



教育部职业教育与成人教育司推荐教材  
五年制高等教育公共课教学用书

# C语言程序设计

## (基础版)

谭浩强      主编  
周海燕      主审  
冯玉山  
崔武子 等   编著

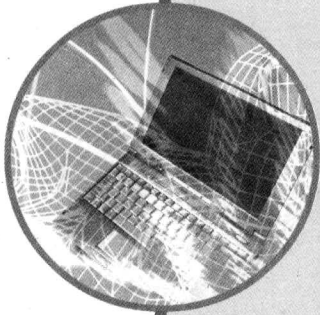


华夏出版社

4  
28

教育部职业教育与成人教育司推荐教材  
五年制高等职业教育公共课教学用书

-97 ③



# C语言程序设计

## (基础版)

谭浩强	主编
周海燕	主审
冯玉山	
崔武子 等	编著

华夏出版社

图书在版编目(CIP)数据

C 语言程序设计/谭浩强主编. - 北京:华夏出版社,2005.6

ISBN 7-5080-3735-9

I. C... II. 谭... 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 060357 号

C 语言程序设计

谭浩强 主编

崔武子等 编著

责任编辑:焦 玉

封面设计:刘 颖

出版发行:华夏出版社

(北京市东直门外香河园北里 4 号 邮编:100028)

经 销:新华书店

印 刷:北京人卫印刷厂

版 次:2005 年 6 月北京第 1 版

2005 年 6 月北京第 1 次印刷

开 本:787×1092 1/16 开

印 张:17.25

字 数:322 千字

定 价:24.00 元

本版图书凡印刷、装订错误,可及时向我社发行部调换

# 序

2004年4~8月,教育部职成司对五年制高职教材重新进行了整体规划,在全面总结吸收“面向21世纪职业教育课程改革和教材建设规划”经验成果的基础上,动员全国各地申报职业教育教材两千余种,经组织专家评审后,制定了《2004—2007年职业教育教材开发编写计划》,这套五年制高职公共课教材,就是按照这个计划编写的。

在编写之前,我们对目前五年制高职公共课的教学情况进行了大量的调查研究,对现行的教材做了深入的分析比较,提出了严缜细密的编写题纲,并上报教育部职成司。在得到职成司有关领导和专家充分肯定后,开始着手这套教材的编写工作。

今年秋季率先推出的是供五年制高职院校秋季入学新生使用的《实用语文》第一册(全4册)、《应用数学基础》第一册(全3册)、《实用英语》第一册(全4册)、《技术物理基础》第一册(全2册)、《计算机应用基础》(全1册)和《应用化学基础》(全1册)。

本套教材的作者一部分是来自五年制高职院校教学一线的教师,一部分是各学科领域的专家学者,他们既具有丰富的教学经验,又都参加过教材的编写工作,具有丰富的教材编写经验。担任各门课程第一主审人的均是该学科领域里的专家,第二主审人则是来自五年制高职教学一线的优秀教师。

针对五年制高职学生目前的生源水平现状,本套教材适当降低了起点和难度,本着“少而精”的原则,使教材的难度深浅适中,既符合学生的实际水平,又加强了教学的针对性,并注意吸收新知识、新观念,强调基础性,突出实用性,体系设计合理,循序渐进,符合学生学习特征和认知规律,结构体例新颖,便于教师和学生使用。

本套教材是根据全国五年制高职教育公共课开发指导委员会《关于编写五年制高职教育公共课规划教材的指导意见》编写的,设计课程容量、课时安排,均考虑了教与学双方面的现实可操作性,让教与学成为一种互动过程,让学生

尽可能地在轻松愉悦中掌握知识。同时,为减轻学生的课业压力,我们把以往教材多配带的《练习册》的内容放到了教材的练习中,让练习成为以点代面、以精带泛的真正切实有效的思训活动。

虽然我们尽了很大努力,但教材中仍难免存在各种缺点、错误和疏漏,敬请广大教学第一线的教师和专家学者们批评指正。同时,随着我国高职教育的发展,教材也要不断发展,不断更新完善。我们将在教材使用过程中不断跟踪反馈意见,不断修订完善,以期把最好的教材奉献给广大师生。

全国五年制高等职业教育公共课开发指导委员会

2005年6月

# 前 言

C 语言程序设计是高等院校普遍开设的一门课程,该课程也是软件工作者必须掌握的一个工具。但由于 C 语言功能多,使用灵活,初学者普遍感到难以掌握。根据这一情况,作者在充分分析本课程特点和学生认知规律的基础上,经过几年多种教学方式的尝试后整理出本书。

本书为五年制高职高专教材,也可用做自学者的指导书。主要对象是 C 语言零起点读者,如果读者已掌握了计算机的初步知识,将会更容易地学习使用本书。

## 本书特点:

### 1. 讲授内容少而精。

为了突出重点,也考虑到各学校的课时限制,因此初学者不常使用的内容这里不做介绍。但对于 C 语言的重点内容则用通俗易懂的方式较详细地予以介绍,而且采用“先设计 C 语言程序框架,后添补具体功能代码”的方式,做到用最短的时间达到能够编写简单应用程序的水平。

### 2. 例题知识点明确。

本书中有大量的例题,对于每一例题,都按问题提出、解题思路、程序代码、运行结果、编程技巧、知识要点等步骤进行介绍,并用矩形框将关键的代码标出来。

### 3. 强调实践中学习。

本书在介绍一些精心设计的内容后,及时安排了课堂实训,其中包括问题提出、实训目标、实训步骤,对于编程题,还包括程序代码。本课程本身就是实践性很强的课程,因此在实践中学习语法、了解概念、掌握算法是学好本课程的最佳途径。

### 4. 逐步并适时介绍调试方法。

调试程序是程序设计的一个重要环节,为了使学生在最需要的阶段学到相应的知识,本书通过具体实例分阶段介绍了调试程序并排除错误的方法。

### 5. 增设了环环相扣、贯穿整体内容的综合实例。

本书给出一个综合实例,其功能从第2章起,随着讲授内容的增多而逐步实现,并在第7章给出完整的程序代码,目的是使学生尽早体会较大应用程序的编写过程。

### 6. 提供独特的索引。

本书可通过附录中提供的索引,按关键字、运算符或库函数快速搜索相关章节或例题,也可以通过附录中各例题的知识要点,搜索相关例题。

### 7. 提供电子教案。

为了提高教师的工作效率和减轻备课的工作量,精心制作了动画丰富、可修改、可直接使用的电子教案。另外,还提供全书所有例题和分阶段介绍的综合实例源代码,它们均在 Turbo C 2.0 中被运行通过。

本书由崔武子主持编写并统稿,朱立平、李红豫、李青和赵重敏老师参加了部分内容的编写和程序的调试。

限于作者水平,书中难免有错误和疏漏之处,恳请读者批评和指正。

编者

2005年5月

## 第1章 C语言的基础知识

- 1.1 认识C语言程序 /3
- 1.2 设计简单C语言程序 /4
- 1.3 选择使用C语言数据类型 /9
  - 1.3.1 常量与变量 /9
  - 1.3.2 整型数据的使用 /10
  - 1.3.3 实型数据的使用 /12
  - 1.3.4 字符型数据的使用 /13
- 1.4 C语言常用运算符与表达式 /15
  - 1.4.1 算术运算符与表达式 /15
  - 1.4.2 赋值运算符与表达式 /17
  - 1.4.3 关系运算符与表达式 /18
  - 1.4.4 逻辑运算符与表达式 /18
- 1.5 控制数据的输入、输出 /20
  - 1.5.1 按格式输入、输出数据 /20
  - 1.5.2 单个字符的输入、输出 /23
- 练习题 /24

## 第2章 程序设计的3种基本结构

- 2.1 结构化程序设计 /27
- 2.2 顺序结构 /28
- 2.3 分支结构 /30
  - 2.3.1 使用if语句解决分支问题 /31
  - 2.3.2 使用switch语句解决分支问题 /39
  - 2.3.3 分支结构应用举例 /44
- 2.4 循环结构 /51
  - 2.4.1 使用while语句解决循环问题 /51
  - 2.4.2 使用do-while语句解决循环问题 /56
  - 2.4.3 使用for语句解决循环问题 /58
  - 2.4.4 嵌套的循环 /64



- 2.4.5 控制循环的执行 /65
- 2.4.6 循环结构应用举例 /68
- \*2.5 一个综合实例的部分程序 /74
- 练习题 /80

### 第3章 使用数组处理大批量数据

- 3.1 使用一维数组解决日常问题 /85
  - 3.1.1 一维数组的定义 /85
  - 3.1.2 一维数组元素的引用 /86
  - 3.1.3 一维数组的应用举例 /88
- 3.2 使用二维数组解决日常问题 /100
  - 3.2.1 二维数组的定义 /101
  - 3.2.2 二维数组元素的引用 /101
  - 3.2.3 二维数组的应用举例 /102
- \*3.3 一个综合实例的部分程序 /106
- 练习题 /122

### 第4章 指针的使用

- 4.1 变量的地址与指针变量 /125
- 4.2 指针变量的定义 /126
- 4.3 使用指针访问变量 /127
- 4.4 使用指针访问数组 /130
  - 4.4.1 使用指针访问一维数组 /130
  - 4.4.2 使用指针访问二维数组 /138
- 4.5 使用指针处理字符串 /140
  - 4.5.1 字符串的概念 /140
  - 4.5.2 调用函数处理字符串 /144
- \*4.6 一个综合实例的部分程序 /147
- 练习题 /154

## 第5章 自定义函数

- 5.1 函数的定义与调用 /157
  - 5.1.1 函数的定义 /158
  - 5.1.2 函数的调用 /161
  - 5.1.3 函数的调用过程 /164
  - 5.1.4 函数的原型说明 /166
  - 5.1.5 函数的返回值 /167
- 5.2 指针与函数调用 /170
- 5.3 数组与函数调用 /174
  - 5.3.1 一维数组名作实参 /174
  - 5.3.2 二维数组名作实参 /177
- 5.4 字符串与函数调用 /179
- 5.5 函数的嵌套调用 /182
- 5.6 变量的存储类别 /184
  - 5.6.1 内部变量和外部变量 /184
  - 5.6.2 动态存储变量和静态存储变量 /186
- \*5.7 一个综合实例的部分程序 /188
- 练习题 /199

## 第6章 结构体

- 6.1 结构体类型的概念和声明 /203
- 6.2 结构体变量的定义和结构体成员的引用 /204
  - 6.2.1 结构体变量的定义与初始化 /204
  - 6.2.2 结构体成员的引用 /205
- 6.3 结构体数组及其应用 /212
- \*6.4 一个综合实例的部分程序 /221
- 练习题 /229

## 第7章 文本文件中数据的读写

- 7.1 文本文件的概念 /233
- 7.2 文本文件的基本操作 /233
- 7.3 文本文件的应用举例 /238
- \*7.4 一个综合实例的完整程序 /244
- 练习题 /253

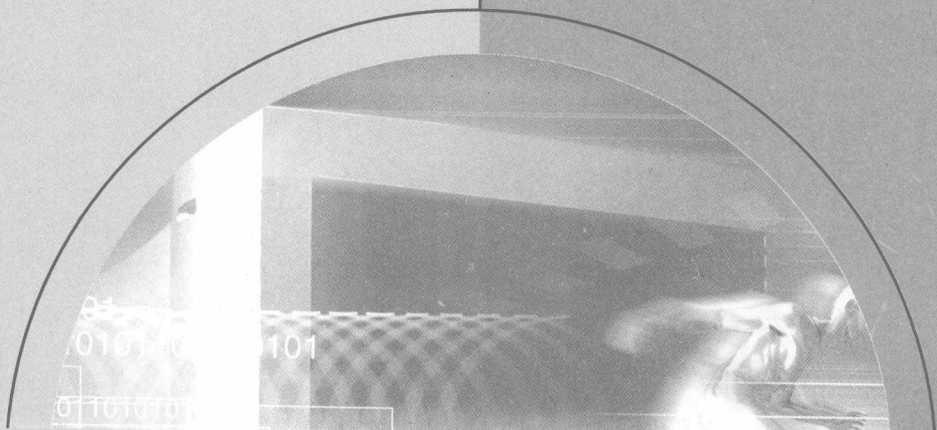
## 附录

- 附录 I C 语言关键字 /254
- 附录 II 常用字符与 ASCII 代码对照表 /254
- 附录 III 常用运算符的优先级和结合方向 /255
- 附录 IV 常用 C 语言库函数 /256
- 附录 V 常见错误、警告信息表 /258
- 附录 VI 关键字、运算符、库函数索引 /260
- 附录 VII 各章例题知识要点 /261

## 参考文献

# 第 1 章 C 语言的基础知识

- 1.1 认识C语言程序
- 1.2 设计简单C语言程序
- 1.3 选择使用C语言数据类型
- 1.4 C语言常用运算符与  
表达式
- 1.5 控制数据的输入、输出





## 1.1 认识 C 语言程序

下面通过一个完整的 C 语言程序,说明 C 语言程序的构成特点和程序的执行过程。程序中的每一部分内容将在后续章节中详细介绍。

例 1.1 一个完整的 C 语言程序示例:

```

#include <stdio.h>
#include <math.h>
#define PI 3.14159

float myarea(float r);

main()
{ float a = -20,b,c;
  b = fabs(a);
  c = myarea(b);
  printf("c=%f\n",c);
}

float myarea(float r)
{ float s;
  s = PI* r* r;
  return s;
}

```

/\* ①此块是预处理命令部分 \*/

/\* ②此块是 myarea 函数的原型说明部分 \*/

/\* ③此块是主函数部分 \*/

/\* ④此块是 myarea 函数的定义部分 \*/

运行结果:

```
c=1256.635986
```

说明:

(1) 一个 C 语言程序可由若干个函数构成,本程序包括两个函数:主函数(块③)和 myarea 函数(块④)。

(2) 任何一个 C 语言程序都包含一个主函数(函数名必须是 main),而且只能包含一个主函数。至于其他函数根据实际情况可

多可少(本程序功能少,所以只包括一个)。

(3)C 语言提供很多标准库函数,可以直接调用这些函数,但在程序的开头要加上包含该函数信息的命令行。例如,程序中使用了标准库函数 `printf`,其功能是在屏幕上显示运行结果,这是输出函数,C 语言系统将所有输入、输出函数的信息存放在“`stdio. h`”文件中,所以程序的开头加了 `#include <stdio. h>`。再如,程序中使用了标准库函数 `fabs`,其功能是计算绝对值,这是数学函数,C 语言系统将所有数学函数的信息存放在 `math. h` 文件中,所以程序的开头又加了 `#include <math. h>`(块①)。

(4)虽然 C 语言系统提供非常丰富的标准库函数,但在实际开发中仅有这些是不够的,这就需要编程者花很多时间自己编写大量的函数。程序中 `myarea` 函数就是我们自己编写的函数,该函数的功能是计算半径为  $r$  的圆面积。主函数调用了该函数,在调用自己定义的函数前,要对该函数进行函数原型说明(块②)。我们将在第 5 章详细介绍函数的定义与调用方法。

(5)程序的开头有了命令行 `#define PI 3. 14159` 后,程序中所有 `PI` 均用 `3. 14159` 代替。

(6)任何一个 C 语言程序都从主函数开始执行。如果在主函数里从上到下按顺序执行各条语句的过程中,遇到调用其他函数的语句(如“`c = myarea(b);`”),则程序执行流程转去调用该函数实现功能后回到主函数,并继续执行其余的语句。程序执行过程的详情参见 5. 1. 3 节。

(7)程序中有 4 个用“`/* */`”括起来的部分,此部分是注释部分,它只起解释的作用,对程序的运行没有任何影响,在编写程序时加注释只是为了方便阅读程序。

知识要点:

认识 C 语言程序。

## 1.2 设计简单 C 语言程序

前面通过例 1.1 介绍了 C 语言程序的基本构成。在 C 语言中最简单的程序是只包含主函数的程序。下面设计一个简单的 C 语言程序,通过该程序了解主函数的框架和 Turbo C 环境的使用。

**例 1.2** 编写程序,在屏幕上先显示提示信息“Enter x and y:”,当用户输入 x 和 y 的值后,输出这两个数的差。

**解题思路:**显示提示信息需要调用标准库函数 printf,而输入数据的功能则要调用标准库函数 scanf 实现。本题目需要将输入的两个数存放在 x 和 y 中,为了完成此操作,在程序中制造两个容器(即定义两个变量 x,y)。程序还要制造一个放两个数差值的容器(即定义存放差值的变量 z)。输出运算结果也要调用 printf 函数。

**程序代码:**

```
#include <stdio.h>          /* 调用库中标准输入、输出函数时加此行 */
main()
{
    int x,y,z;              /* 定义3个变量 x,y,z */
    printf("Enter x and y:"); /* 显示提示信息 */
    scanf("%d%d",&x,&y);    /* 输入的两个数分别存放在 x 和 y 中 */
    z=x-y;                  /* 计算 x,y 的差值,并存放在 z 中 */
    printf("The difference between %d and %d is %d.\n",x,y,z); /* 输出差 */
}
```

**运行结果:**

```
Enter x and y:3 6<回车>    (本书将用户输入的内容用下划线标出)
The difference between 3 and 6 is -3.
```

**编程技巧:**

(1)搭建主函数的框架,并根据该框架逐项填补。

**主函数的一般框架:**

```
main()
{
    定义变量部分
    实现功能部分
}
```

大括号中的内容为函数体。函数体包括两个部分,在“定义变量部分”中,定义程序要用到的所有变量,C语言规定变量必须先定义后使用。有关变量将在 1.3.1 节详细介绍。在“实现功能部分”中,编写本题目要实现的功能代码。本题目只需要使用 3 个变量,所以定义了 3 个;要实现的功能包括显示提示信息、输入数据、计算差值、输出计算结果,所以用了 4 条语句完成。

(2)在屏幕上显示内容要调用 printf 函数。程序中语句“printf(“Enter x and y:”);”的功能是显示双引号中的所有字符“Enter x and y:”,而语句“printf(“The difference between %d and %d is %d.\n”,x,y,z);”的功能是在显示双引号中字符的同时,3 个“%d”的位置上依次用 x, y 和 z 的值替换。“%d”是格式说明符,



在输出整数时使用，“\n”是换行符。我们将在 1.5.1 节详细讨论 printf 函数。

(3) 从键盘输入数据要调用 scanf 函数。程序中语句“scanf(“%d %d”, &x, &y);”的功能是要求用户从键盘输入两个整数, 并把它们分别存放在变量 x 和 y 中。在输入整数时要使用格式说明符“%d”。有关 scanf 函数的介绍参见 1.5.1 节。

(4) 编写程序时应尽量多加一些注释, 这会对阅读程序带来很大的方便。

(5) 编写程序后一定要通过上机验证程序的正确性, 程序中的有些错误特别隐蔽, 用眼睛有时检查不出来。

社会上流行多种 C 语言程序开发环境, 本书选用 Turbo C 2.0 集成环境。Turbo C 集成环境是一个编译效率高、运行速度快、使用方便的综合性软件。在这里我们可以非常方便地进行编辑、编译、连接和运行任何一个 C 语言的操作。下面通过一个例题简单介绍 Turbo C 集成环境(以后随着所学知识的增多, 将逐步补充介绍其他操作方法):

(1) 安装 Turbo C 2.0 集成环境的方法。将 Turbo C 2.0 直接复制到 C 盘 TC 文件夹中即可完成安装。

(2) 启动 Turbo C 2.0 集成环境的方法。双击 TC.EXE 文件即可。Turbo C 2.0 集成环境如图 1.1 所示。

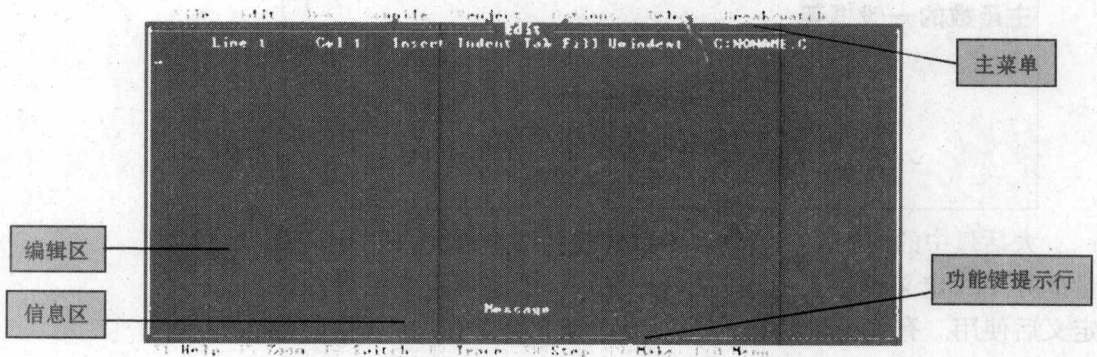


图 1.1 Turbo C 2.0 集成环境

(3) 编辑程序的方法。编辑程序是在编辑区中输入或修改 C 语言源程序的过程。输入程序时可以观察到 Line 和 Col 分别跟踪显示的光标所在行号和列号。编辑区中的 Insert 表示插入或改写状态, 可通过键盘上的 Insert 键切换。在编辑源程序时, 经常使用的快捷键和块操作命令如表 1.1 所示。