



普通高等教育“十二五”规划教材



21世纪高等学校计算机系列实用规划教材

C语言程序设计

王家海 邓长春 主编

本书配有
电子课件



东北大学出版社
Northeastern University Press



普通高等教育“十二五”规划教材



21世纪高等学校计算机系列实用规划教材

C 语言程序设计

王家海 邓长春 主编



东北大学出版社

• 沈阳 •

© 王家海 邓长春 2010

图书在版编目 (CIP) 数据

C 语言程序设计 / 王家海, 邓长春主编. —沈阳: 东北大学出版社, 2010.6
ISBN 978-7-81102-836-2 (2011.1 重印)

I. ①C… II. ①王… ②邓… III. ①C 语言—程序设计 IV. ①TP312

中国版本图书馆 CIP 数据核字 (2010) 第 132636 号

内 容 简 介

本教材根据《普通高等教育 C 语言程序设计课程教学大纲》的基本要求编写, 以“够用、实用”为原则, 以培养学生的应用创新能力为目标。所选的实例既有科学性和易学性, 又有趣味性和实用性; 例题由浅入深、循序渐进, 相关章节前后衔接、逐步扩展。旨在使学生了解 C 语言程序设计的基本知识, 掌握程序设计的思路和方法, 具备编程解决实际问题的基本能力, 为学习计算机后续课程打下坚实基础。

全书概念清晰, 讲解透彻, 重点突出, 示例典型, 实用性强; 可作为高职高专和应用型本科计算机相关专业的教材, 并可供软件开发人员参考。

出 版 者: 东北大学出版社

地址: 沈阳市和平区文化路 3 号巷 11 号

邮编: 110004

电话: 024—83687331 (市场部) 83680267 (社务室)

传真: 024—83680180 (市场部) 83680265 (社务室)

E-mail: neuph @ neupress.com

http: // www. neupress. com

印 刷 者: 抚顺光辉彩色广告印刷有限公司

发 行 者: 东北大学出版社

幅面尺寸: 184mm × 260mm

印 张: 18

字 数: 465 千字

出版时间: 2010 年 6 月第 1 版

印刷时间: 2011 年 1 月第 2 次印刷

责任编辑: 郭爱民 刘 莹

封面设计: 唐敏智

责任校对: 王艺霏

责任出版: 杨华宁

ISBN 978-7-81102-836-2

定 价: 32.00 元

《C 语言程序设计》编委会

主 编：王家海 邓长春

副主编：崔海源 闫英战

编 委（以姓氏笔画为序）：

包汉宗 李香林 陈 锦

余晓兰 杨俊成 张红军

胡伟平 陶 颖 葛颖增

前 言

C语言是一种计算机程序设计语言。它兼有高级语言和汇编语言的特点,既可以作为系统设计语言,编写系统应用程序,又可以作为应用程序设计语言,编写不依赖计算机硬件的应用程序,因此,应用范围极为广泛。

本书参照普通高等教育C语言程序设计课程教学大纲的基本要求编写,将培养应用创新能力的目标融会、贯穿于教材之中,以“够用、实用”为原则,精简传统C语言的教材内容知识,通过简明扼要、通俗易懂地介绍C语言的基本概念、基本语法和编程方法,使学生能在短时间内快速掌握程序设计的方法,选用的实例既有科学性和易学性的演示类例题,又有趣味性和实用性的设计类例题,例题由浅入深、循序渐进,相关章节保持连续性,前后衔接,逐步扩展。由于计算机语言是实践性非常强的课程,在各章中均配有实训内容,以加强上机实践这一必须重视的关键环节。本书旨在使学生了解程序设计语言的基本知识,掌握程序设计的基本方法与思路,进而较为深入地理解程序的内涵,并基本具备编程解决实际问题的能力,也为学习计算机后续课程打下扎实基础。

本书共分12章。第1章重点介绍C语言的特点、程序的结构特点,以及在Microsoft Visual C++ 6.0环境下运行C语言程序的操作方法。第2章介绍C语言编程必须掌握的一些基础知识,包括最基本的数据类型、对数进行运算的运算符和数据的输入输出。第3,4,5章通过实例讲述C语言程序的三种基本结构(顺序结构、选择结构、循环结构)的特点及程序的设计方法。第6,7,10章分别讲述C语言中的高级的数据类型(数组、指针、结构体等),培养读者对数据的组织能力,为进一步表示复杂的数据结构打下基础。第8,9章介绍的函数和编译预处理有助于培养读者模块化程序设计及解决大型应用问题的能力。第11章位运算的讲述有助于进一步加深对数据在计算机中存储方式的理解。第12章介绍了解决大量数据的输入输出问题的方法。

本书第1,2,11,12章由王家海、包汉宗、李香林、胡伟平编写;第3,4,5章由崔海源、余晓兰编写;第6,7章由闫英战、陶颖编写;第8,9,10章由邓长春、杨俊成、张红军、葛颖增编写;实验一至实验十三由陈锦编写。

全书概念清晰、结构合理、内容严谨、讲解透彻、重点突出、示例典型、

实用性强,既考虑到初学者的特点,又能满足软件设计人员的工作需要,不仅可以作为高等院校本、专科学生初学计算机高级语言程序设计的教材和计算机培训班学员 C 语言的考前培训教材,也适合广大软件开发人员和自学人员参考阅读。

由于作者水平有限,时间比较仓促,书中难免有错误和疏漏之处,恳请广大读者批评指正。

编者

2010年6月

目 录

第1章 C语言程序概述	1
1.1 C语言程序的基本结构	1
1.1.1 实 例	1
1.1.2 C语言程序的基本结构	3
1.2 C语言程序的运行	3
1.2.1 C语言程序运行过程	3
1.2.2 C语言开发环境 VC++6.0 简介	4
1.3 本章小结	6
1.4 实 训	6
实训 1.1	6
实训 1.2	7
实训 1.3	7
习 题	7
第2章 数据的类型与基本操作	8
2.1 一个C程序实例	8
2.2 基本数据类型	9
2.2.1 整数类型	9
2.2.2 实数类型	10
2.2.3 字符类型	10
2.3 常量和变量	10
2.3.1 常 量	10
2.3.2 变 量	12
2.4 运算符和表达式	14
2.4.1 算术运算符与算术表达式	14
2.4.2 赋值运算符与赋值表达式	15
2.4.3 数据类型转换	15
2.4.4 几个特殊的运算符	16
2.5 本章小结	18
2.6 实 训	18
实训 2.1	18

实训 2.2	19
实训 2.3	19
实训 2.4	19
习 题	20
第 3 章 顺序结构程序设计	21
3.1 一个顺序结构程序实例	21
3.2 数据的输入输出	22
3.2.1 printf 输出函数	22
3.2.2 scanf 输入函数	24
3.2.3 字符输入输出函数	26
3.3 顺序结构应用实例	27
3.4 本章小结	28
3.5 实 训	29
实训 3.1	29
实训 3.2	29
实训 3.3	29
习 题	30
第 4 章 选择结构程序	32
4.1 一个选择结构程序实例	32
4.2 关系运算与逻辑运算	33
4.2.1 关系运算符与关系表达式	33
4.2.2 逻辑运算符与逻辑表达式	34
4.3 由 if 语句构成的选择结构	35
4.3.1 if 语句	35
4.3.2 if~else 语句	36
4.3.3 if 的嵌套	37
4.3.4 条件表达式	39
4.4 switch 语句和 break 语句	40
4.5 选择结构应用实例	43
4.6 本章小结	45
4.7 实 训	45
实训 4.1	45
实训 4.2	46
实训 4.3	47
习 题	48
第 5 章 循环结构程序设计	51
5.1 一个循环结构程序实例	51

5.2	while 语句	52
5.3	do-while 语句	53
5.4	for 语句	54
5.5	多重循环	55
5.6	break 和 continue 语句	57
5.6.1	break 语句	57
5.6.2	continue 语句	57
5.7	应用举例	58
5.7.1	累加累乘	58
5.7.2	求极值	60
5.7.3	迭代	61
5.7.4	穷举	62
5.8	本章小结	64
5.9	实训	64
实训 5.1		64
实训 5.2		65
实训 5.3		65
实训 5.4		66
习 题		67
第 6 章	数 组	69
6.1	一个数组程序实例	69
6.2	一维数组	70
6.2.1	一维数组的定义和初始化	70
6.2.2	一维数组应用举例	72
6.3	二维数组	74
6.3.1	二维数组的定义和初始化	74
6.3.2	二维数组应用举例	76
6.4	字符串	80
6.4.1	字符数组的一般操作方法	80
6.4.2	字符串处理函数	83
6.5	本章小结	85
6.6	实训	86
实训 6.1		86
实训 6.2		86
实训 6.3		87
实训 6.4		87
实训 6.5		88
习 题		88

第7章 指 针	91
7.1 一个指针程序实例	91
7.2 变量的指针	92
7.3 数组指针	94
7.3.1 一维数组的指针	94
7.3.2 二维数组的指针	98
7.3.3 字符串的指针	101
7.3.4 动态存储分配	104
7.4 指针应用实例	105
7.5 本章小结	106
7.6 实 训	107
实训 7.1	107
实训 7.2	107
实训 7.3	107
实训 7.4	108
习 题	108
第8章 函 数	111
8.1 一个函数的例子	111
8.2 函数的定义和调用	112
8.2.1 函数的定义	112
8.2.2 函数的调用与函数的返回	113
8.3 参数传递	114
8.3.1 变量作为函数形参	114
8.3.2 指针变量作为函数形参	116
8.3.3 数组作为函数形参	117
8.4 函数的嵌套调用	120
8.5 变量的作用域	121
8.5.1 局部变量	121
8.5.2 全局变量	124
8.5.3 变量存储类型与模块化程序设计	126
8.6 本章小结	126
8.7 实 训	127
实训 8.1	127
实训 8.2	128
实训 8.3	129
实训 8.4	130
习 题	130

第 9 章 编译预处理	132
9.1 宏定义	132
9.1.1 符号串宏	132
9.1.2 带参数宏	133
9.1.3 函数与宏比较	134
9.2 文件包含	134
9.3 本章小结	135
9.4 实 训	135
习 题	136
第 10 章 结构体与共用体	137
10.1 一个结构体的例子	137
10.2 结构体的定义与引用	138
10.2.1 结构体类型和结构体变量的定义	138
10.2.2 结构体变量的使用	141
10.3 结构体数组与结构体指针	142
10.3.1 结构体数组	142
10.3.2 结构体指针	144
10.4 链 表	145
10.4.1 链表概述	145
10.4.2 链表的基本操作	148
10.4.3 链表操作完整程序	153
10.5 共用体	158
10.6 枚 举	160
10.7 用 typedef 定义类型符	161
10.8 本章小结	161
10.9 实 训	162
实训 10.1	162
实训 10.2	163
实训 10.3	163
实训 10.4	163
习 题	164
第 11 章 位运算	168
实 训	171
习 题	171

第 12 章 文 件	172
12.1 文件概述	172
12.1.1 文件概念和类型	172
12.1.2 文件指针	173
12.2 文件基本操作	173
12.2.1 文件的打开和关闭	173
12.2.2 文件的读写	175
12.3 应用实例	176
12.4 本章小结	182
12.5 实 训	183
实训 12.1	183
实训 12.2	183
习 题	184
实 验	185
实验一 C 程序的运行环境	185
实验二 数据类型、运算符和表达式	190
实验三 顺序结构和选择结构程序设计	194
实验四 循环结构程序设计	198
实验五 数组应用	201
实验六 指向变量的指针	204
实验七 指向数组的指针	206
实验八 函数应用	209
实验九 学生成绩管理系统 (用函数实现)	214
实验十 结构体、共用体和位运算	225
实验十一 学生成绩管理系统 (用结构体实现)	229
实验十二 文 件	240
实验十三 学生成绩管理系统 (用文件实现)	242
附 录	255
附录 A 《C 语言程序设计》实验教学大纲	255
附录 B 编译出错信息	258
附录 C ASCII 码对照表	268
附录 D 运算符和结合性	269
附录 E 库函数	270

第 1 章

C 语言程序概述

【学习目的】 掌握 C 语言程序的基本结构和上机操作过程,使读者对 C 语言程序有一个概貌的了解,为后面的学习打下基础。

1.1 C 语言程序的基本结构

1.1.1 实 例

下面介绍几个简单的程序例子,说明 C 语言的组成特点。

【例 1.1】 在屏幕上输出一串字符: This is a C program

【程序代码】

```
#include <stdio.h>

void main()
{
    printf("This is a C program");
}
```

【运行结果】

This is a C program

说明:

- ① main()。主函数。C 语言程序由函数构成,但有且只有一个主函数。
- ② { }。函数体界定符,花括号中的内容被称为函数体,每个函数的函数体必须用一对花括号括起来,花括号必须成对使用。
- ③ printf()。标准输出函数,将程序运行结果显示到输出设备(显示器)上。
- ④ #include <stdio.h>。预处理命令。当程序中要使用输入函数或输出函数时,应有此行。

【例 1.2】 已知 $a = 10$, $b = 30$, 求两数之和 sum。

【程序代码】

```
#include <stdio.h>
void main()
{
    int a, b, sum;
    a = 10;
    b = 30;
    sum = a + b;
    printf("sum = %d\n", sum);
}
/* 主函数, 程序从此开始运行 */
/* 函数体开始 */
/* 定义语句 */
/* 赋值语句 */
/* 赋值语句 */
/* 赋值语句 */
/* 输出语句 */
/* 函数体结束 */
```

【运行结果】

sum = 40

说明:

① int a, b, sum; 语句。说明 a, b, sum 为三个整型变量, 可以通过赋值操作改变变量的值。

② a = 10; 语句。将整数 10 送到 a 的存储单元中。sum = a + b; 语句: 首先取 a 和 b 两个存储单元中的数据在运算器中相加, 然后将结果保存在 sum 变量单元中。

③ printf("sum = %d\n", sum); 语句。按照 "sum = %d\n" 格式控制的要求输出 sum 的值。

④ "/* */"。注释, 目的是为了增加程序的可读性。注释可以插入到程序中任何位置, 对程序的执行没有任何影响, 编译时将被过滤掉。

【例 1.3】从键盘输入两个整数, 输出最大值。

【程序代码】

```
#include <stdio.h>
int max(int x, int y)
{
    int z;
    if(x > y)
        z = x;
    else
        z = y;
    return z;
}
/* 定义函数 */
/* 返回函数值 */

void main()
{
    int a, b, c;
    printf("输入两个整数 a b:");
    scanf("%d %d", &a, &b);
    c = max(a, b);
    printf("%d %d 中的最大值为: %d\n", a, b, c);
}
/* 标准输入函数 */
/* 调用 max() 函数 */
/* 输出结果 */
```

【运行结果】

输入两个整数 a b: 15 17<回车>

15 17 中的最大值为: 17

说明:

- ① 本程序由两个函数组成: 主函数 main() 和自定义函数 max()。
- ② 程序从 main() 函数开始执行; printf() 函数输出显示一个字符串, 具有提示信息的作用; scanf() 函数从键盘输入数据, 分别给变量 a 和变量 b 赋值, 使得变量 a 和 b 从键盘上获取值; 执行 c=max(a, b); 时, 程序转移到 max() 函数, 遇到 return 语句, 返回主函数, 继续执行。

1.1.2 C语言程序的基本结构

从以上实例可以看出 C 语言程序的基本结构。

① C 语言程序是由函数构成的, 函数是 C 程序的基本单位。

② 一个函数由两部分组成:

函数头: 即函数的第一行;

函数体: 即函数头下面, 用花括号 “{ }” 括起来的部分。

③ 函数体由语句构成, 语句以分号 “;” 结束。

④ 一个 C 程序可以由一个或多个函数组成, 但必须有一个且只能有一个 main() 函数, 即主函数。一个 C 程序总是从 main 函数开始执行, 不论 main 函数在整个程序中的位置。

⑤ 每行通常写一条语句。有些短语句也可以一行写多条, 长语句也可以一条写成多行。

⑥ 在程序中尽量使用注释信息, 增强程序的可读性。注释信息是用注释符标识的, 注释符开头用 “/*”, 结束用 “*/”, 其间的字符为注释信息。

1.2 C语言程序的运行

1.2.1 C语言程序运行过程

由高级语言编写的程序称为源程序, 计算机不能直接识别和执行, 必须由语言处理程序将其翻译成由 0 和 1 构成的二进制指令代码。按照 C 语言规则编写的程序, 要想得到最终结果, 应经过以下几个步骤。

(1) 编辑源程序

编辑是指使用文本编辑工具软件输入和修改 C 语言源程序, 最后以文本文件的形式存放在磁盘上, 文件名由用户自己选定, 扩展名一般为 “.c”。编辑器既可以是任何一种文本编辑软件, 比如 Turbo C 和 VC++ 专用编辑系统, 也可以是写字板、记事本等字处理软件。

(2) 编译源程序

编译是将 C 源程序翻译成二进制目标程序。编译是由编译程序来完成的, 编译程序对源程序自动进行句法和语法检查, 当发现错误时, 便将错误的类型和程序中出错的位置显示

出来,以帮助用户修改源程序。若未发现句法和语法错误,则自动地形成目标程序,其扩展名为“.obj”。

(3) 连接程序

编译后的目标文件尽管是二进制代码文件,但计算机还不能直接执行该程序,必须使用 C 语言处理系统提供的连接程序,生成扩展名为 .EXE 的可执行文件。程序中各函数间的调用结合是由连接程序完成的,系统提供的输出函数和用户定义的函数都要进行连接。如果连接过程中出现错误信息,也必须回到第一步重新修改源程序,并重新开始编译和连接,直到生成可执行文件。

(4) 运行程序

运行程序,并检查运行结果。若是算法错误,则只能回到第一步修改源程序,再重新编译连接和运行,直到得到正确的结果。

1.2.2 C 语言开发环境 VC++ 6.0 简介

(1) 启动 VC++

单击“开始”菜单,选择“程序”→“Microsoft Visual Stdio 6.0”→“Microsoft Visual C++ 6.0”。

(2) 创建 C++ 程序环境

① 在“文件 (Files)”菜单下选择“新建 (New)”命令,打开“新建”对话框,如图 1-1 所示。

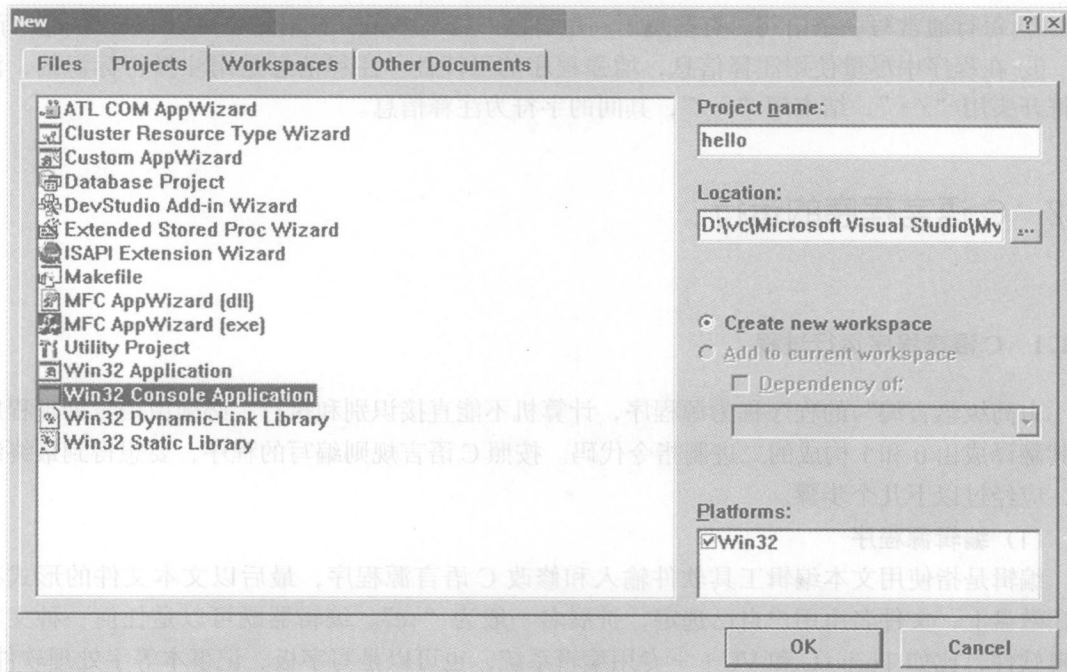


图 1-1 新建项目界面

② 在“工程 (Projects)”选项卡中指定下列选项:

- 选中 Win32 Console Application 工程类型;
- 在“工程名称 (Project name)”栏中输入工程名称, 如输入: hello;
- 单击“位置 (Location)”栏后的 , 可指定应用程序存放的位置 (或默认);
- 选中  **创建新工作区** (Create new workspace) (默认);
- 在 **平台** (Platforms) 中, 选中 Win32 (默认);
- 设置完成后, 单击“确定 (OK)”按钮, 打开 AppWizard 对话框。

③ 在应用向导 (AppWizard) 对话框中, 选中 An empty project。然后单击“完成 (Finish)”按钮, 显示“新建工程信息 (New project information)”对话框, 内容如下:

Empty console application.

No files will be created or add to the project.

单击“确定 (OK)”按钮, 则系统自动创建一个 hello 类。

④ 源程序编写。点击“文件 Files”菜单中“新建 New”命令, 则弹出如图 1-2 所示“新建”窗口。

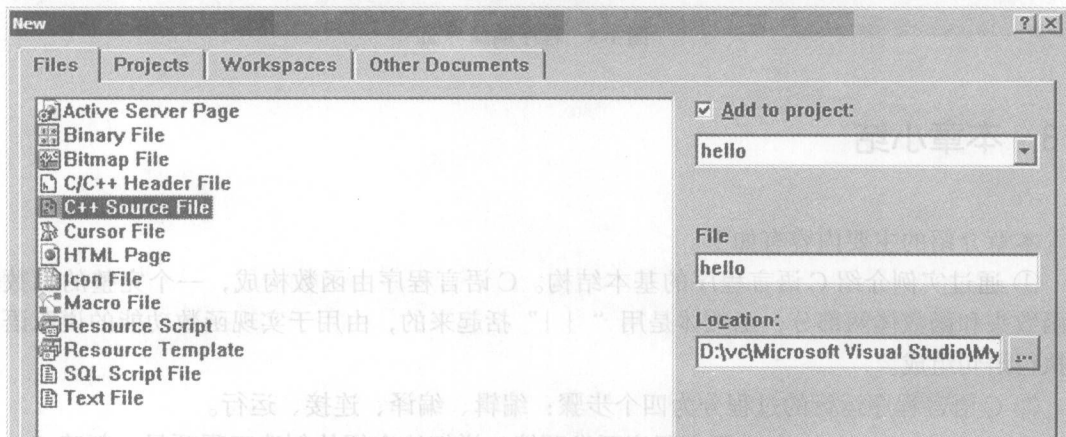


图 1-2 新建文件界面

在弹出的“新建 (New)”窗口的“文件 (Files)”对话框中, 选中“C++ Source File”, 在“文件 (File name)”栏中输入新建的 C++ 源文件名, 如 hello, 确认添加工程检查框被选中 (默认), 然后单击“确定 (Finish)”按钮, 建立一个空的 hello.cpp 文件。

(3) 编辑源程序

在图 1-3 所示的程序编辑窗口中输入程序, 编辑好的程序称为源程序, C++ 源程序的扩展名为 .cpp。

(4) 程序编译

选择编译工具栏中的“编译”按钮  进行编译, 编译信息显示在输出窗口中。如有错误, 则必须修改源程序再重新编译; 否则, 无法进行下一步。

(5) 连接程序

选择编译工具栏中的“构建”按钮  进行连接。如有错误, 则必须修改源程序再重新编译和连接。

(6) 运行程序

选择编译工具栏中的“执行”按钮 。