

全国高职高专规划教材

C语言程序设计 实训教程

Programming C
in Practice

马尚风 主 编
何 林 杜风雷 副主编



科学出版社

www.sciencep.com



全国高职高专规划教材

C 语言程序设计实训教程

马尚风 主 编

何 林 杜风雷 副主编

科 学 出 版 社

北 京

内 容 简 介

本书是《C 语言程序设计》的配套书,用于上机实训。本书共 8 章,分别是:C 程序设计基础、程序的控制结构、函数、指针与数组、指针与函数、结构体与共用体、文件、C 程序设计大型作业。每章给出实训目的与内容、实训知识准备、实训案例与实训项目,每章的最后还附有一定的练习。

本书先从示范案例开始,通过思考与给出的提示,使学生独立地完成实训项目。

本书可作为高职高专相应课程的实训教材,也是本专科相关专业学生、自考学员和专业教师颇有帮助的辅助教材。

图书在版编目(CIP)数据

C 语言程序设计实训教程/马尚风主编. —北京:科学出版社, 2003
(全国高职高专规划教材)

ISBN 7-03-011920-7

I. C... II. 马... III. C 语言—程序设计—高等学校: 技术学校—教学参考资料 IV. TP312

中国版本图书馆 CIP 数据核字(2003)第 062351 号

策划编辑:李振格/责任编辑:丁波
责任印制:吕春珉/封面设计:东方人华平面设计部

科学出版社 出版

北京东黄城根北街16号

邮政编码:100717

<http://www.sciencep.com>

新蕾印刷厂 印刷

科学出版社发行 各地新华书店经销

*

2003 年 8 月第 1 版 开本: 787×1092 1/16

2003 年 8 月第一次印刷 印张: 11

印数: 1—5 000 字数: 243 000

定价: 17.00 元

(如有印装质量问题,我社负责调换〈环伟〉)

全国高职高专规划教材编委会名单

主任 俞瑞钊

副主任 陈庆章 蒋联海 周必水 刘加海

委员 (以姓氏笔画为序)

王雷 王筱慧 方 程 方锦明 卢菊洪 代绍庆

吕何新 朱 炜 刘向荣 江爱民 江锦祥 孙光弟

李天真 李永平 李良财 李明钧 李益明 余根墀

汪志达 沈凤池 沈安衢 张 元 张学辉 张锦祥

张德发 陈月波 陈晓燕 邵应珍 范剑波 欧阳江林

周国民 周建阳 赵小明 胡海影 秦学礼 徐文杰

凌 彦 曹哲新 戚海燕 龚祥国 章剑林 蒋黎红

董方武 鲁俊生 谢 川 谢晓飞 楼 丰 楼程伟

鞠洪尧

秘书长 熊盛新

本书编写人员名单

主 编 马尚风

副主编 何 林 杜风雷

撰稿人 刘鲁楣 李忠成 傅绍娟

前 言

C 语言是一种优秀的结构化程序设计语言，因为它具有高级语言和低级语言两方面的优点，因而可以把 C 语言称为中级语言。C 语言结构严谨、数据类型完整、语句简练灵活、运算符丰富。自从编写 UNIX 操作系统获得成功以来，一直受到计算机业内人士的好评。即使在面向对象的语言广泛使用的今天，C 语言仍然是高校计算机专业的一门专业基础课。同时，C 语言本身也在不断发展，如 C++、VC，充分显示了 C 语言的活力。

面对蓬勃发展的高职教育，急需一套适合高职院校教学特点的计算机教材。浙江省高职院校的一批富有教学经验的计算机专业教师合力打造了一批计算机专业教材。

《C 语言程序设计》、《C 语言程序设计实训教程》和《C 语言程序设计例题与习题》就是其中的一套。这三本教材相辅相成，分别用于理论教学、上机实训以及课后练习提高。如果能消化这套教材，那么你就已经能熟练掌握 C 语言了。

C 语言是一门实践性很强的课程，应有一半的学时上机操作。本书内容按《C 语言程序设计》章节编排，各章首先对实训知识摘要介绍，然后设计与各章知识点相配合的实训案例与实训练习，初学者务必要认真完成、仔细体会。对学有余力的同学来说，还可以完成各章后的实训思考题。第 8 章是一个完整的 C 语言程序实例，可以帮助初学者了解大型程序的编写技巧，也是对 C 语言知识的综合应用。

本书由马尚风老师编写第 1 章和第 5 章，何林老师编写第 2 章和第 8 章，刘鲁楣老师编写第 3 章，杜风雷老师编写第 4 章，李忠成老师编写第 6 章，傅绍娟老师编写第 7 章，由马尚风老师完成最后的定稿工作。由于时间仓促加之水平有限，书中难免会有不足之处，敬请读者批评指正。

编 者

2003 年 6 月

目 录

第 1 章 C 程序设计基础.....	1
1.1 实训知识准备.....	1
1.1.1 C 语言特点.....	1
1.1.2 C 程序的数据类型.....	1
1.1.3 常量与变量.....	2
1.1.4 输入/输出函数.....	2
1.1.5 自定义函数.....	2
1.1.6 运算符与表达式.....	2
1.1.7 数组.....	3
1.1.8 间接访问.....	3
1.2 实训案例.....	3
1.3 Turbo C 的集成环境.....	4
1.3.1 进入 Turbo C.....	4
1.3.2 集成环境.....	5
1.3.3 编辑文件.....	6
1.3.4 编译、连接与运行.....	6
1.3.5 实训练习.....	7
1.4 Visual C++ 6.0 上机步骤.....	8
1.5 实训项目一.....	10
1.5.1 输入与输出.....	10
1.5.2 函数调用.....	10
1.5.3 变量.....	12
1.5.4 实训练习.....	13
1.6 实训项目二.....	14
1.6.1 算术表达式.....	14
1.6.2 关系表达式.....	15
1.6.3 逻辑运算.....	15
1.6.4 自增自减运算.....	16
1.6.5 实训练习.....	17
1.7 实训项目三.....	18
1.7.1 指针.....	18
1.7.2 一维数组.....	19

1.7.3 实训练习.....	20
习题.....	21
第 2 章 程序的控制结构	24
2.1 实训知识准备.....	24
2.1.1 算法的表示.....	24
2.1.2 三种基本结构.....	24
2.1.3 结构化程序设计方法.....	26
2.1.4 if 语句及嵌套.....	26
2.1.5 switch 语句.....	27
2.1.6 while 语句.....	27
2.1.7 do...while 语句.....	27
2.1.8 for 语句.....	27
2.1.9 循环嵌套.....	28
2.1.10 break 语句和 continue 语句.....	28
2.2 实训案例.....	28
2.3 实训项目一.....	29
2.3.1 if 语句.....	29
2.3.2 if 语句嵌套.....	30
2.3.3 实训练习.....	32
2.4 实训项目二.....	32
2.4.1 条件运算符与条件运算.....	32
2.4.2 switch 语句.....	33
2.4.3 实训练习.....	34
2.5 实训项目三.....	34
2.5.1 while 循环.....	34
2.5.2 do-while 循环.....	35
2.5.3 for 循环.....	36
2.5.4 实训练习.....	37
2.6 实训项目四.....	38
2.6.1 循环嵌套.....	38
2.6.2 break 和 continue 语句.....	39
2.6.3 实训练习.....	40
习题.....	40
第 3 章 函数	45
3.1. 实训知识准备.....	45
3.1.1 函数的说明、定义与调用.....	45
3.1.2 函数的返回值.....	45
3.1.3 函数间的数据传递格式.....	45
3.1.4 函数调用.....	46

3.1.5 变量的作用域和生存期.....	47
3.1.6 字符串处理函数.....	47
3.1.7 文件包含.....	47
3.2 实训案例.....	48
3.3 实训项目一.....	49
3.3.1 普通变量作实参.....	49
3.3.2 实训练习.....	50
3.4 实训项目二.....	51
3.4.1 函数嵌套调用.....	51
3.4.2 实训练习.....	53
3.5 实训项目三.....	54
3.5.1 递归调用.....	54
3.5.2 实训练习.....	55
3.6 实训项目四.....	57
3.6.1 数组名作实参.....	57
3.6.2 实训练习.....	58
习题.....	59
第4章 指针与数组.....	64
4.1 实训知识准备.....	64
4.1.1 指向一维数组的指针变量.....	64
4.1.2 指向字符串的指针变量.....	65
4.1.3 指向二维数组的指针变量.....	66
4.1.4 二级指针变量.....	67
4.2 实训案例.....	67
4.3 实训项目一.....	70
4.3.1 一维数组和指针.....	70
4.3.2 实训练习.....	73
4.4 实训项目二.....	74
4.4.1 字符串和字符指针.....	74
4.4.2 实训练习.....	75
4.5 实训项目三.....	75
4.5.1 二维数组和指针.....	75
4.5.2 实训练习.....	77
4.6 实训项目四.....	78
4.6.1 综合应用.....	78
4.6.2 二级指针变量访问.....	81
4.6.3 实训练习.....	81
习题.....	82
第5章 指针与函数.....	88

5.1 实训知识准备.....	88
5.1.1 函数的指针.....	88
5.1.2 返回指针的函数.....	88
5.1.3 命令行参数.....	89
5.2 实训案例.....	89
5.3 实训项目一.....	90
5.3.1 指向函数的指针变量.....	90
5.3.2 指向函数的指针变量作参数.....	92
5.3.3 实训练习.....	93
5.4 实训项目二.....	95
5.4.1 返回指针的函数.....	95
5.4.2 指针作为函数的形式参数.....	97
5.4.3 实训练习.....	98
5.5 实训项目三.....	99
5.5.1 命令行参数.....	99
5.5.2 实训练习.....	101
习题.....	102
第 6 章 结构体与共用体.....	106
6.1 实训知识准备.....	106
6.1.1 结构体.....	106
6.1.2 共用体.....	109
6.2 实训案例.....	109
6.3 实训项目一.....	112
6.3.1 结构体变量.....	112
6.3.2 结构体数组.....	113
6.3.3 指向结构体的指针.....	114
6.3.4 结构体与函数.....	115
6.3.5 实训练习.....	116
6.4 实训项目二.....	117
6.4.1 建立与输出静态链表.....	117
6.4.2 链表的动态存储分配.....	118
6.4.3 实训练习.....	119
6.5 实训项目三.....	120
6.5.1 共用体变量.....	120
6.5.2 结构体和共用体的嵌套使用.....	121
习题.....	122
第 7 章 文件.....	128
7.1 实训知识准备.....	128
7.1.1 使用数据文件的优点.....	128

7.1.2 文件操作时应注意的要点.....	129
7.2 实训案例.....	130
7.3 实训项目一.....	132
7.4 实训项目二.....	134
7.4.1 结构体的读写.....	134
7.4.2 文件的格式读写.....	136
7.4.3 实训练习.....	137
7.5 实训项目三.....	138
7.5.1 文件的随机读写.....	138
7.5.2 实训练习.....	141
习题.....	142
第8章 C 程序设计大型作业.....	146
8.1 实训知识准备.....	146
8.1.1 基本知识.....	146
8.1.2 软件测试.....	146
8.1.3 测试用例设计技术.....	147
8.1.4 程序维护.....	147
8.2 实训项目.....	147
8.2.1 实训题.....	147
8.2.2 实训要求.....	147
8.2.3 系统设计.....	148
8.3 源代码参考.....	151

第 1 章 C 程序设计基础

实训目的

通过本章的实训，了解 C 程序的编写与上机调试过程，熟悉 Turbo C 和 Visual C++ 6.2 的集成环境，掌握程序编辑、编译、运行的方法，重点掌握数据基本类型、变量、函数、表达式的概念，掌握 C 程序的输入函数、输出函数、赋值语句、间接访问的使用方法。

实训内容

本章主要介绍 C 程序设计基础知识，内容包括数据基本类型、变量、函数、表达式的概念和输入函数、输出函数、赋值语句、间接访问的使用方法以及使用 Turbo C 和 Visual C++ 6.2 的集成环境进行程序编辑、编译、运行的方法。

1.1 实训知识准备

1.1.1 C 语言特点

C 语言具有如下几个特点。

- C 程序由函数构成，函数分为 main 函数、库函数和自定义函数三类。
- 函数=函数首部+函数体，其中函数体有声明和执行两部分。
- 函数体声明部分是对程序中使用的变量、数组、函数等进行数据类型说明。
- 执行部分是 C 语言语句，C 语言语句有控制语句、函数调用语句、表达式语句、空语句和复合语句五类，每个语句由分号结束。
- C 程序中字母的大小写是有区别，语句中通常使用小写字母。

1.1.2 C 程序的数据类型

- | | | | |
|--------|---|-----|------------------------------------|
| ① 基本类型 | { | 整型 | int, long, unsigned, unsigned long |
| | | 实型 | float, double |
| | | 字符型 | char |
| | | 枚举 | |
| ② 构造 | { | 数组 | |
| | | 结构体 | |
| | | 共用体 | |
| ③ 指针类型 | | | |
| ④ 空类型 | | | |

1.1.3 常量与变量

在程序运行过程中，其值不能被改变的量称为常量，其值可以被改变的量称为变量。

常量可分为整型（如 1, 16L）、实型（如 -2, 3.1e-5）、字符型（如 'B', '\n'）等类型，另外可在程序中使用符号常量，如：`#define PI 3.14159`。

一个变量在内存中占据某一个存储单元。变量名表示存储单元的值，运算符 & 可取得存储单元的地址。变量要先“先说明，后使用”，变量说明语句：

数据类型 变量名表；

变量用标识符（以字母、数字、下划线组成且第一个字不可为数字）表示。可以通过初始化、赋值语句、输入函数，使变量获得其具体数据。

注意 整型变量相除，结果为一个整型数。字符型变量在内存中是将相应的 ASCII 码放到存储单元中。实型数据因为有效位而存在误差，如 $1.0/3*3$ 结果不等于 1。

1.1.4 输入/输出函数

格式化输出函数：

`printf`（格式控制，输出项表）

输出数据时可以加入普通字符作为提示，格式字符与输出项的类型要相符合，如 `printf("x=%4d,y=%.2fn",x,y+z);`

格式化输入函数：

`scanf`（格式控制，输入项地址表）

在输入数据时，空格、Tab 键、回车都表示一个数据结束，输入数据要严格按格式控制的要求进行。如 `scanf("%d%d",&x1,&x2);`

字符输出函数：

`putchar`（字符型变量）

字符输入函数：

字符型变量=`getchar`（）

1.1.5 自定义函数

自定义函数和变量一样，在其主调函数中也必须“先说明，后使用”。函数说明语句：
数据类型 函数名（形参表）；

有返回值的函数一般用表达式语句调用，无返回值的函数用函数调用语句调用。

所有函数都是互相平行的，所以自定义函数要在函数外部独立定义。函数定义格式：

数据类型 函数名（形参表）

{函数体（说明部分+语句）}

函数的返回值通过函数体中的 `return` 语句获得，返回值的类型与函数定义的类型一致。

1.1.6 运算符与表达式

运算符要注意优先级，常用的运算符优先次序：

括号>单目运算>算术运算>关系运算>逻辑运算>三目运算>赋值运算>逗号运算。

注意 单目运算、三目运算、赋值运算的结合方向是右至左。

运算符将操作数连接成表达式、因此有算术表达式、关系表达式、逻辑表达式、条件表达式、赋值表达式和逗号表达式等。

1.1.7 数组

数组是一组有序数据的集合，数组中每一个元素的类型相同。用数组名和下标来惟一确定数组中的元素。C 语言规定数组下标从 0 开始，一般不能引用整个数组，只能逐个引用元素。数组也要先定义后使用，一维数组的说明语句：

数据类型 数组名 [元素个数]；

1.1.8 间接访问

变量的地址称为指针，指针变量是用来存放地址的变量。

C 语言允许通过实体（变量、函数、数组等）的名字或实体的地址来引用一个实体在内存中存放的数据。通过实体名或实体地址访问实体的方法称为直接访问；先将实体的地址放在一个变量中，通过该变量的值（即地址）来访问实体的方法称为间接访问。

如有定义

```
int x,y,*p;
```

则

```
x=8; /* 通过变量名直接访问 x */
scanf("%d",&y); /* 通过变量地址直接访问 y */
p=&x;
printf("%d",*p); /* 间接访问 x */
```

1.2 实训案例

案例 1.1 已知两个电阻分别为 $10\ \Omega$ 、 $20\ \Omega$ ，求并联后电阻值。

主函数的 N-S 图，如图 1.1 所示。

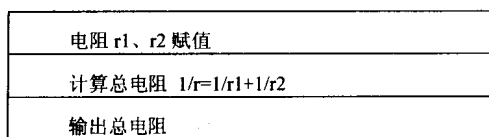
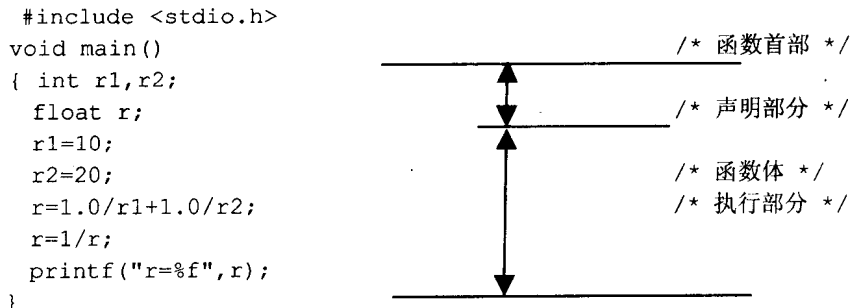


图 1.1 主函数的 N-S 图



该程序由于 r1、r2 已知，所以可用赋值语句给 r1、r2 赋值，注意 r1、r2 是整型变量，所以不能用 1/r1+1/r2（结果为 0）作算术表达式。变量 r 在程序中被赋值 2 次，第二次赋值后，第一次的数据被覆盖。

也可以用初始化方法改写本例：

```
#include <stdio.h>
void main( )
{ int r1=10,r2=20;
  float r;
  r=1.0/r1+1.0/r2;
  printf("r=%f",1/r);          /* 输出项是算术表达式 */
}
```

如果 r1、r2 的值需要从键盘临时输入，本例可改写为：

```
#include <stdio.h>
void main( )
{ int r1,r2;
  float r;
  printf("please input 2 int:\n");
  scanf("%d%d",&r1,&r2);      /* 输入数据时以空格隔开 */
  r=1.0/r1+1.0/r2;
  printf("r=%f.2",1/r);      /* 输出项保留 2 位小数 */
}
```

本例用了两个库函数，scanf() 函数输入数据，printf() 函数输出数据。Printf() /scanf() 存放在“stdio.h”文件中，与其他库函数调用方法不同的是它们可以直接调用而不必使用#include 命令。

1.3 Turbo C 的集成环境

1.3.1 进入 Turbo C

Turbo C 安装完毕后，用户的磁盘（假设 D 盘）上将会增加以下子目录和文件。

- D:\tc: 其中包括 tc.exe、tcc.exe、make.exe 等执行文件。
- D:\tc\include: 其中包括 stdio.h、math.h、string.h 等头文件。
- D:\tc\lib: 其中包括 maths.lib、mathl.lib、graphics.lib 等库函数文件。
- D:\tc\bgi: 其中包括 tc 运行时所需的信息。

可以在桌面上选择“开始→程序→MS-DOS 方式”菜单，进入“DOS 方式”窗口，在该窗口中使用下面 DOS 命令进入 Turbo C 环境。

```
>d: /          设 Turbo C 安装在 D 盘
>cd\tc /      进入 TC 子目录
>tc /         运行 TC 程序
```

Turbo C 的集成环境如图 1.2 所示。



图 1.2 Turbo C 的集成环境

1.3.2 集成环境

(1) 主菜单窗口位于 TC 屏幕的顶部，它有 8 个主菜单。

File	Edit	Run	Compile	Project	Options	Debug	Break/Watch
文件	编辑	运行	编译	项目	选项	调试	断点/监视

每一个主菜单还有其子菜单，分别用来实现各项操作。

(2) 编辑窗口在主菜单窗口的下面，正上方有“Edit”标志。编辑窗口的作用是对 Turbo C 源程序进行输入和编辑。源程序都在这个窗口中显示。在编辑窗口的上部有一行提示

Line 1 Col 1 Insert Indent Tab Fill Unindent C:\NONAME.C

其中 Line 1 和 Col 1 表示当前光标的位置，“Insert”表示处于插入状态，该状态由 Insert 键控制。该行最右端显示的是当前正在编辑的文件名，对新文件自动命名为 noname.c。如果从磁盘调入一个已存在的文件，则在该位置上显示的是该文件的名字。

(3) 信息窗口在屏幕的下部，用来显示编译和连接时的有关信息。在信息窗口上方有“Message”字样作为标志。在编辑源程序时不用此窗口。

(4) 功能键提示行在屏幕的最下方（在信息窗口的下面）。它显示一些功能键的作用。

① F1-Help（帮助），任何时候按 F1 键都会显示帮助信息。

② F5-Zoom（分区控制），如果当前在编辑窗口工作，按 F5 键就不显示信息窗口，它的作用是扩大编辑窗口。若再按一次 F5 键，就会恢复信息窗口。

③ F6-Switch（转换），按 F6 键就激活信息窗口，此时编辑窗口不能工作。若再按一次 F6 键，就又激活编辑窗口，此时可以在编辑窗口中编辑源程序。

④ F7-Trace（跟踪），用于跟踪程序的运行情况。

⑤ F8-Step（按步执行），每按一次 F8 键，执行程序中的一个 C 语句。

⑥ F9-Make（生成目标文件）：对源程序进行编译和连接，生成 OBJ 文件和 EXE 文件，但不进行运行。

⑦ F10-MENU（菜单）：激活主菜单，光标在菜单 File 上（此时 File 以反相显示）。常用菜单见表 1.1。

表 1.1 常用菜单

File	Run	Compile	Options
Load	Run	Compile to OBJ	Compiler
Pick	Program reset	Make EXE file	Linker
New	Go to cursor	Link EXE file	Environment
Save	Trace into	Build all	Directories
Write to	Step over	Primary c file	Arguments
Directory	User screen	Get info	Save options
Change dir			Retrieve options
Os shell			
Quit			

1.3.3 编辑文件

1. 编辑一个新文件

如果要输入和编辑一个新的 C 程序，应该先选主菜单中的 File 菜单（按 Alt+F 键或 F10 键出现下拉菜单），用键盘上的 ↓ 和 ↑ 键找到子菜单项 New，然后按回车键，编辑窗口就被清除干净，光标定位在左上角（第 1 行、第 1 列）。

在 Turbo C 提供的一个全屏幕编辑环境中，用户将已编好的源程序逐行输入，如发现错误可随时修改。输完程序后应对程序作检查，并改正已发现的错误。

2. 编辑一个已存在的文件

假如以前已存盘的源文件需要进行编辑，随时可以把它从磁盘中调出来。选择菜单 File→Load，这时，屏幕上出现一个包括*.c 的“装入文件”对话框，要求用户输入准备调入的文件路径和文件名。

3. 指定工作目录

工作目录是指用户文件所在的目录。为了管理上的方便和安全，应分别建立工作目录：选择菜单 File→Change dir 项并按回车键后，就会出现一个“新目录输入框”，提示用户输入所选择的工作目录名。以后各人可以在各自的子目录中进行源文件编辑，编译生成的目标文件也将会存放在些子目录中。

1.3.4 编译、连接与运行

按 F9 键可以完成对源程序的编译（生成.OBJ 文件）和连接（.OBJ 文件与库函数、包含文件连接生成.EXE 文件）。

编译和连接过程顺利通过，屏幕上出现的编译窗口高亮度的 SUCCESS，这时应及时将源程序保存起来：选择菜单 File→Save 或 Write to；按回车键后，TC 就会弹出一个对话框，要求用户指定文件名。

如果程序有错的话，编译窗口出现高亮度的 ERRORS。出错有两类：警告 Warning（仍可编译但不保证结果正确）和错误 Error（系统不能容任，必须改正才能重新编译连