



普通高等教育“十一五”国家级规划教材 计算机系列教材

C/C++程序设计案例教程

仇芒仙 朱 蓉 主 编
魏远旺 李永刚 副主编

清华大学出版社





普通高等教育“十一五”国家级规划教材 计算机系列教材

仇芒仙 朱 蓉 主 编
魏远旺 李永刚 副主编

C/C++程序设计案例教程

清华大学出版社
北京

内 容 简 介

本书是普通高等教育“十一五”国家规划教材,是“C/C++ 语言程序设计”精品课程建设的配套教材。

全书内容共分两篇。第一篇是基础篇,介绍 C 语言程序设计的基本知识及其应用,主要内容包括数据类型、程序控制结构、数组、函数、结构体与共用体、文件等;第二篇是提高篇,介绍 C++ 程序设计方法。主要内容有面向对象程序设计、C++ 程序的基本知识、类与对象、继承与组合、指针与多态性、重载与模板等。

本书按照“精讲多练、重在实践”的原则,以“案例驱动知识”的写作方法,以 C 语言内容为入门,循序渐进,由浅入深地将面向过程程序设计和面向对象程序设计的知识有机地结合起来,体现了教材的应用性、创造性和先进性。

本书可作为高等学校程序设计课程的教材,也可以作为培训或自学教材。

本书封面贴有清华大学出版社防伪标签,无标签者不得销售。

版权所有,侵权必究。侵权举报电话:010-62782989 13701121933

图书在版编目(CIP)数据

C/C++ 程序设计案例教程/仇芒仙,朱蓉主编.--北京:清华大学出版社,2012.8

(计算机系列教材)

ISBN 978-7-302-28803-9

I. ①C… II. ①仇… ②朱… III. ①C 语言—程序设计—高等学校—教材 IV. ①TP312

中国版本图书馆 CIP 数据核字(2012)第 098704 号

责任编辑:魏江江 薛 阳

封面设计:常雪影

责任校对:白 蕾

责任印制:张雪娇

出版发行:清华大学出版社

网 址: <http://www.tup.com.cn>, <http://www.wqbook.com>

地 址:北京清华大学学研大厦 A 座 邮 编:100084

社 总 机:010-62770175 邮 购:010-62786544

投稿与读者服务:010-62776969, c-service@tup.tsinghua.edu.cn

质 量 反 馈:010-62772015, zhiliang@tup.tsinghua.edu.cn

课 件 下 载: <http://www.tup.com.cn>, 010-62795954

印 装 者:北京密云胶印厂

经 销:全国新华书店

开 本:185mm×260mm

印 张:22

字 数:531 千字

版 次:2012 年 8 月第 1 版

印 次:2012 年 8 月第 1 次印刷

印 数:1~3000

定 价:34.50 元

产品编号:041579-01

《C 语言程序设计案例教程》(含实验)自 2009 年第 1 版印刷以来,受到广大读者的欢迎,随着计算机技术应用的普及,信息时代对 IT 人员提出了更高的要求与挑战,不仅要掌握面向过程的 C 程序设计语言,还要掌握面向对象的程序设计语言,Visual C++ 成为继 C 语言之后的首选的面向对象的程序设计语言。为此,编者按照普通高等教育“十一五”国家规划教材建设的要求,结合新的教育教学的要求对第 1 版教材进行了必要的修订。

修订后的教材其教学目标与前版相同,重点在培养学生用计算机解决实际问题的能力,提高学生的编程和软件开发能力,为计算机科学与技术各相关专业学生后续课程的学习与实践打下良好的基础。同时,根据读者的反馈意见,将数组与函数的内容在顺序上进行了调整,增加了 C++ 程序设计的知识,总结教学中的经验,对部分内容进行了修改,教材中的实例都在 Visual C++6.0 环境下完成。本教材推荐使用时间为一学年,第 1 学期讲授 C 语言程序设计(64 学时或 80 学时,含实验 32 学时);第 2 学期讲授 C++ 程序设计(大概 64 学时,含实验 32 学时)。通过一学年的学习使读者能够循序渐进地掌握 C/C++ 程序设计方法与技巧,为今后的程序研发工作夯实基础。

《C/C++ 程序设计案例教程》全书共分两篇。

第一篇是基础篇,重点介绍 C 语言程序设计内容,共分 9 章:第 1 章和 2 章介绍 C 语言的基本知识;第 3 章是数据类型及应用,介绍 C 语言中所用到的各种基本数据类型;第 4 章介绍 C 语言的三种程序控制结构,即顺序结构、选择结构与循环程序结构及其应用;第 5 章是数组及应用。主要介绍一维数组、二维数组及字符数组的定义、初始化及引用。着重对指针访问数组进行了举例;第 6 章是函数及应用,主要介绍了函数的定义、函数的调用、全局变量与局部变量的应用,重点介绍了指针作为函数参数的引用;第 7 章是结构体与共用体的定义与引用,对单链表的建立、插入与删除进行了举例;第 8 章是文件的读写及应用;第 9 章是位运算操作。

第二篇是提高篇,重点介绍 C++ 程序设计方法,共分 7 章:第 10 章和第 11 章主要介绍面向对象程序设计与 C++ 程序的基本知识,重点介绍与 C 语言程序设计的异同内容;第 12 章是类与对象,介绍 C++ 中类的定义及其应用;第 13 章是继承与组合,介绍继承的定义与组合的应用;第 14 章是多态性,主要介绍多态性的作用以及多态性的实现;第 15 章是异常处理机制,介绍 C++ 的异常处理机制和自定义异常类,重点介绍如何通过 try-throw-catch 语句实现异常处理;第 16 章是综合应用,运用 C++ 的知识点开发一个简单的图形对象程序。

另外,为了更好地培养学生的编程能力,配合计算机科学与技术、信息管理专业课程建设的需要,在编写《C/C++程序设计案例教程》的过程中,写作组的老师们根据多年教学实践与经验,提出编写了《C/C++程序设计学习指南》的配套辅助教学教材,为课堂教学、实验指导、学生自学提供立体的教学资源。本书既适合高等院校的本科学生使用,也适合于热爱软件开发人员作为自学的参考资料。

本教材的作者都是高等院校的教师。本书的出版是“C/C++程序设计”精品课程建设中所催生的产物,在课程建设过程中,编写组成员对教材内容、教学方法进行了多次的研讨,最后确定了以“案例驱动知识”的写作方法,经过几年的使用,受到广大读者的好评。

本教材的第一篇由仇芒仙(第1~3章和第5章)、魏远旺(第4章)、张丽华(第6~8章)、李永刚(第9章)编写;第二篇由朱蓉(第10章)、许杰(第11章和第12章)、赵利平(第13和第15章)、叶利华(第14章和第16章)编写。全书由仇芒仙教授和朱蓉博士统稿。在本书再版之际,对修订过程中给予帮助并提出了建设性的意见的殷联甫、龚迅炜、殷海明等表示感谢。

对支持本书再版的清华大学出版社的领导及魏江江表示感谢!

感谢嘉兴学院计算机科学与技术学科的支持!

由于编者水平有限,书中疏漏之处在所难免,敬请各位读者及同仁不吝赐教,谢谢!

编 者

2012年4月

第一篇 基础篇

第 1 章 C 程序概述	3
1.1 C 程序的结构	3
1.1.1 C 程序实例	3
1.1.2 C 程序构成	4
1.1.3 书写程序时应遵循的原则	6
1.2 C 语言的发展	6
1.2.1 C 语言的发展	7
1.2.2 C 语言的特点	8
1.3 数据在计算机中的存储形式与表示方法	9
1.3.1 内存的组织形式	9
1.3.2 原码、反码和补码	10
习题 1	10
第 2 章 程序与算法	12
2.1 利用计算机解决问题的基本过程	12
2.2 程序的算法	13
2.2.1 算法的概念	14
2.2.2 算法的描述	14
2.3 结构化程序设计	17
2.3.1 结构化程序的基本结构	17
2.3.2 结构化程序设计的特点与方法	18
2.3.3 结构化程序设计的过程	18
2.3.4 程序设计的原则	19
2.4 C 程序的运行步骤	21
习题 2	22

第 3 章 数据类型及其应用	23
3.1 C 语言的数据类型	23
3.2 常量	24
3.3 变量	26
3.3.1 变量的命名	26
3.3.2 变量的分类与说明	27
3.3.3 变量的地址——指针	29
3.4 运算符和表达式	32
3.4.1 常见的运算符	32
3.4.2 表达式	33
3.4.3 数据类型转换	40
3.5 数据的输出	42
3.5.1 printf() 函数	42
3.5.2 字符输出函数 putchar()	44
3.6 数据的输入	45
3.6.1 scanf() 函数	45
3.6.2 字符输入函数 getchar()	47
习题 3	48
第 4 章 控制结构	51
4.1 顺序结构	51
4.2 选择结构	52
4.2.1 if 语句	52
4.2.2 switch 语句	56
4.3 循环结构	59
4.3.1 用 while 语句实现循环	59
4.3.2 用 do-while 语句实现循环	61
4.3.3 用 for 语句实现循环	63
4.3.4 循环嵌套	66
4.3.5 break 语句和 continue 语句	68
4.3.6 goto 语句	70
4.4 综合实例	71
习题 4	72
第 5 章 数组	76
5.1 一维数组	76
5.1.1 一维数组的定义和引用	78
5.1.2 一维数组的初始化	78

5.1.3	一维数组的应用	80
5.2	字符串与字符数组	81
5.2.1	字符数组的定义和引用	82
5.2.2	字符串处理函数	83
5.2.3	字符数组综合举例	88
5.3	二维数组	89
5.3.1	二维数组的定义和引用	90
5.3.2	二维数组的初始化	90
5.3.3	二维数组的应用	91
5.3.4	多维数组	93
5.4	数组与指针	94
5.4.1	用指针访问数组元素	94
5.4.2	指针数组	99
5.4.3	指向数组的指针	99
5.4.4	多级指针	101
5.4.5	与指针有关的一些问题	102
习题 5		103
第 6 章	函数	106
6.1	概述	106
6.2	函数的定义	108
6.2.1	有返回值函数的定义	109
6.2.2	无返回值函数的定义	110
6.3	函数的调用	110
6.3.1	函数调用的形式	110
6.3.2	函数的参数	111
6.3.3	被调函数的原型声明	113
6.4	函数的嵌套调用和递归调用	114
6.4.1	函数的嵌套调用	114
6.4.2	函数的递归调用	115
6.5	变量的作用域与生存期	117
6.5.1	局部变量和全局变量	118
6.5.2	变量的生存期	120
6.5.3	存储类别小结	125
6.6	指针与函数	127
6.6.1	指针变量作为函数参数	127
6.6.2	数组作为函数参数	129
6.7	返回指针的函数和函数指针	132
6.7.1	返回指针的函数	132

6.7.2 函数指针	133
6.8 main 函数的参数	135
6.9 编译预处理	137
6.9.1 宏定义	138
6.9.2 文件包含	139
6.9.3 条件编译	140
习题 6	141
第 7 章 结构体和共用体	146
7.1 概述	146
7.1.1 引例	147
7.1.2 结构体类型说明	149
7.2 结构体变量	150
7.2.1 结构体变量的定义	150
7.2.2 结构体变量的初始化	152
7.2.3 结构体变量的引用	153
7.3 typedef 的使用	156
7.4 结构体数组	157
7.4.1 结构体数组的定义和初始化	157
7.4.2 结构体数组的引用	158
7.5 结构体指针	160
7.5.1 指向结构体变量的指针	160
7.5.2 指向结构体数组的指针	163
7.6 链表	165
7.6.1 链表概述	166
7.6.2 静态链表	166
7.6.3 动态内存管理函数	168
7.6.4 动态链表的基本操作	169
7.6.5 单链表综合应用实例	173
7.7 共用体	174
7.8 枚举类型	178
习题 7	180
第 8 章 文件	182
8.1 文件概述	182
8.1.1 文件分类	182
8.1.2 文件指针	183
8.2 文件的打开与关闭	183
8.2.1 文件的打开	184

8.2.2 文件的关闭	185
8.3 文件的读写	186
8.3.1 格式化读写函数	186
8.3.2 字符读写函数	188
8.3.3 字符串读写函数	190
8.3.4 数据块读写函数	191
8.4 文件的其他操作	193
8.4.1 文件的定位	193
8.4.2 文件的检测	196
习题 8	197
第 9 章 位操作	198
9.1 位运算符	198
9.1.1 按位与运算	199
9.1.2 按位或运算	199
9.1.3 按位异或运算	199
9.1.4 求反运算	200
9.1.5 左移运算	200
9.1.6 右移运算	201
9.2 位段	201
习题 9	204

第二篇 提 高 篇

第 10 章 C++ 程序设计基础	207
10.1 面向对象程序设计概述	207
10.2 编程方法	209
10.2.1 结构化编程	210
10.2.2 面向对象编程	210
习题 10	210
第 11 章 C++ 的基本语法	212
11.1 一个简单的 C++ 程序	212
11.2 C++ 与 C 在基本语法上的区别	213
11.2.1 构词	213
11.2.2 构造类型	214
11.2.3 构句	218
11.3 C 为何会演变为 C++	220
习题 11	221

第 12 章 类与对象	222
12.1 类与对象	222
12.2 类的定义	223
12.2.1 定义类的内涵	223
12.2.2 结构与类	224
12.2.3 访问控制	225
12.2.4 完整的类定义	227
12.2.5 类的成员	229
12.2.6 类的友元	234
12.2.7 类的大小	234
12.3 对象应用	236
12.3.1 对象的生存周期	236
12.3.2 对象的拷贝	237
12.3.3 对象作为类的成员	238
12.3.4 对象的 this 指针	239
12.3.5 对象之间的消息	241
12.4 C++为何还不是纯粹的 OO?	241
12.4.1 什么是 OO?	241
12.4.2 C++五大经典定律	242
12.4.3 为何不是纯 OO?	242
习题 12	243
第 13 章 继承与组合	245
13.1 继承	245
13.1.1 派生类定义	246
13.1.2 继承方式	248
13.1.3 重写基类函数	253
13.1.4 派生类和基类的构造函数与析构函数	254
13.1.5 多继承	257
13.1.6 类型兼容规则	263
13.2 组合	264
习题 13	265
第 14 章 多态性	269
14.1 基本概念	269
14.2 函数重载	269
14.3 运算符重载	271
14.3.1 运算符重载的限制	274

14.3.2	运算符重载的成员和非成员	275
14.3.3	运算符重载分类	275
14.4	虚函数	283
14.4.1	虚函数基础	283
14.4.2	虚析构造函数	287
14.5	抽象类	289
14.5.1	纯虚函数	289
14.5.2	抽象类	290
14.5.3	重载、隐藏和覆盖	293
14.6	模板	295
14.6.1	函数模板	295
14.6.2	类模板	297
14.6.3	STL 简单应用	299
习题 14		303
第 15 章	异常处理机制	309
15.1	异常处理的使用	309
15.1.1	异常处理机制	311
15.1.2	抛出异常	311
15.1.3	声明异常	314
15.1.4	异常传播	314
15.1.5	使用 C++ 异常类	314
15.2	自定义异常类	316
习题 15		318
第 16 章	综合应用	320
案例简述		320

第一篇 基础篇

C 语言程序设计的基本知识及其应用

第 1 章 C 程序概述

本章主要知识点：

- ◆ C 程序的结构；
- ◆ C 语言的发展；
- ◆ 数据在计算机中的存储形式与表示方法。

从 1946 年世界上第一台电子计算机诞生到现在,计算机的发展经历了 4 个阶段,迄今为止,仍然采用计算机之父“冯·诺依曼”的思想——存储程序的原理,即计算机要想做任何事情,都要按照一定的“程序”去做,也就是我们先编写好程序,然后输入到计算机中,计算机才能按照“程序”的要求去完成。因此你想做一个能够驾驭计算机的高手,那么就一定要掌握用计算机语言编程的本领,而 C 语言是目前国际上广泛流行的计算机语言,它也是步入程序设计殿堂的敲门砖,也是学好计算机相关专业后续课程的专业基础课,目前很多游戏软件、杀毒软件、工具软件、控制软件等都是用它编写的。

那么,C 语言能干什么,用 C 语言又如何编写程序呢?下面通过实例来了解。

1.1 C 程序的结构

在了解 C 语言程序之前,我们先通过几个实例对 C 程序结构有一个初步认识。本节主要介绍 C 语言程序的基本结构。

1.1.1 C 程序实例

“程序”是计算机完成确定任务的具体步骤。

【实例 1-1】 在屏幕上输出“I am student !”。

```
/* 实例 1-1 */  
#include <stdio.h>  
void main()  
{  
    printf("I am student !");  
}
```

```
/* 注释语句 */  
/* 预处理命令 */  
/* 主函数 */  
/* 输出字符串 "I am student !" */
```

分析说明：以上程序只用到输出库函数 printf,因此在程序开头一定要有 #include <stdio.h> 预处理命令,此命令的功能是将“stdio.h”包含在当前 C 源程序中,因为库函数 printf 属于“stdio.h”文件的一部分,换句话说,就是建立了头文件“stdio.h”与当前 C 程序的链接。

运行结果如图 1-1 所示。

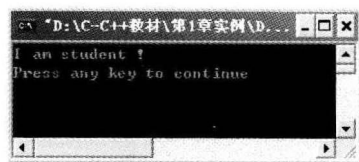


图 1-1 屏幕输出示例运行结果

【实例 1-2】 求两个数中的最大数。

```

/* 实例 1-2 */
#include <stdio.h>
void main()                                /* 主函数 */
{
    int x,y,t;                             /* 说明部分,定义整型变量 x,y,t */
    int max(int,int);                      /* 函数声明 */
    printf(" please input x,y: ");        /* 在屏幕上显示字符串 "please input x,y: " */
    scanf ("%d,%d",&x,&y);                /* 输入 x,y 的值 */
    t= max(x,y);                           /* 赋值语句 */
    printf("x=%d,y=%d,max=%d",x,y,t);    /* 输出 x,y,t 的值 */
}
int max(int a,int b)                       /* 函数 max */
{
    if(a>b)                                /* 条件语句 */
        return a;                         /* 条件成立,返回值 a */
    else
        return b;                         /* 条件不成立,返回值 b */
}

```

运行结果如图 1-2 所示。

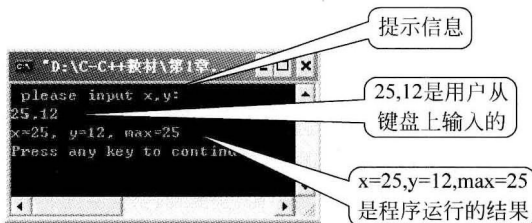


图 1-2 求最大数程序的运行结果

分析说明：实例 1-2 是由两个函数组成，一个是主函数 `void main()`，另一个是 `int max(int a,int b)` 函数。`void` 是函数的类型，表示此函数执行后不会返回函数值，即“空函数”。`int max` 函数是求 `a`、`b` 两个数中最大者，函数返回值是整型的。

1.1.2 C 程序构成

以上两个程序都是 C 语言的“源程序”。不难看出 C 语言程序的构成：

(1) C 程序由函数组成。一个 C 程序至少包含且仅包含一个主函数 `main()`，也可以是包含一个主函数 `main()` 和若干个其他函数。C 程序的运行是从 `main` 函数开始，结束在 `main` 函数。`main` 函数后面的 `()` 是表示函数的属性，是函数的标志。

函数由两部分组成：函数首部和函数体。实例 1-2 中的 `void main()`、`int max(int a,int b)` 都是函数首部，它表示函数的类型、函数名和函数参数及参数的类型。

例如 `max()` 的首部为：

int	max	(	int	a ,	int	b)
↑	↑		↑	↑	↑	↑
函数的类型	函数名		参数的类型	函数参数	参数的类型	函数参数

max()的首部表示函数 max 返回值是整型的 int ,并且有两个整型参数 a 和 b。

花括号{}中是函数的内容,也称函数体。函数体由一系列语句组成。

(2) C 程序的语句,每个语句由分号结束。C 程序的语句大概分为:注释语句、说明语句、执行语句和空语句。

注释语句:由“/ * ”与“ * /”括起来的一串字符。字符串可以是中文字符也可能是英文字符,注释语句是非执行语句,其作用是方便别人阅读程序所加的说明。它可以出现在程序的任何地方。

说明语句:说明语句一般放在程序体的前面,比如“int x,y,t;”和“int max(int,int);”语句,前者是说明 x,y,t 是“int”类型,后者是说明 max 函数是“int”类型并带有两个“int”的参数,也可以写成“int max(int a ,int b);”。

点拨 “int max(int,int);”语句也称函数声明语句。因 max 是被调函数,即它可以被主函数 main 或其他函数来调用,在实例 1-2 中,主函数 main 在前,被调函数的定义在后,所以在程序主函数 main 中要先对 max 进行说明。

执行语句:表示能执行特定的操作。

C 语言的执行语句有表达式语句、函数调用语句、控制语句、复合语句、赋值语句和空语句。

① 表达式语句

表达式语句由表达式加上分号“;”组成。

其一般形式为:表达式;

执行表达式语句就是计算表达式的值。例如:a=3 为一个赋值表达式,若在后面加上一个分号则构成赋值语句 a=3; 即一个表达式的最后加一个分号就成为一个语句。

② 函数调用语句

由函数名、实际参数加上分号“;”组成。

其一般形式为:函数名(实际参数表);

执行函数语句就是调用函数体并把实际参数赋予函数定义中的形式参数,然后执行被调函数体中的语句,计算函数值并返回。

例如:printf("C Program");

该语句功能是调用库函数 printf(),输出字符串“C Program”。

有关函数定义与调用的详细内容见后续章节。

③ 控制语句

控制语句用于控制程序的流程,以实现程序的各种结构方式。它们由特定的语句定义符组成。C 语言共有 9 种控制语句,可分成以下 3 类。

a. 条件判断语句:if 语句,switch 语句。

b. 循环执行语句:do while 语句,while 语句,for 语句。

c. 转向语句:break 语句,goto 语句,continue 语句,return 语句。

④ 复合语句

把多个语句用{}括起来组成的一个语句称复合语句。在程序中应把复合语句看成是单