举例.....	68
本章小结.....	70
习题.....	71
第 6 章 数组.....	73
6.1 问题的提出.....	73
6.2 一维数组.....	73
6.3 二维数组.....	79
6.4 字符数组.....	83
本章小结.....	89
习题.....	89
第 7 章 函数.....	91
7.1 问题的提出.....	91
7.2 函数的定义.....	93

7.3 函数说明	98
7.4 函数调用	99
7.5 参数传递	102
7.6 函数的递归调用	108
7.7 局部变量与全局变量	112
7.8 变量的存储类别	116
7.9 内部函数和外部函数	124
本章小结	125
习题	126
第 8 章 指针	129
8.1 指针的基本概念	129
8.2 指针与变量	129
8.3 指针与一维数组	132
8.4 指针与二维数组	135
8.5 指针与函数	138
8.6 指针数组和二级指针	143
本章小结	147
习题	147
第 9 章 结构体与其他数据类型	149
9.1 结构体	149
9.2 结构体数组	156
9.3 结构体与指针	159
9.4 共用体	170
9.5 枚举类型	173
本章小结	174
习题	175
第 10 章 文件	176
10.1 C 文件概述	176
10.2 文件的打开和关闭	178
10.3 文件的读写操作	180
10.4 文件的定位	186
10.5 程序举例	188
本章小结	192

习题.....	192
附录 A 常用字符与 ASCII 代码对照表	198
附录 B C 语言中的关键字	199
附录 C 运算符的优先级和结合性	200
附录 D Turbo C 常用库函数.....	201
参考文献.....	205

第 1 章 C 语言概述

本章学习重点

- (1) 明确 C 语言的发展历史和主要特点。
- (2) 掌握 C 语言程序的调试步骤和 Turbo C 集成开发环境。

1.1 C 语言发展概述和主要特点

1.1.1 C 语言的发展历史

C 语言是国际上广泛流行的计算机高级语言，与 UNIX 操作系统有着密切的关系，它的发明者是 Dennis Ritchie 和 Brian W. Kernighan，他们开发 C 语言的主要目的是为了更好地描述 UNIX 操作系统。

1970 年，美国贝尔实验室的 KEN Thompson 以 BCPL 语言为基础设计出了很简单而且很接近硬件的 B 语言，并用 B 语言写了第一个操作系统，在 PDP-7 上实现。但 B 语言过于简单，缺乏丰富的数据类型。

1972 年至 1973 年间，贝尔实验室的 D. M. Ritchie 在 B 语言的基础上设计出了 C 语言。C 语言既保持了 BCPL 和 B 语言的优点（精炼，接近硬件），又克服了二者的缺点（过于简单，数据无类型）。1973 年，K. Thompson 和 D. M. Ritchie 两人合作把 UNIX 的 90% 以上用 C 语言改写，即 UNIX 第 5 版。

后来 C 语言作了多次改进，在 1978 年，Brian W. Kernighan 和 Dennis Ritchie 合著了一本著名的书《The C Programming Language》，该书介绍的 C 语言被称为标准 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，ANSI 公布了标准 ANSI C。

1990 年，国际标准化组织（International Standard Organization, ISO）也采用了 ANSI C 标准，称为 ANSI/ISO standard C。

本书的叙述基本上以 ANSI C 为基础。目前流行的各种 C 语言编译系统版本虽然基本是相同的，但也有一些不同。在微机上使用的有 Microsoft C、Turbo C、Quick C、BORLAND C 等，它们的不同版本又略有差异。

1.1.2 C 语言的主要特点

虽然目前读者还不能亲身体会 C 语言的特点，但希望通过下面的叙述可以使读者对 C 语

言的特点有一个基本的了解，并在今后的学习中不断加深对该语言的理解。

C 语言的主要特点如下：

(1) 语言简洁、紧凑，使用方便、灵活。C 语言一共只有 32 个关键字（见附录 B）、9 种控制语句，程序书写自由，主要用小写表示。

(2) C 语言同时具备高级语言和低级语言的特征。高级语言具有的可读性好、容易记忆、可移植性强等优点 C 语言都有，同时 C 语言还具备汇编语言的大部分功能，如允许直接访问物理地址、能进行位操作以及指定用寄存器存放变量等。

(3) C 语言是理想的结构化程序设计语言，具有结构化的控制语句（如 if...else 语句、while 语句、do...while 语句、switch 语句、for 语句）。

(4) C 语言支持模块化程序设计。C 语言的程序是由函数构成的，每个函数可以单独编写和调试。

(5) C 语言的运算符多，功能强大，共有 34 种运算符（见附录 C）。

(6) C 语言具有丰富的数据类型。C 语言的数据类型有整型、实型、字符型、数组类型、指针类型、结构体类型、共用体类型等。

(7) 生成的代码质量高。一般只比汇编语言生成的目标代码效率低 10%~20%。

1.2 简单的 C 程序介绍

下面将通过几个简单的程序例子，使读者对 C 程序的组成和特性有个基本的认识。

【例 1-1】在屏幕上显示 This is a c program 字样。

```
/* ---A program to print This is a c program--- */
#include "stdio.h"
void main()
{
    printf("This is a c program.\n");
}
```

本程序的作用是输出以下一行信息：

This is a c program.

这是一个简单的 C 程序，但是已经充分说明了 C 程序的基本组成。该程序包括了三个部分：注释、预处理命令及函数定义。对该程序的分析 and 说明如下：

(1) 程序开始用/*和*/括起来的是注释行，用于说明程序的功能和目的，只是作注释用，对编译和运行不起作用。注释可以放在程序的任何位置。

(2) 以#开始的是预处理命令。这些命令是在编译系统翻译代码之前需要由预处理程序处理的。本例中的#include "stdio.h"命令是请求预处理程序将文件stdio.h包含到程序中来，作为程序的一部分。

(3) main()表示主函数，每个 C 程序都必须有一个主函数。用{}括起来的部分是函数体，

C 程序的执行是从主函数的第一句开始，到主函数中的最后一句结束。

(4) C 语言的执行语句和说明语句以“;”作为结束符。

(5) `printf` 是 C 语言提供的标准输入输出函数，它的功能是将用两个双引号括起来的内容原样输出到显示器上。

【例 1-2】 计算 $a+b$ ，并在屏幕上输出结果。

```
/* -----output sum of a add b----- */
#include "stdio.h "
void main()
{
    int a,b,sum;
    a=2;
    b=3;
    sum=a+b;
    printf("sum is %d\n ",sum);
}
```

输出结果:

sum is 5

分析与说明:

- (1) 程序的第 4 行是定义变量 a 、 b 、 sum 为整型变量。
- (2) 第 5 行和第 6 行是赋值语句，使 a 和 b 的值分别为 2 和 3。
- (3) 第 7 行是使 sum 的值为 $a+b$ ，即将 a 和 b 相加之和送给 sum 。
- (4) 第 8 行中的“`%d`”是输入输出的“格式字符串”，用来指定输入输出时的数据类型和格式，“`%d`”表示十进制整数类型。

【例 1-3】 通过调用自定义函数计算 $a+b$ ，并在屏幕上输出结果。

```
# include "stdio.h "
void add(int x,int y);          /* 函数原型说明 */
void main()
{ int a,b,sum;
  a=2;b=2;
  add(a,b);
}
void add(int x,int y)
{ int sum;
  sum=x+y;
  printf("sum is %d\n",sum);
}
```

运行结果:

sum is 5

分析与说明:

C 语言中除主函数外, 程序员还可自己定义其他函数, 这些函数可以像前两例中的 `printf` 一样被调用。`printf` 是系统提供的库函数, 使用时不必定义。

本例中有两个函数, 主函数 `main` 及 `add` 函数。`main` 函数中有一句函数调用语句 `add(a,b);`, 该语句执行时将参数 `a` 和 `b` 的值分别传递给 `add` 函数中的 `x` 和 `y`, 程序转到 `add` 函数执行。`add` 函数计算 `sum` 的值并在屏幕上显示出来, `add` 函数执行完后, 返回到 `main` 函数调用语句的下一句继续执行。

注意: 程序的执行总是从主函数开始的, 与函数存放顺序无关。

1.3 C 程序的调试

1.3.1 C 程序的调试步骤

在上一节中我们已看到了一些简单的 C 程序, 实际上计算机并不能直接执行用高级语言编写的程序。为了使计算机能执行高级语言源程序, 必须先用一种称为“编译程序”的软件, 把源程序翻译成二进制形式的“目标程序”, 然后将该目标程序与系统的函数库和其他目标程序连接起来, 形成可执行的目标程序。

C 语言的编译程序属于编译系统。要完成一个 C 程序的调试, 必须经过编辑源程序、编译源程序、链接目标程序和运行可执行程序四个步骤。

C 语言的源程序是符合 C 语言语法的程序文本文件, 又称为源程序文件, 扩展名为 `.c`。许多文本编辑器都可以用来编辑源程序, 例如 Windows 写字板、Word 以及 DOS 的 Edit 等。

编辑完成以后是编译。对编辑好的文本文件进行成功编译后将生成目标程序, 目标程序的文件名与源程序的文件名相同, 扩展名是 `.obj`。编译程序的任务是对源程序进行语法和语义分析及检查, 只有源程序的语法和语义都是正确的, 才能生成目标程序; 否则, 应该重新回到编辑阶段修改源程序。

编译成功以后, 将目标程序与系统的函数库和其他目标程序链接起来, 形成可执行的目标程序。可执行的目标程序其文件名不变, 扩展名是 `.exe`。

最后一步就是运行可执行的目标程序(执行文件)。如果执行程序达不到预期的结果, 我们必须重复“编辑、编译、链接、运行”这四个步骤。C 语言程序的开发过程如图 1-1 所示。

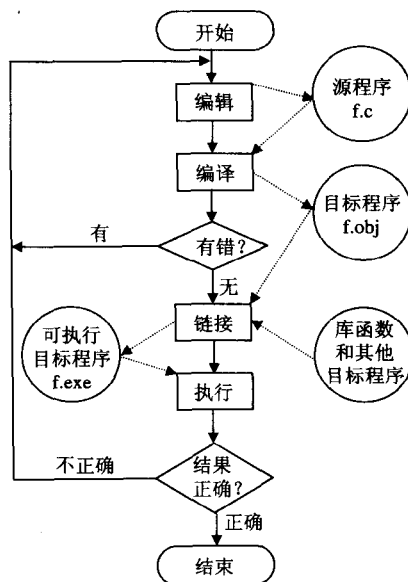


图 1-1

1.3.2 用 Turbo C 集成开发环境运行 C 程序的步骤

Turbo C 是目前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、链接和运行等操作全部集中在一个界面上进行，使用十分方便。下面介绍使用 Turbo C 2.0 编译程序调试 C 程序的方法。

(1) 调用 Turbo C 程序。Turbo C 2.0 是 DOS 环境下的调试程序，因此首先要进入 DOS 命令行环境，然后进入到 Turbo C 编译程序所在的子目录（例如 TC 子目录），之后只需键入 tc 目录即可。进入 Turbo C 2.0 集成开发环境后，看到的界面如图 1-2 所示。

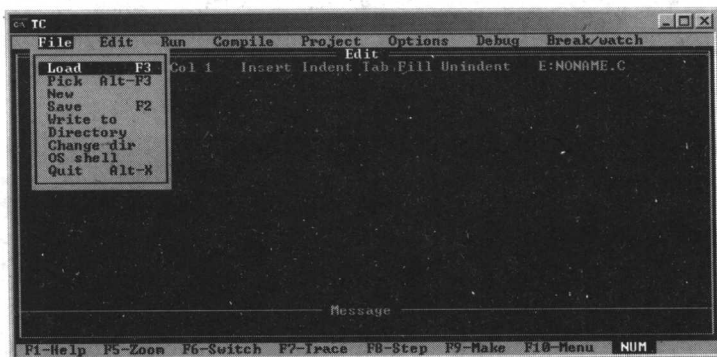


图 1-2

从图 1-2 可以看到，在集成环境的上部有一行“主菜单”，其中包括下面 8 个菜单项：File、Edit、Run、Compile、Project、Option、Debug、Break/Watch。

用户可通过以上 8 个菜单项来选择使用 Turbo C 集成环境所提供的各项主要功能。这 8 个菜单项分别代表：文件操作、编辑、运行、编译、项目文件、选项、调试、中断/观察等功能。

用键盘上的“←”和“→”键可以选择菜单条中所需要的菜单项，被选中的项以“反相”形式显示。此时若按回车键，就会出现一个下拉菜单。例如选中 File 菜单并按回车键后，屏幕上 File 下面出现下拉菜单，如图 1-2 所示。它是一个子菜单，提供多项选择。可以用“↓”键选择所需要的项，按回车键将执行该命令。

(2) 编辑 C 源程序文件。在 File 菜单下选择 New 子菜单项命令，系统的光标处于等待输入 C 源程序状态，如图 1-3 所示。

输入完源程序后，必须将源程序文件保存。保存文件的方法是在 File 菜单下选择 Save 子菜单项命令并执行，在随后弹出的【保存文件】对话框中输入文件名*.c 后按回车键，如图 1-4 所示。

(3) 编译源文件。选择 Compile 主菜单项并在其下拉子菜单中选择 Compile to OBJ 命令，开始进行编译。如果出现语法错误，系统将自动显示错误的对话框。按任意键回到编辑状态程序出错的地方，如图 1-5 所示。

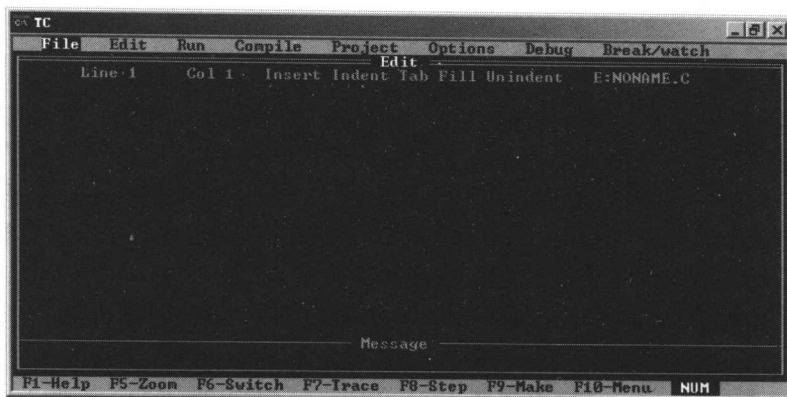


图 1-3

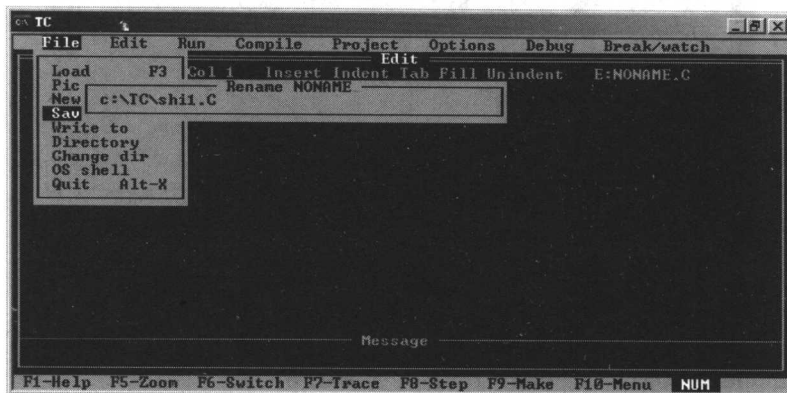


图 1-4

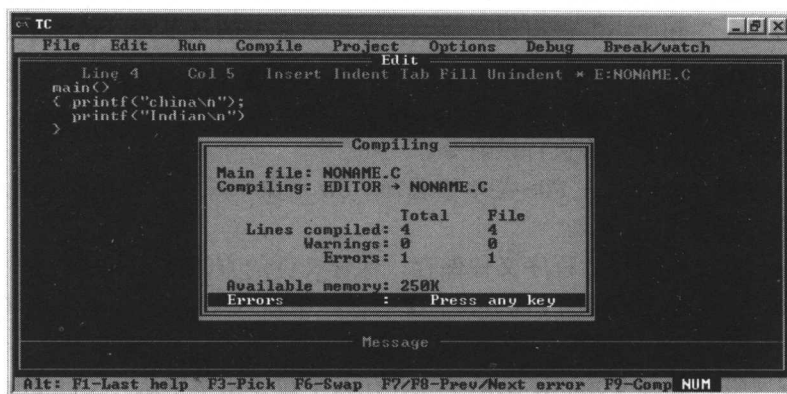


图 1-5

(4) 链接程序。在 **Compile** 菜单中选择 **Link** 子菜单项命令，程序将被链接成可执行程序。也可按 **F9** 键，一次完成编译和链接。

(5) 执行程序。在 **Run** 菜单中选择 **Run** 子菜单命令，或按快捷键 **Ctrl+F9**。

程序运行后必须按快捷键 **Alt+F5**，才能看到运行的结果。

(6) 退出 Turbo C 环境。选择 **File** 菜单项的 **Quit** 子菜单项命令，或按快捷键 **Alt+X**。

限于篇幅，对于其他命令的使用方法不在此叙述，读者可以查阅相关资料，尤其是 Turbo C 的用户说明书。在实践中积累经验也是掌握开发环境的有效方法。

本章小结

本章介绍了 C 语言的发展历史和主要特点、简单 C 程序的构成和 C 程序的调试开发过程。需要注意以下几个问题：

- (1) 计算机语言一般分为低级语言和高级语言。
- (2) C 语言同时具备低级语言和高级语言的特点。
- (3) C 语言程序由一个或多个函数组成，至少有一个且只能有一个 **main()** 函数。
- (4) C 语言程序由语句组成，通常每一行为一条基本语句，以分号结束。
- (5) C 语言程序必须经过编译、链接，最后才可运行。

习 题

1. 简述 C 语言的主要特点。
2. 请参照本章例题编写一个 C 程序，输出以下信息：

```
-----  
your name  
-----
```

3. 上机运行本章中的 3 个例题，熟悉 Turbo C 的上机方法和步骤。
4. 调试 C 程序需要经过几个步骤？编辑、编译、链接所生成的文件的扩展名分别是什么？

第2章 C语言的基本知识

本章学习重点

- (1) 明确 C 语言中标识符的取名规则。
- (2) 掌握 C 语言中有哪些数据类型。
- (3) 掌握整型变量、实型变量、字符变量的定义方法。
- (4) 掌握利用 printf 函数输出整型数据、实型数据和字符型数据的方法。
- (5) 掌握利用 scanf 函数输入整型数据、实型数据和字符数据的方法。
- (6) 掌握利用 putchar 函数和 getchar 函数分别输出和输入字符型数据的方法。

本章中介绍的知识是编写 C 程序的基础，一定要牢固掌握。实际上，C 程序设计的核心是数据运算，而数据是以某种特定的形式存在的，因此必须先掌握 C 语言编程中最基本的数据类型，以及它们的输入输出方法。本章讨论的数据类型是 C 语言的基本数据类型。

2.1 C 语言的数据类型

数据类型是高级语言程序设计中的一个重要概念。不同的数据类型在内存中的存储方式是不同的，不同数据类型的数据在内存中所占的字节数也大多不一样。

2.1.1 C 语言的数据类型分类

C 语言具有非常丰富的数据类型。C 语言的数据类型如图 2-1 所示。

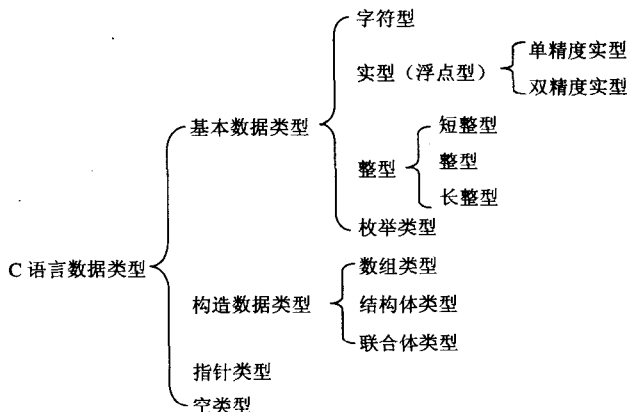


图 2-1

C 语言中数据有常量与变量之分。本章主要介绍基本数据类型。

2.1.2 基本数据类型

C 语言的基本数据类型是构成其他数据类型的基础。C 语言的基本数据类型包括字符型、实型、整型和枚举类型。

以 IBM PC 为例, 其基本数据类型及所占字节数见表 2-1。单精度实数提供 7 位有效数字, 双精度实数提供 15~16 位有效数字, 数值的范围因机器系统而异。

表 2-1 数字的范围

类 型	所占位数	数的范围	说 明
int	16	-32768~32767	普通整型 (简称整型)
short[int]	16	-32768~32767	短整型
long[int]	32	-2147483648~2147483647	长整型
unsigned[int]	16	0~65535	无符号整型
unsigned short	16	0~65535	无符号短整型
unsigned long	32	0~4294967295	无符号长整型
float	32	$10^{-38} \sim 10^{38}$	单精度实型
double	64	$10^{-308} \sim 10^{308}$	双精度实型
char	8		字符型

2.2 字符集和标识符

2.2.1 字符集

字符集是高级语言的编译系统所能识别的字母、数字和特殊符号。每种高级语言都有自己特定的字符集。

C 语言的字符集包括:

- (1) 大小写英文字母: A, B, ..., Z, a, b, ..., z
- (2) 数字: 0, 1, 2, ..., 9
- (3) 运算符: + - * / % > < = & | ? ! ^ ~
- (4) 括号: () { } []
- (5) 特殊符号: \ _ \$ #
- (6) 标点符号: ' " : ;
- (7) 空白符: 空格符、换行符、制表符

字符集中的字符按照 C 语言语法组合起来, 就能通过编译系统的语法和词法分析。不在