

21世纪高职高专系列规划教材

C语言程序设计与实践

主 编 段红玉



北京师范大学出版集团
BEIJING NORMAL UNIVERSITY PUBLISHING GROUP
北京师范大学出版社

21世纪高职高专系列规划教材

C语言程序设计与实践

主 编 段红玉

副主编 王 辉 陈炎龙

常州大学图书馆
藏书章



北京师范大学出版集团
BEIJING NORMAL UNIVERSITY PUBLISHING GROUP
北京师范大学出版社

图书在版编目(CIP) 数据

C 语言程序设计与实践 / 段红玉主编. —北京: 北京师范大学出版社, 2010.8

21 世纪高职高专系列规划教材

ISBN 978-7-303-11443-6

I. ① C… II. ①段… III. ① C 语言—程序设计—高等学校: 技术学校—教材 IV. ① TP312

中国版本图书馆 CIP 数据核字 (2010) 第 162138 号

出版发行: 北京师范大学出版社 www.bnup.com.cn

北京新街口外大街 19 号

邮政编码: 100875

印 刷: 北京京师印务有限公司

经 销: 全国新华书店

开 本: 184 mm × 260 mm

印 张: 16.75

字 数: 277 千字

版 次: 2010 年 8 月第 1 版

印 次: 2010 年 8 月第 1 次印刷

定 价: 28.00 元

策划编辑: 庞海龙 责任编辑: 庞海龙

美术编辑: 高 霞 装帧设计: 华鲁印联

责任校对: 李 茵 责任印制: 张 坤

版权所有 侵权必究

反盗版、侵权举报电话: 010-58800697

北京读者服务部电话: 010-58808104

外埠邮购电话: 010-58808083

本书如有印装质量问题, 请与印制管理部联系调换。

印制管理部电话: 010-58800825

出版说明

高等职业教育是新世纪我国高等教育大众化进程中的一个亮点，正由规模扩张转向内涵发展。高等职业教育内涵发展的核心是课程建设。只有有一套充分体现高等职业教育规律、符合高职学生学习特点以及与职业岗位或职业岗位群相匹配的课程体系，才能有效发挥高等职业教育的特长，为社会各行各业培养具备全面素质和良好综合职业能力的高层次、应用型人才。

北京师范大学出版社是教育部职业教育教材出版基地之一，有着二十余年的职业教育教材出版历史，积累了丰富的高等职业教育教材编辑出版经验。近年来，在教育部高等教育司、职业教育与成人教育司以及北京师范大学的支持下，北京师范大学出版社汇聚教育界、出版界的专家及高等职业院校的优秀教师组建了“全国职业教育教材改革与出版领导小组”，具体负责指导职业教育教材研发工作，以为高等职业教育的课程建设贡献一份力量。目前，我社按照“就业导向、能力本位、任务驱动”等职业教育新理念的要求，研发了高职高专文化基础课、专业主干课教材近二百个品种，其中近30种被列为国家级“十一五”普通高等教育规划教材。这些教材具有如下特点：

1. 紧密结合高等职业教育改革与发展的需求。这批教材依据教育部或相关行业协会颁布的课程标准或教学纲要，针对高等职业教育的培养目标，以就业导向、能力本位为指导，以综合职业能力培养为重点，以为学生职业生涯发展服务为目的，设计教材体系、选择教材内容，体现出先进性、科学性和时代性的特点。

2. 针对高职学生的学习特点精心设计教材的栏目。这批教材注重学生学习兴趣的激发，在表现形式上力求灵活多样、新颖精致，既体现教材内容的特点，又与高职高专院校学生的学习习惯、认知能力和相应的职业岗位群的要求相适应。各书有选择地设计了以下栏目：

学习目标：简明扼要地指出各章的学习方向，引导学生有的放矢地学习。

案例分析：以实例创设学习情境，引导学生学习新知识，形成新技能。

提个醒：告诉学生在学习相关内容的过程中应注意的问题，以提高学习的效率和效益。

小思考：用有趣而有效的问题，启迪学生的思维。

小资料：提供相关材料或背景资料，拓展学生的视野。

小知识：生动而有趣的知识点，帮助学生吃透学习内容，增强学习兴趣。

本章小结：概括本章的主要内容，有助于学生从整体上把握知识结构和复习巩固所学内容。

思考与练习：精心设计各种类型的练习题，供学生复习、实践使用，以全面提升学生的综合能力。

3. 紧密结合行业发展动态。这批教材充分吸收了行业的新知识、新技术、新工艺、新规范，并注重根据行业的发展及时更新教材的内容，突出教材的职业性与实践性。

4. 形成了立体化、网络化的资源。我们在组织教材研发的过程中，配套研发了电子教案、课件或实验、实习指导材料等。

综合看，这些教材理念先进、内容丰富、形式新颖、语言通俗，注重理论知识的“必需、够用”，更强化以实践能力、创新能力为重点的综合职业能力的培养。

高职高专教材建设是一项复杂的、系统的工作。我们将在未来的日子里，与高等职业教育的改革同行，致力出版精品教材，服务并促进高等职业教育的发展。

全国职业教育教材改革与出版领导小组
北京师范大学出版社

前 言

C语言是目前最流行的面向过程的程序设计语言之一，具有简洁、紧凑、灵活、实用、高效、可移植性好等优点，深受广大用户欢迎。C语言数据类型丰富，既可以用来编写系统程序，又可以用来编写应用程序，因此学习和使用C语言已经成为广大计算机人员和学生的迫切需求。

C语言是计算机各专业的必修课程，也是高职高专院校非计算机专业开设的程序设计课程之一。作为计算机类的专业基础课，课程目的是使学生掌握程序设计的基本方法并逐步形成正确的程序设计思想，能够熟练地使用C语言进行程序设计并具备调试程序的能力，为后续课程及其他程序设计课程的学习和应用打下基础。对于非计算机专业来说，该课程有实际应用价值，为用计算机解决实际问题提供了方法，是后续理论和实践教学的基础和重要工具，同时也是全国计算机等级考试所统一要求的课程之一。

本书共有8个项目，每个项目包括“项目导读”、“学习目标”、“问题提出”、“安排任务”、“思路指导”、“解决方案”、“归纳分析”、“技能训练”和“实验”等模块。“学习目标”部分明确学习目标，给出每一项目应该掌握的内容，以应用任务带动知识点的学习；“解决方案”主要给出本项目所安排的任务的具体解决方案；“归纳分析”部分主要介绍在解决此任务的过程中所应用的具体知识点；“技能训练”进一步针对所学知识，强化其方法和要点的掌握；“实验”模块给出有针对性、综合性较强的实践应用题目，对所学知识进行实践检验。因此，本书在编写过程中，着力突出以下特色。

1. 采用启发式教学思想来安排每一个项目。为培养学生逻辑思维 and 创新能力，在每一个项目安排的过程中，都遵循启发式教学原则，通过各种教学方式和手段激发学生的学习欲望，让学生能积极投入到学习活动中，积极思维，发现问题，提出问题，并逐步解决问题。

2. 内容灵活，实例丰富。从问题提出到解题思路指导，从程序代码编写到程序运行结果分析，对整个程序设计过程都做了详细的讲解。书中有的任务的解决方法不止一个，可以使读者开拓思路，培养创新精神，学习更加灵活，做到举一反三，为开发实际项目打好基础。每个项目后的实验内容为读者对知识的学习和巩固提供了保障。

3. “递进式”的讲解思路，设计独特。课堂教学内容采用问题导入式，每节课都从几个任务入手，在复习以前讲解过的内容基础上，以问题的方式导入新的内容。从而能够有效的激发学生的学习兴趣。精心安排实用性强、趣味性高的任务，在任务中导入问题，在问题的解决中归纳总结，然后再扩展问题、完善解决方案，层层递进，步步深入，逐步完善，不断优化，从而把一个个的知识点消化在一个个问题的解决中。

参与本书编写的教师，都是多年来从事C语言程序设计课程教学的一线教师，他们积累了丰富的知识和教学经验，书中的许多学习方法都是他们教学经验的总结。本

书由段红玉担任主编，王辉、陈炎龙担任副主编，刘灿、王自力、张乐乐、王听忠参与了教材的编写工作。

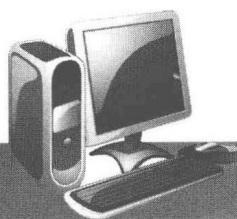
由于编者水平有限，书中疏漏和不足之处在所难免，敬请有关专家和广大读者不吝指正，编者的电子邮箱是 dhy0371@sohu.com。

目 录

项目 1	掌握 C 语言的基本结构	1
任务 1	理解 C 语言的基本结构	2
任务 2	C 程序的编译连接与运行的过程	5
任务 3	选择合理的数据类型	13
任务 4	选择正确的整型数据	18
任务 5	选择正确的实型数据	22
任务 6	选择正确的字符型数据	24
任务 7	使用正确的算术运算符	28
任务 8	使用正确的赋值运算符	33
实验 1		35
项目 2	顺序程序设计	37
任务 1	格式输出语句——设计菜单	38
任务 2	格式输入语句——解决一些简单的数学问题	39
任务 3	交换数据	45
任务 4	格式控制——设计成绩报表	47
任务 5	编写一个体重测量仪	51
实验 2		53
项目 3	选择结构程序设计	55
任务 1	双分支选择语句——体重测量仪改进版（一）	56
任务 2	单分支选择语句——简单排序	60
任务 3	逻辑运算符——电梯控制程序	62
任务 4	合理选择单分支选择语句和双分支选择语句——身高预测	66
任务 5	多分支选择语句 if-else——体重测量仪改进版（二）	68
任务 6	嵌套 if 语句——求解分段函数	72
任务 7	switch 语句——模拟自动饮料机	75
任务 8	switch 语句中多个 case 语句相同情况的处理	81

实验 3	84
项目 4 循环结构程序设计	86
任务 1 while 语句——超市收费系统	87
任务 2 for 语句——神探抓贼	91
任务 3 do-while 语句——密码验证	96
任务 4 break 语句和 continue 语句	102
实验 4	104
项目 5 数组	107
任务 1 定义与引用一维数组——求超市商品利润	108
任务 2 数据统计——餐饮服务质量调查打分	112
任务 3 数据查询	118
任务 4 数据修改	120
任务 5 数据插入与删除	123
任务 6 数据排序	125
任务 7 二维数组	128
实验 5	132
项目 6 函数	137
任务 1 函数的定义	138
任务 2 函数的调用	140
任务 3 函数的嵌套调用和递归调用	151
任务 4 调用函数处理数组中的数据	152
任务 5 函数和常用算法综合编程——成绩排名次	156
实验 6	161
项目 7 结构体	163
任务 1 结构体变量的定义和引用	164
任务 2 结构体数组的定义和初始化	167
任务 3 结构体指针变量的定义与使用	170
任务 4 链表的定义和使用	172
任务 5 链表的插入操作	176
任务 6 链表的删除操作	178
实验 7	182

项目 8	指针	183
任务 1	通过指针访问普通变量	184
任务 2	通过指针访问数组	186
任务 3	通过指针访问字符串	192
实验 8		199
附 录		201
附录 A	ASCII 码	201
附录 B	Turbo C 2.0 集成开发环境的使用	203
附录 C	文本编辑器使用	209
附录 D	Turbo C 编译出错信息	218
附录 E	C 语言库函数	231
附录 F	C 语言程序设计与实践课程设计指导书	244
附录 G	两个经典案例源代码	247
	主要参考文献	257



项目1

掌握 C 语言的基本结构

项目导读

通过 C 程序实例分析着手，使读者从宏观上整体把握和了解 C 程序的基本结构、C 程序的开发调试过程，并且理解 C 语言的基本数据类型的表示形式、存储格式及相关的运算，掌握变量定义的方法及意义，能够灵活运用各种表达式，为 C 语言编程打下基础，对 C 程序开发设计有较深入的感性认识，进而强化对计算机语言和程序设计的理解，为进一步学习 C 语言程序设计打下基础。

学习目标

1. 理解 C 程序的基本结构。
2. 掌握 C 程序的编译连接与运行的过程。
3. 理解常量和变量的含义。
4. 熟悉基本的数据类型。
5. 掌握 C 语言各种运算符和表达式的应用。



任务 1 理解 C 语言的基本结构

问题提出

编写程序需要遵循一定的格式吗？是什么样的格式呢？

安排任务

阅读加法器源程序，掌握其基本结构。

# include <stdio.h>	/* 预处理命令 */
int sum(int x,int y)	/* 定义 sum 函数 */
{ int z;	/* 声明定义变量 z */
z= x+ y;	/* 计算 z 等于 x 与 y 的和值 */
return(z); }	/* 将 z 的值返回,通过 sum 带回调用处 */
main()	/* 主函数 */
{int a,b,s;	/* 声明定义变量 a,b 和 s */
printf("请输入加数和被加数\n");	/* 提示输入两个数 */
scanf("% d% d",&a,&b);	/* 输入变量 a 和 b 的值 */
s= sum(a,b);	/* 调用 sum 函数,将得到的值赋给 s */
printf("s= % d\n",s);}	/* 输出变量 s 的值 */

程序运行结果如下：

```
请输入加数和被加数
58 90
s=148
```

归纳分析

1. 程序基本结构

从这个加法器程序中，我们可以把 C 程序的一般结构用一个表格表示出来，如表 1.1 所示。



表 1.1

预处理命令			# include <stdio.h>
自定义函数 sum。除了主函数外，在使用其他自定义函数前，应对这些进行函数原型说明			int sum(int x,int y)
一对大括号内的是函数体，计算两个数之和			{int z;z= x+ y;return(z);}
主 函 数	函数首部		main()
	函 数 体	函数体开始	{
		定义变量 声明语句：定义三个变量	int a, b, s;
		输出“请输入加数和被加数：”	printf ("请输入加数和被加数\n");
		通过键盘输入两个数	scanf ("% d% d", &a, &b);
		把求得的两个数赋给变量 s	s= sum (a, b);
		输出和值 s	printf ("s= % d\n", s);
		函数体结束	}

2. # include <stdio.h>

这一行是预处理命令，以 # 开头的语句我们称为预处理命令，我们也可以写成 #include “stdio.h”，通常还会有 #include “math.h” 和 #include “string.h” 等。通过观察这个程序可以看到在程序中我们用到了库函数 printf（输出函数）和库函数 scanf（输入函数），C 语言系统把所有的输入输出函数的信息存放在 #include <stdio.h> 文件中，所以我们在用这两个函数的时候要用预编译命令 #include 将 stdio.h 这个“头文件”（stdio.h、math.h 和 string.h 等文件通常放在程序的开头被我们称为头文件，扩展名 h 是 head 的缩写）包含到用户源文件中。C 语言提供了丰富的标准库函数可以让我们直接使用，但是在使用之前我们必须在程序的开头加上预处理命令。

3. 函数

通过观察这个程序大家可以看到有 main 函数、sum 函数、printf 函数和 scanf 函数，所以我们说函数是 C 语言程序的基本单位，并且一个 C 程序必须要有一个 main 函数。在程序运行的过程中一个 C 程序必须从 main 函数开始（不管 main 函数是放在什么位置）并且以 main 函数结束。main（）函数可以返回一个值，也可以不返回值。如果某个函数没有返回值，那么在它的前面有一个关键字 void。

4. C 程序书写格式

- 1) C 程序的每一条语句都必须以分号“;”结束。
- 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。



3) 为避免遗漏必须配对使用的符号, 如注释符号、函数体的起止标识符(花括号)圆括号等。

4) 关键语句要使用注释, C 系统不对注释符进行编译。C 语言的注释格式为: `/* ... */`。“`/`”和“`*`”必须成对使用, 且“`/`”和“`*`”以及“`*`”和“`/`”之间不能有空格, 否则都出错。注释的位置, 可以单占 1 行, 也可以跟在语句的后面。

5) 用 `{}` 括起来的部分, 通常表示程序的某一层次结构。`{}` 一般与该结构语句的第一个字母对齐, 并单独占一行。

6) 低一层次的语句或说明可比高一层次的语句或说明缩进若干格后书写。以便看起来更加清晰, 增加程序的可读性。

■ 知识扩展

尝试把加法器程序改写成只有主函数的实现方式。

```
# include <stdio.h>           /* 预处理命令 */
void main()                   /* 主函数 */
{int a,b,sum;                 /* 声明定义变量 a,b 和 sum */
    printf("请输入加数和被加数\n"); /* 提示输入两个数 */
    scanf("% d% d",&a,&b);      /* 输入变量 a 和 b 的值 */
    sum= a+ b;                /* 计算 sum 等于 a 与 b 的和值 */
    printf("sum= % d\n",sum); } /* 输出变量 sum 的值 */
```

程序运行结果如下:

```
请输入加数和被加数
58 90
s= 148
```

本程序只有主要函数, 主函数的一般框架如下。

```
main ()
```

```
{
```

定义变量部分: 在这部分中定义所用到的变量, 如果需要, 也要对所调用的函数进行声明。

功能语句部分: 由若干个语句组成。

```
}
```

■ 技能训练

分析下面程序的结构: 输入两个整数, 计算两者较大的数, 并输出, 程序如下。



```

void main()                                /* 主函数 */
{
    int a,b,c;                             /* main 函数体开始 */
    scanf("% d,% d",&a,&b);               /* scanf 是输入函数,& 是“取地址” */
    c= max(a,b);                           /* 调用 max,将调用结果赋给 c */
    printf("max= % d\n",c);               /* 输出 c 的值 */
}                                           /* main 函数体结束 */
int max(int x,int y)                       /* 自定义函数,计算两数中较大的数 */
{
    int z;                                /* max 函数体开始 */
    if(x> y)z= x;                          /* 声明部分,定义变量 */
    else z= y;
    return z;                             /* 将 z 值返回,通过 max 带回调用处 */
}                                           /* max 函数体结束 */

```

任务 2 C 程序的编译连接与运行的过程

问题提出

编写好的一个程序如何运行呢?

安排任务

编译连接与运行如下的 C 程序。

```

# include <stdio.h>
void main(){
printf("          \\\ - - //          \n");
printf("          ( @ @ )          \n");
printf(" ┌───────────oOOo- ( _)- oOOo┐          \n");
printf(" | 大家好;                                |          \n");
printf(" | C 语言的学习现在开始了。              |          \n");

```



```

printf(" |                                0ooo |                \n");
printf(" |_____oooO_____( )_|                \n");
printf("                ( ) ) /                \n");
printf("                \\\( ( _/                \n");
printf("                \\\_)                \n");
printf(" * /.      .      *      .                \n");
printf(" .\\*      .      []      *      _                \n");
printf(" * /.      ./\\~ ~ ~ ~ ~ ~ ~ ~ ~ ~ ~ '\\. | ◆                \n");
printf(" \\* , / , . , \\ , ..... , \\ . ◆                \n");
printf(" ||      | # | 田 田 | | | ◆                \n");
printf(" || |      |      | ' | | o                \n");
printf(" || # # ■■■■■■■■■■■■■■■■■■■■■■=                \n");

```

解决方案

1. 启动 Visual C++ 6.0 环境

方法：单击“开始”→“程序”→“Microsoft Visual studio 6”→“Microsoft Visual C++ 6.0”命令，启动 Visual C++。

2. 编辑源程序文件

(1) 建立新工程项目

1) 单击“文件”→“新建”弹出“新建”对话框。

2) 选择“工程”选项卡→单击“Win32 Console Application”→在“工程”文件框中输入项目名，如“wx”，在“位置”框输入或选择新项目所在位置→单击“确定”按钮，如图 1.1 所示。

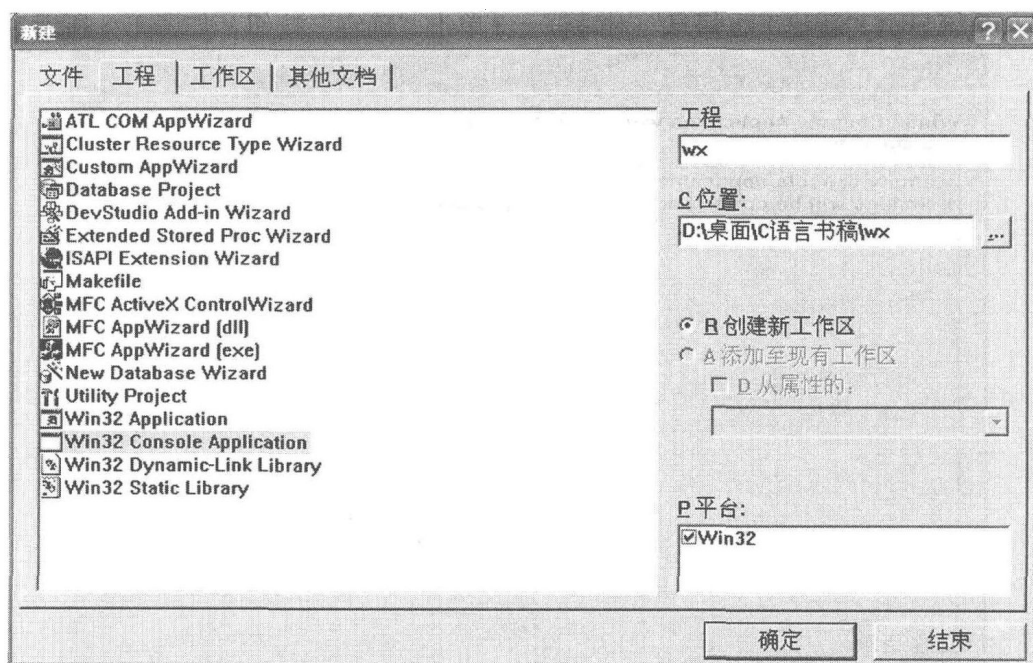


图 1.1

3) 弹出“Win32 Console Application Step 1 of 1”对话框→单击“An empty project”按钮和“完成”按钮，如图 1.2 所示。

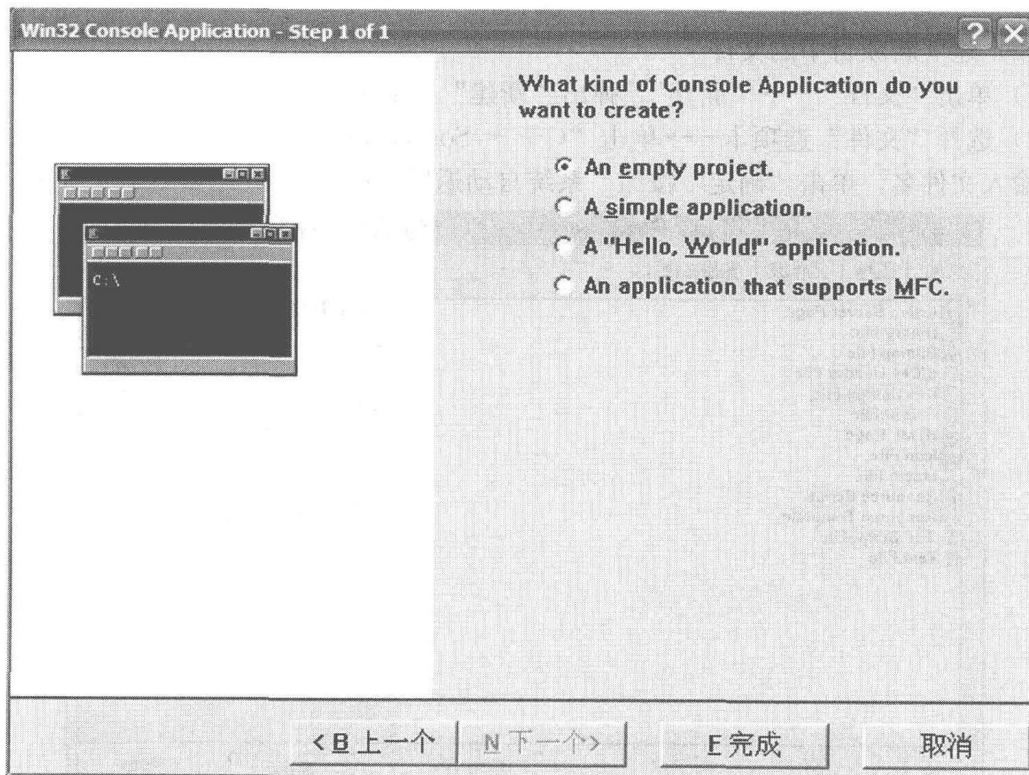


图 1.2