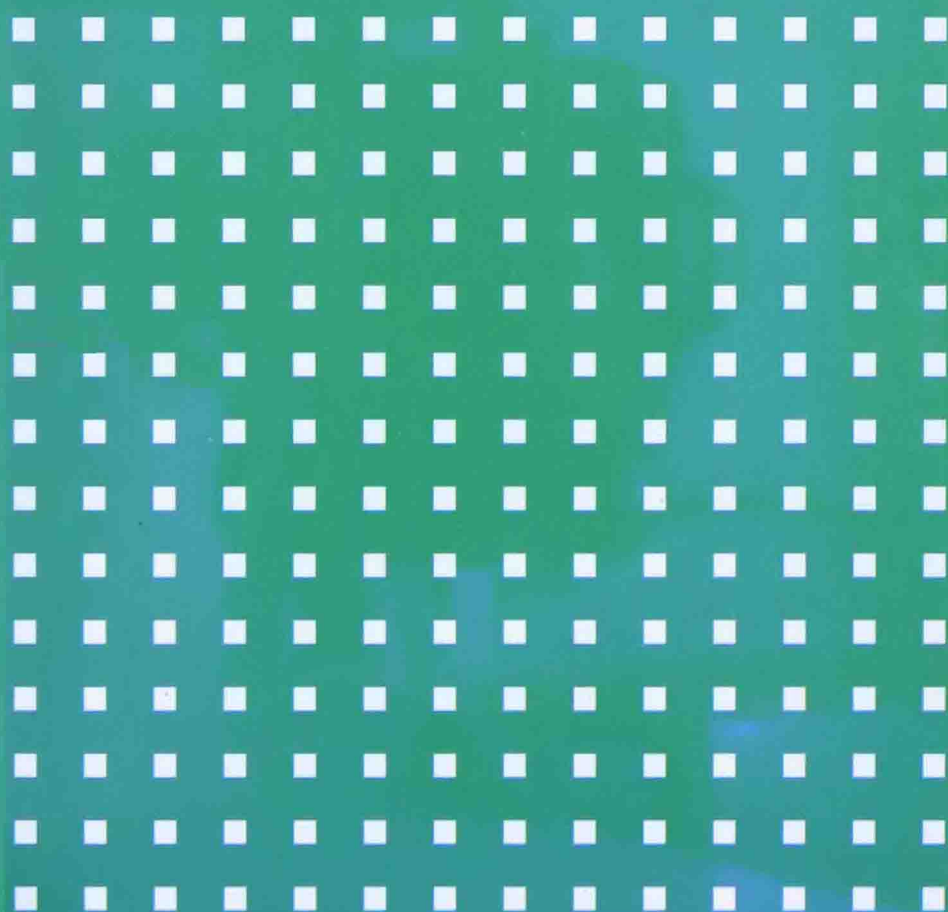


C程序设计案例教程 习题解答与上机指导

林小茶 编著



高等学校计算机专业教材精选·算法与程序设计

C程序设计案例教程 习题解答与上机指导

林小茶 编著

清华大学出版社
北 京

内 容 简 介

本书是与清华大学出版社出版的教材《C 程序设计案例教程》(林小茶编著)配套的习题解答与上机指导,同时也适合单独作为学习 C 程序设计的辅导教材。本书给出了主教材部分习题的参考答案,上机指导部分依据教学进度设计了 8 个实验,除了实验目的和要求外,还有对实验步骤的详尽描述,同时对使用 Visual C++ 6.0 调试程序的方法也做了详细介绍,希望为学习者带来方便。

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 程序设计案例教程习题解答与上机指导/林小茶编著. —北京:清华大学出版社,2015

高等学校计算机专业教材精选·算法与程序设计

ISBN 978-7-302-38325-3

I. ①C… II. ①林… III. ①C 语言—程序设计—高等学校—教学参考资料 IV. ①TP312

中国版本图书馆 CIP 数据核字(2014)第 241290 号

责任编辑:张 民 薛 阳

封面设计:傅瑞学

责任校对:白 蕾

责任印制:刘海龙

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:三河市少明印务有限公司

经 销:全国新华书店

开 本:185mm×260mm 印 张:13.5

字 数:326 千字

版 次:2015 年 1 月第 1 版

印 次:2015 年 1 月第 1 次印刷

印 数:1~2000

定 价:26.00 元

产品编号:061531-01

前 言

本书是与清华大学出版社出版的教材《C 程序设计案例教程》(林小茶编著)配套的习题解答与上机指导。

学过程序设计的人都有体会,看别人编写的程序好像挺明白的,而一旦要自己编写一个程序会感觉无从下手,这是因为程序设计是一门对实践环节要求很高的课程。初学者要想真正学会 C 程序设计,最重要的是抓住两个关键环节:一个是多做程序设计的习题、多编程;另一个就是多上机,写在纸上的程序是否正确,最好的办法就是上机验证一下。

习题解答并没有给出全部习题的参考答案,只给出了大部分程序设计题目的答案,原因在于教学过程中发现很多同学在原封不动地抄答案。其实,仔细阅读和学习主教材,答案都是可以找到的,在找寻答案的过程中,还可以学习教材的内容。

考虑到初学者对调试工具的掌握有一定的难度,因此本书实验指导部分写得非常详细,对每个实验按步骤进行描述,由难而易,由浅入深,并尽量使用截图来说明。学生只要按照书中描述的步骤去做,一定可以逐步掌握调试程序的方法,也给指导上机的老师带来极大的方便。

提供习题解答的主要目的是给学习者一个参考和借鉴,作者在这里要特别强调一点,程序设计是一个创作的过程,就像写作文似的,针对一个命题,作文可以有很多种写法,因为不同的人思考问题的方法不同。同样,解决一个实际问题的 C 程序肯定不是唯一的,因此,在阅读本书的参考答案之前,希望读者已经独立思考过主教材中的习题,不要急于看答案,这样才有助于程序设计水平的提高,并且不要把本书的参考源程序作为唯一的答案,如果能写出更好的程序,那是作者所乐于看到的。作者对本书的一些习题也使用了不同的方法和思路,希望能对读者有更多的帮助。

殷切希望读者对本书的内容和编写方法提出宝贵的意见和建议,特此感谢。

作者联系方式: xiaocl@bistu.edu.cn

编 者
2014 年 12 月

目 录

第一部分 《C 程序设计案例教程》部分习题解答	1
第 1 章 认识 C 语言	1
第 2 章 顺序结构程序设计	3
第 3 章 选择结构程序设计	11
第 4 章 基础知识深化	21
第 5 章 循环结构程序设计	27
第 6 章 函数	44
第 7 章 数组	66
第 8 章 指针	89
第 9 章 结构体、联合体与枚举	111
第 10 章 文件	136
第二部分 C 程序设计与上机指导	152
第 11 章 上机指导	152
11.1 实验 1 C 基础练习	152
11.1.1 实验目的和要求	152
11.1.2 实验指导	152
11.2 实验 2 顺序与选择结构	156
11.2.1 实验目的和要求	156
11.2.2 实验指导	156
11.3 实验 3 循环结构	164
11.3.1 实验目的和要求	164
11.3.2 实验指导	164
11.4 实验 4 函数	172
11.4.1 实验目的和要求	172
11.4.2 实验指导	172
11.5 实验 5 数组	180
11.5.1 实验目的和要求	180
11.5.2 实验指导	180

11.6	实验 6 指针	188
11.6.1	实验目的和要求	188
11.6.2	实验指导	188
11.7	实验 7 结构体、联合体和枚举	195
11.7.1	实验目的和要求	195
11.7.2	实验指导	195
11.8	实验 8 文件	199
11.8.1	实验目的和要求	199
11.8.2	实验指导	199
附录	Visual C++ 6.0 的动态调试命令	206

第一部分 《C 程序设计案例教程》

部分习题解答

第 1 章 认识 C 语言

【1-1】 编写程序显示下列图案,并上机调试该程序。

```
* * * * *
 * * * * *
  * * * * *
   * * * *
    * * *
```

解:

```
/* ----- */
/* 显示三角形----- */
#include "stdio.h"
int main()
{
    printf(" * * * * * \n");
    printf("  * * * * * \n");
    printf("   * * * * * \n");
    printf("    * * * * \n");
    printf("     * * \n");
    return 0;
}
```

【1-2】 编写程序显示下列界面,并上机调试该程序。

```
-----
请输入你的姓名:
-----
请输入你的学号:
-----
```

解:

```
/* ----- */
```

```
/* 登录界面----- */
#include "stdio.h"
int main()
{
    printf("-----\n");
    printf("请输入你的姓名:      \n");
    printf("-----\n");
    printf("请输入你的学号:      \n");
    printf("-----\n");
    return 0;
}
```

第2章 顺序结构程序设计

【2-6】 多项选择题(下列各题中每道题有 A、B、C、D 4 个选项,正确答案超过一个,请选择正确答案)。

(1) 可以作为 C 语言的用户定义字的是()。

- A. 4a B. value_ C. _m66 D. ok?

解: 正确答案是 B 和 C。

分析: 4a 的第一个字符是 4, 不符合 C 语言规定标识符的第一个字符不能是数字的规定; 而 ok? 中使用了符号?, 这是 C 语言的标识符规则所不允许的。

(2) 不能作为 C 语言的用户定义字的是()。

- A. age B. int C. a+c D. did

解: 正确答案是 B 和 C。

分析: int 是 C 语言的关键字, 不能用关键字作为用户定义字; a+c 中使用了符号+, C 语言的标识符规则不允许使用该符号。

(3) 不能作为 C 语言的用户定义字的是()。

- A. a * b B. mail C. do D. _x

解: 正确答案是 A 和 C。

分析: a * b 中使用了符号*, C 语言的标识符规则不允许使用该符号; do 是 C 语言的关键字, 不能用关键字作为用户定义字。

(4) 下列常量中, 在 C 语言中合法的整数是()。

- A. 4567 B. 0877 C. 09 D. -10

解: 正确答案是 A、D。

分析: 按整数的书写规则, 0877 和 09 的第 1 位是数字 0, 表示该数是八进制数, 表示八进制的数字是从 0 到 7, 不能有 9 或 8。

(5) 下列常量中, 在 C 语言中不合法的整数是()。

- A. 2f B. 0x400l C. 08 D. 0x12ed

解: 正确答案是 A、C。

分析: 按整数的书写规则, 2f 表示的是普通的十进制数, 表示十进制的数字是从 0 到 9, 不能有 f; 08 的第 1 位是数字 0, 表示该数是八进制数, 表示八进制的数字是从 0 到 7, 不能有 8。

(6) 下列常量中, 十进制值为 23 的是()。

- A. 027 B. 0x17 C. '\007' D. '\027'

解: 正确答案是 A、B、D。

分析: 027 是八进制数, 其十进制值为 $2 \times 8 + 7 = 23$; 0x17 是十六进制数, 其十进制值为 $1 \times 16 + 7 = 23$; '\007' 是一个字符, 十进制值为 7; '\027' 是一个字符, 该字符的 ASCII 值是八进制数 027, 那么该字符在内存存储时是十进制值 23。

【2-7】 请修改下列程序,使其能够通过编译,并正确运行。

(1)

```
#include {stdio.h}
Int Main
{
    int a=10;b=20;c;
    c=a * b;
    printf(' %d',c);
    return 0;
}
```

解: 正确程序是:

```
#include "stdio.h"                /* 修改 */
int main()                        /* 修改 */
{
    int a=10,b=20,c;              /* 修改 */
    c=a * b;
    printf(" %d",c);              /* 修改 */
    return 0;
}
```

分析: 原题中的程序错误有下面几处。

① #include 后面的文件名应该用双引号括起,也可以用符号<和符号>将文件名括起。

② “Int Main”一行中大写 I 和 M 都是错误的,C 语言的关键字应该是小写。另外,main 后面缺少了函数符号()。

③ “int a=10;b=20;c;”一句中的 int 只定义了 a,就结束了定义语句,“b=20;”中的 b 就没有定义了。将分号改为逗号就行了。当然,修改成“int a=10;int b=20;int c;”也是正确的。

(2)

```
#include "stdio.h"
int main();
{
    char c
    getchar(c);
    printf("%s",c);
    return 0;
}
```

解: 正确程序是:

```
#include "stdio.h"
int main()                        /* 修改 */
{
    char c;                       /* 修改 */
    c=getchar();                  /* 修改 */
    printf("%c",c);               /* 修改 */
}
```

```
return 0;
}
```

分析：原题中的程序错误有下面几处。

- ① “main()”后面不能有分号。
- ② 语句“char c”后面应该有分号,这是一个说明语句。
- ③ getchar 函数的调用方法不是 getchar(c),而应该是 c=getchar()。
- ④ %s 是控制输出字符串的,而%c 才是控制输出一个字符的。所以用%s 控制输出字符变量 c 是错误的。

(3)

```
#include "stdio.h" int main()
{   double f=7.12;
    char c="c";
    printf("%d\n",int(f%3));
    printf("%c",c);
    return 0;
}
```

解：正确程序是：

```
#include "stdio.h"
int main()                                /* 修改 */
{   float f=7.12;
    char c='c';                          /* 修改 */
    printf("%d\n", (int)f%3);             /* 修改 */
    printf("%c",c);
    return 0;
}
```

分析：原题中的程序错误有下面几处。

- ① #include "stdio.h" 与 int main()放在同一行,是错误的,应该分两行书写。
- ② “char c="c";”一句中 c 是一个字符变量,不能将字符串常量值赋给它。
- ③ f%3 是错误的,运算符%的操作对象必须是整型数。

(4)

```
#include "stdio.h"
int main()
{
    integer I;
    long j;
    printf("Enter an integer:");
    scanf("%d",&i);
    j=I*I;
    printf("I=%d  j=%d\n,j);
```

```

    return 0;
}

```

解：正确程序是：

```

#include "stdio.h"
int main()
{
    int I;                /* 修改 */
    long j;
    printf("Enter an integer:");    /* 修改 */
    scanf("%d",&I);        /* 修改 */
    j=I*I;
    printf("I=%d  j=%ld\n",I,j);    /* 修改 */
    return 0;
}

```

分析：原题中的程序错误有下面几处。

- ① integer 不是 C 语言的关键字。
- ② “printf(“Enter an integer:);”函数调用中右边的双引号没有写。
- ③ “scanf(“%d”,&i);”中的 i 没有经过定义,经过定义的是 I,而不是 i。C 语言严格区分大小写。
- ④ “printf(“I= %d j= %d\n,j);”一句不但少了右边的双引号,显示的数据个数也与控制字符串不符合,并且 j 是长整型数,应该用 %ld 控制输出。

【2-8】 判断下列程序的运行结果。

(1)

```

#include "stdio.h"
int main()
{
    int i=0x12,j=012;
    printf("\ni=0%o,j=0%o",i,j);
    return 0;
}

```

解：

程序的运行结果是：

i= 022,j= 012

分析：“int i=0x12,j=012;”一句中赋值给变量 i 的整数常量 0x12 是十六进制数,相当于十进制数 18,八进制数 22。因此按八进制数输出变量 i 就是 22,%o 规定输出格式是按八进制数输出数据,%o 之前的数字 0 是程序员为了使显示结果与十进制有所区别,特意加的一个直接显示的字符。

“int i=0x12,j=012;”一句中赋值给变量 j 的整数常量 010 是八进制数。因此按八进制数输出变量 j 还是 12,同样,%o 之前的数字 0 也是特意加的一个直接显示的

数 22 的八进制和

33 转换成二进制

位形成十六进制的

数 160 表示的字符

是 p, 十六进制数 6e 表示的字符是 n。

【2-9】 按照下列要求编写程序。

(1) 编写程序用 %c 输出字符变量的方法输出下面的图案。

```
aaaaaaa
 bbbbbb
  ccccc
   ddddd
```

解:

```
#include "stdio.h"
int main()
{
    char a='a',s=' ',n='\n';
    printf("%c%c%c%c%c%c%c%c",a,a,a,a,a,a,a,n);
    printf("%c%c%c%c%c%c%c%c",s,a+1,a+1,a+1,a+1,a+1,a+1,n);
    printf("%c%c%c%c%c%c%c%c",s,s,a+2,a+2,a+2,a+2,a+2,a+2,n);
    printf("%c%c%c%c%c%c%c%c",s,s,s,a+3,a+3,a+3,a+3,a+3,n);
    return 0;
}
```

分析: 本题要求用 printf 函数输出字符常量。因此不能将要输出的内容直接放在控制字符串中。因而定义了三个变量, 分别存储星号、空格和回车符, 然后用 %c 控制显示。

(2) 编写程序输入三个数, 求它们的平均值并输出, 用浮点数据处理。

解:

方法一

```
#include "stdio.h"
int main()
{
    double a,b,c,aver;
    printf("请输入三个浮点数:");
    scanf("%lf%lf%lf",&a,&b,&c);
    aver=(a+b+c)/3;
    printf("平均值=%lf\n",aver);
    return 0;
}
```

方法二

```
#include "stdio.h"
int main()
{
    double a,sum=0;
    printf("请输入三个浮点数:");
    scanf("%lf",&a);
```

```

sum=sum+a;
scanf("%lf",&a);
sum=sum+a;
scanf("%lf",&a);
sum=sum+a;
printf("平均值=%lf\n",sum/3);
return 0;
}

```

运行情况:

请输入三个浮点数:1 2 3 ✓
平均值=2.000000

分析: 方法一的编程思路是一次读入三个浮点数,用一个公式求它们的平均值。求平均值时,不能将公式写成“ $\text{sum} = 1/3 * (a + b + c)$ ”,因为 $1/3$ 是 0 了! 初学者要特别注意!

方法二的编程思路是一次读入一个浮点数,读三次,每读一个用公式“ $\text{sum} = \text{sum} + a$;”将读入的数累加到存储单元 sum 中。最后直接用 $\text{sum}/3$ 计算平均值。充分理解本方法将有助于学习循环语句。因为语句“ $\text{scanf}("%lf",&a);$ ”和“ $\text{sum} = \text{sum} + a$;”一共重复使用了三次。

注意: sum 的初值要设置为 0。

(3) 编写程序将输入的英里转换到千米。每英里等于 5280 英尺,每英尺等于 12 英寸,每英寸等于 2.54cm,每千米等于 100000cm。

解:

```

#include "stdio.h"
int main()
{
    double mile,k;
    printf("请输入英里:");
    scanf("%lf",&mile);
    k=mile*5280*12*2.54/100000;
    printf(" %lf 英里= %lf 千米\n",mile,k);
    return 0;
}

```

运行情况:

请输入英里:8 ✓
8.000000 英里=12.874752 千米

分析: 尽管本题并不难,提醒读者在编写程序时注意公式“ $k = \text{mile} * 5280 * 12 * 2.54 / 100000$;”的正确写法。正确写法并不唯一,但是公式“ $k = 5280 / 100000 * 12 * 2.54 * \text{mile}$;”就肯定会出错。

(4) 假设港币与人民币的汇率是 1 港币兑换 1.0607 元人民币,编写程序输入人民币的钱数,输出能兑换的港币金额。

解:

```
#include "stdio.h"
int main()
{
    double rmb,hkdollar;
    printf("请输入人民币:");
    scanf("%lf",&rmb);
    hkdollar=rmb/1.0607;
    printf("人民币 %.2lf 可以换 %.2lf  港元\n",rmb,hkdollar);
    return 0;
}
```

运行情况:

请输入人民币: 4000 ✓
人民币 4000 可以换 3771.09 港元

分析: 题目比较简单,只要公式写对就行了。注意使用浮点数运算。

(5) 编写程序输入年利率 I (例如 2.5%),存款总数 S (例如 30 000 元),计算一年后的本息合计并输出。

解:

```
#include "stdio.h"
int main()
{
    double i,s;
    printf("请输入利率和本金:");
    scanf("%lf%lf",&i,&s);
    s=s*(1+i);
    printf("一年后的本息合计=%.2lf\n",s);
    return 0;
}
```

运行情况:

请输入利率和本金:0.025 30000 ✓
一年后的本息合计=30750.00

分析: 题目比较简单,只要公式写对就行了。注意使用浮点数运算。

第 3 章 选择结构程序设计

【3-1】 语句替换练习,替换后要保证程序段功能相同。

(1) 用 if else if 形式代替下面程序段中的 switch 语句。

```
switch (k)
{
    case 2:
        case 9: printf("Error!\n");break;
        case 3:
        case 10: printf("Right!\n");break;
        default: printf("Try again!\n");
}
```

解:

```
if (k==2||K==9)
    printf("Error!\n");
else if (k==3||K==10)
    printf("Right!\n");
else
    printf("Try again!\n");
```

(2) 用 switch 语句代替下面程序段中的 if else if 语句(k 为整数)。

```
if (k==0)
    printf("Error!\n");
else if (k>=1&&k<=3)
    printf("Right!\n");
else if (k==4||k==5)
    printf("Just OK!\n");
else
    printf("Try again!\n");
```

解:

```
switch (k)
{
    case 0: printf("Error!\n");break;
    case 1: case 2: case 3: printf("Right!\n");break;
    case 4: case 6: printf("Just OK!\n");break;
    default: printf("Try again!\n");
}
```

(3) 用条件运算符代替下面程序段中的 if else 语句。