



21世纪全国高等院校计算机教育“十一五”规划教材
丛书主编 全国高等学校计算机教育研究会课程与教材建设委员会主任 李大友

C语言程序设计

上机指导及习题解答

主 编 张志宏 李春霆
副主编 李晓东 肖向东
黄文生 田文生



21世纪全国高等院校计算机教育“十一五”规划教材

C语言程序设计上机指导 及习题解答

本书编委会 编著

中国计划出版社

图书在版编目(CIP)数据

C语言程序设计上机指导及习题解答 / 《C语言程序设计上机指导及习题解答》编委会编著. —北京: 中国计划出版社, 2008. 1

21世纪全国高等院校计算机教育“十一五”规划教材
ISBN 978-7-80242-063-2

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

中国版本图书馆CIP数据核字(2008)第005261号

内 容 简 介

本书是《C语言程序设计》一书的配套用书。书中采用与《C语言程序设计》一书相同的章节顺序, 每章由上机指导和习题解答两个部分组成。“上机指导”部分根据知识点的难易安排了有针对性的基础实验和拓展实验, 有效地指导学习者理论联系实际, 培养动手编程能力; “习题解答”部分对于主教材中难度较大的题目给出了详细解题思路, 有效地引导学习者培养正确的解题思路。附录部分给出了主流的程序调试工具的使用说明, 为立志从事编程工作的学习者打下基础。

本书既适合作为高等院校教学用书, 也适合计算机培训班上机操作使用。

21世纪全国高等院校计算机教育“十一五”规划教材 C语言程序设计上机指导及习题解答

本书编委会 编著

☆

中国计划出版社出版

(地址: 北京市西城区木樨地北里甲11号国宏大厦C座4层)

(邮政编码: 100038 电话: 63906433 63906381)

新华书店北京发行所发行

北京市艺辉印刷有限公司印刷

787×1092毫米 1/16 13.25印张 322千字

2008年1月第一版 2008年1月第一次印刷

印数1—5000册

☆

ISBN 978-7-80242-063-2

定价: 20.00元

丛书编委会

主任：李大友

副主任：王行言 郑 莉

委员：（按音序排列）

樊金生 冯春辉 高延武 韩金仓 刘凤田

刘 云 刘建臣 刘三满 罗德茹 彭宣戈

齐玉斌 孙晨霞 王潜平 王书海 姚 华

杨晓斌 张广斌 赵建明 赵连胜 邹修明

本书编委会

主 编：张志宏 李春霆

副主编：李晓东 肖向东 黄文生 田文生

参 编：高 飞 姜 河 王 颖 张 兵 张春娣

张向阳 高海忠 周雯琦

丛 书 序

随着我国高等教育发展与改革的逐步深化，越来越多的高等院校将其自身定位于工程型或应用型，立足于培养能够满足各行各业需求的，素质高、能力强的应用型专业人才。与此同时，由于信息化是当今社会与经济必然趋势，因而应用信息技术的能力亦将会成为衡量人才水平的重要标尺。由此可见，培养既具有专业知识，又拥有良好信息技术应用能力的人才，是现今高等教育发展与改革的目标之一。

目前，教育部有关计算机教育的教学指导委员会、全国高等学校计算机教育研究会等学术团体、各高等院校的专家学者已经在计算机教学与教材改革方面做了大量的工作，许多一线教师已经在计算机教学和科研方面积累了许多宝贵经验。在这些条件下，通过将其教研成果汇总并转化为教材的形式向全国各高等院校推广，对于促进高等院校计算机教育的发展与改革，培养应用型专业人才，是一件十分有意义的事情。

鉴于以上情况，中国计划出版社与全国高等学校计算机教育研究会决定联合策划组织、编写出版了本套“21世纪全国普通高等院校计算机教育‘十一五’规划教材”。为实施精品战略，出版社与全国高等学校计算机教育研究会在全国范围内进行了系统、详细的调查，对各层各类教学指导性文件进行了认真、深入的研究，对国内外已出版同类教材进行了客观、理性的分析，组织专家学者、一线教师及企业人员展开研讨，以期打造切实符合实际教学需求的精品教材。

为配合各学校的精品课程建设工程，本套教材以国家级精品课程指标为指引方向，借鉴其他兄弟出版社的先进经验和成功案例，提出了建设“立体化教学资源平台”的概念，其内容包括教材、教学辅导资料、教学资源包、网络平台等内容，并将在后续培训、论文发表等多方面满足教师与精品课程建设的需求。

本套教材具有以下特点。

1. 定位明确，应用为本

本套教材定位于高等院校学生计算机应用能力的培养，不仅要使学生理解计算机相关的基本理论与基本知识，还要使学生掌握利用计算机解决实际问题的能力。要使学生在面对一个实际问题时，不仅要知其然，还要知其所以然，更要会其如何然，最终，要具备实际操作应用能力。

2. 案例驱动，能力培养

本套教材通过从实际应用中精炼出的案例来辅助知识的讲授与能力的培养，在案例设计时从其科学性、实用性及开放性出发，尽量营造贴近实际应用的环境，激发学生的学习兴趣，从而提高教学效率，提高学生的实际应用能力。

3. 资源丰富，便于教学

我社免费为选用本套教材中图书的教师提供如下资源服务：

- 多媒体电子课件（PowerPoint 格式）
- 所有案例的相关素材（图片、声音与源程序等）与最终结果

- 所有习题的素材与答案
- 两套模拟测试题及答案
- 不定期组织教师培训

高等院校计算机教育的发展与改革不会停止，各院校的实际情况又有所不同，我们恳请各位老师在使用过程中提出批评与建议，以便及时改进教材欠妥与不足之处，使本套教材日趋完善。

我们相信在各位专家学者与一线教师的支持与帮助下，本套教材一定能成为特点鲜明、质量上乘的精品教材，同时，我们也希望通过本套教材的出版为高等院校计算机教育的发展与改革做出自己的一份贡献。

丛书编委会

前 言

本书是《C 语言程序设计》的配套书，同时也可以单独作为学习 C 语言的上机实验指导书。

本书采用与《C 语言程序设计》一书相同的章节顺序，每章由上机指导和习题解答两部分组成。第一部分“上机指导”包括基础实验和拓展实验，“基础实验”中给出上机实验的目的及要求及完成实验的步骤，所安排的实验与原书内容紧密相关，便于进行实验教学；“拓展实验”选取了较为典型的程序，将知识深化拓展，以提高读者的编程能力，同时引导读者深入思考，激发编程兴趣。第二部分“习题解答”是《C 语言程序设计》一书相应章节思考与练习题的参考答案，并对部分难度较大的选择题进行了解析，上机操作题的程序设计给出了参考程序与程序分析，以便于读者对相关知识点的深入理解。

在附录部分重点介绍了目前比较常用的 Turbo C++ 3.0 与 Visual C++ 6.0 集成环境下的 C 语言上机操作方法，使读者能较快地掌握在这两个集成环境下 C 程序的开发过程，相信对读者的上机练习会有较大的帮助。

本书由张志宏、李春霆主编，李晓东、肖向东、黄文生、田文生担任副主编，其中第 1、2 章由姜河、黄文生编写，第 3、4 章由王颖、李晓东编写，第 5、6 章由张兵、田文生编写，第 7、8 章由张春娣、肖向东编写，第 9、10 章由张向阳、高飞编写，第 11、12 章由高海忠、周雯琦编写。附录由李春霆编写。本书由李春霆统稿，高海忠参与了本书所涉及全部代码的调试，全书最后由张志宏负责审稿。

本书既适合作为学校教学用书，也适合计算机培训班使用，还可供报考计算机等级考试者和其他自学者参考。

由于时间仓促与编者水平有限，不足与欠妥之处在所难免，恳请广大读者不吝指正。

编者

2007年11月

目 录

第1章 绪论	1
1.1 上机指导	1
1.1.1 基础实验	1
1.1.2 拓展实验	3
1.2 习题解答	3
第2章 数据类型及其运算	6
2.1 上机指导	6
2.1.1 基础实验	6
2.1.2 拓展实验	8
2.2 习题解答	9
第3章 结构化程序设计	16
3.1 上机指导	16
3.1.1 基础实验	16
3.1.2 拓展实验	17
3.2 习题解答	17
第4章 选择结构程序设计	23
4.1 上机指导	23
4.1.1 基础实验	23
4.1.2 拓展实验	25
4.2 习题解答	25
第5章 循环结构程序设计	36
5.1 上机指导	36
5.1.1 基础实验	36
5.1.2 拓展实验	38
5.2 习题解答	39
第6章 数组	56
6.1 上机指导	56
6.1.1 基础实验	56
6.1.2 拓展实验	58
6.2 习题解答	60

第7章 函数	79
7.1 上机指导	79
7.1.1 基础实验	79
7.1.2 拓展实验	83
7.2 习题解答	84
第8章 编译预处理	100
8.1 上机指导	100
8.1.1 基础实验	100
8.1.2 拓展实验	101
8.2 习题解答	102
第9章 结构体与共用体	110
9.1 上机指导	110
9.1.1 基础实验	110
9.1.2 拓展实验	114
9.2 习题解答	115
第10章 指针	130
10.1 上机指导	130
10.1.1 基础实验	130
10.1.2 拓展实验	134
10.2 习题解答	137
第11章 位运算	151
11.1 上机指导	151
11.1.1 基础实验	151
11.1.2 拓展实验	152
11.2 习题解答	152
第12章 文件	158
12.1 上机指导	158
12.1.1 基础实验	158
12.1.2 拓展实验	159
12.2 习题解答	161
附录 C语言开发环境介绍	175
主要参考文献	201

第 1 章

绪论

1.1 上机指导

1.1.1 基础实验

1. 实验目的

- (1) 了解和使用 Turbo C++ 3.0 或 Visual C++ 6.0 集成开发环境。
- (2) 熟悉所用集成开发环境的基本编辑命令、常用功能菜单及快捷键，学会常规窗口的操作。
- (3) 通过运行简单的 C 程序，初步了解 C 源程序的特点。
- (4) 学习完整的 C 程序开发过程（编辑、编译、连接、调试、运行及查看结果）。

2. 实验内容

- (1) 检查所用计算机系统安装的 C 编译系统，并确定它所在的子目录。
- (2) 启动所用的 C 编译集成环境。
- (3) 熟悉 C 编译集成环境的界面和有关菜单的使用方法。
- (4) 输入并运行一个简单的、正确的程序。

输入下面的程序：

```
#include <stdio.h>
void main()
{   printf("*****");
    printf("\nThis is my first program\n");
    printf("*****");
}
```

根据本书附录部分所介绍的方法对源程序进行编译，观察屏幕上显示的编译信息。如果显示出错信息，则应找出原因并将其改正，再进行编译，如果正确，则进行连接。如果编译连接正确，运行程序，观察并分析运行结果。

(5) 输入并编辑一个有错误的 C 程序。

```
#include <stdio.h>
void main()
{   int a=2,b
    b=2*a+5;
    print("b is : %d\n,b);
}
```

进行编译, 仔细分析编译信息窗口, 可能显示有多个错误, 对于每个错误分析其产生的原因, 逐个修改直到正确。最后请与教材上的程序进行对照。

运行程序, 分析运行结果。

(6) 输入并运行一个需要在运行时输入数据的程序。

输入下面的程序:

```
#include <stdio.h>
void main()
{   int a,b;
    printf("Please input a: ");
    scanf("%d",&a);
    b=2*a+5;
    printf("b is : %d",b);
}
```

编译并运行, 观察运行结果。

【运行结果】

```
Please input a: 5
b is : 15
```

注意: 5表示在运行时从键盘输入5, 然后按回车键。用户输入的部分用下划线表示, 表示按回车键。下同。

(7) 输入并运行一个有函数调用的程序。

输入下面的程序:

```
#include <stdio.h>
void max(int a,int b)
{   int c;
    if(a>b)
        c=a;
    else
        c=b;
    printf("Max is :%d",c);
}
void main()
```

```

{   int a,b;
    scanf("%d,%d",&a,&b);
    max(a,b);
}

```

编译并运行，在运行时输入 7, 12，观察并分析运行结果。

【运行结果】

```

7,12✓
Max is :12

```

1.1.2 拓展实验

编写一个 C 程序，输入 a、b 和 c 三个值，输出其中的最大者。

【提示】 参照基础实验（7）示例编写程序，并按如下步骤调试程序。

- (1) 启动所用的 C 编译集成环境，输入编写的源程序。
- (2) 检查程序有无错误（包括语法错误和逻辑错误），有则改之。
- (3) 编译和连接，仔细分析编译信息，如有错误则应找出原因并将其改正。
- (4) 运行程序，分析结果。
- (5) 修改程序（例如故意改成错的）：分析其编译和运行情况。
- (6) 将调试好的程序保存在自己的用户目录中，文件名自定。
- (7) 将编辑窗口清空，再将该文件读入，检查窗口中的内容是否为刚才存盘的程序。
- (8) 关闭所用的 C 编译集成环境，找到刚才使用的用户子目录，浏览其中的文件，确认是否存在刚才保存的后缀为.c 的 C 源程序文件和编译连接后生成的后缀为.exe 的可执行文件。

1.2 习题解答

1. 选择题

(1) C 语言程序的基本结构单位是_____。

- A. 函数 B. 语句 C. 字符 D. 程序行

【答案】 A

(2) 一个 C 程序的执行是_____。

- A. 从本程序的 main 函数开始，到本程序的 main 函数结束
 B. 从本程序的 main 函数开始，到本程序的最后一个函数结束
 C. 从本程序的第一个函数开始，到本程序的最后一个函数结束
 D. 从本程序的第一个函数开始，到本程序的 main 函数结束

【答案】 A

(3) C 语言规定在一个 C 程序中，main 函数的位置_____。

- A. 必须在开始 B. 必须在最后
 C. 可以任意 D. 必须在系统调用的库函数之后

【答案】 C

(4) 能直接让计算机识别的语言是_____。

- A. 高级语言
- B. 机器语言
- C. 汇编语言
- D. C语言

【答案】 B

(5) 编辑程序的功能是_____。

- A. 调试程序
- B. 建立并修改文件
- C. 将C源程序编译成目标程序
- D. 命令计算机执行指定的操作

【答案】 B

2. 问答题

(1) 汇编语言与高级语言有何区别?

【答案】 计算机语言分为低级语言和高级语言两大类。汇编语言属于低级语言，是面向机器的语言，用这种语言编写的程序只适用于某种特定类型的计算机；而高级语言脱离机器指令用人们更加容易理解的方式来编写程序，可适用于多种类型的计算机，即高级语言比汇编语言的可移植性好。

(2) C程序中为什么要加注释语句?

【答案】 C程序中增加注释语句可以提高程序的可读性，从而使程序更加容易理解。

(3) 根据例题写出一个C语言程序包括哪些部分。

【答案】 一个C语言程序至少包含一个main函数，也可以包含一个main函数和若干个其他函数。即C程序是由函数构成的，函数是C程序的基本单位。

3. 上机操作题

(1) 根据例 1-1 思考，在第 3 行之前再多输出一段文字： I will study hard.

【参考程序】

```
#include <stdio.h>
void main()
{ printf("*****");
  printf("\nThis is my first program\n");
  printf("I will study hard\n");
  printf("*****");
}
```

【运行结果】

```
*****
This is my first program
I will study hard
*****
```

(2) 根据例 1-2 思考，输出 b 的值是 4*a，其中 a 的值为 4，运行结果为 16。

【参考程序】

```
#include <stdio.h>
void main()
{   int a=4,b;
    b=4*a;
    printf("b is : %d\n",b);
}
```

【运行结果】

b is : 16

第 2 章

数据类型及其运算

2.1 上机指导

2.1.1 基础实验

1. 实验目的

- (1) 掌握 C 语言的数据类型、常量、变量、运算符和表达式等概念。
- (2) 掌握变量的定义方法。
- (3) 掌握变量的赋值方法。
- (4) 掌握表达式的运算顺序。
- (5) 掌握自增、自减运算符的使用规律。
- (6) 掌握 C 语言程序的编辑、运行和调试方法。

2. 实验内容

- (1) 输入并运行下列程序。

```
#include <stdio.h>
void main()
{
    char c1,c2;
    c1='a';
    c2='b';
    printf("%c%c\n",c1,c2);
}
```

- ① 运行此程序，观察运行结果。

【运行结果】

ab

【程序分析】 本程序定义了两个字符型变量 c1 和 c2，变量的值是字符的 ASCII 码值。

在输出语句的格式串中，要求以字符格式输出，因此输出了 ab。

② 在上面的 printf 语句的下面再增加一个 printf 语句：printf("%d%d\n",c1,c2); 再运行程序，并观察结果。

【运行结果】

```
ab
9798
```

【程序分析】 第一行的 ab 是第一个输出语句的输出结果，第二行中的 9798，分别是字符 a 和字符 b 的 ASCII 码值，因为第二个输出语句的格式串中要求变量以十进制数格式输出。

③ 将 char c1,c2;行改为：int c1,c2;，运行并观察结果。

【运行结果】

```
ab
9798
```

【程序分析】 c1、c2 变量无论是定义成 int 型还是 char 型，为其赋值为字符型常量时，都是存储字符的 ASCII 码值。

④ 再将 c1='a';c2='b';两行改成：c1=a;c2=b;，注意不要加单撇号，观察运行结果。

【运行结果】 程序不能通过编译。

【程序分析】 这时无论把 c1、c2 定义成 char 型还是 int 型，程序都不能通过。若定义为 char 型，此时赋值语句中的 a 和 b 都不是 char 型常量，因此不能赋值；若定义为 int 型，此时赋值语句中的 a 和 b 都不是 int 型常量，因此也不能赋值，这就是 C 语言的数据类型及变量定义的内涵。

⑤ 再将 c1='a';c2='b';两行改成：c1="a";c2="b";，运行并观察结果。

【运行结果】 程序不能通过编译。

【程序分析】 形如"a"或"b"的数据属于字符串常量，C 语言不提供字符串变量，因此不能这样赋值。

(2) 输入并运行下列程序。

```
#include <stdio.h>
void main()
{
    int i,j,m,n;
    i=8;
    j=10;
    m=++i;
    n=j++;
    printf("%d,%d,%d,%d\n",i,j,m,n);
}
```

① 运行并观察结果。

【运行结果】

9,11,9,10

【程序分析】 程序定义了4个整型变量，i、j、m和n，i、j分别赋值8和10，i变量自增后（8+1=9）赋给变量m，所以i、m变量输出时的值为9；j变量（10）先赋给n后再自增，所以n变量的值为10，j变量输出时的值为11。

② 将程序进行如下修改，运行并观察结果。

```
#include <stdio.h>
void main()
{
    int i,j;
    i=8;
    j=10;
    printf("%d,%d,%d,%d\n",i++,j++,i,j);
}
```

【运行结果】

8,10,8,10

【程序分析】 输出语句的计算顺序是从右到左，这时i=8，j=10，输出i++、j++表达式的值也是8和10，然后再自增；输出时，还是从左到右输出。若把i、j和i++、j++的位置互换，请再次运行，此时输出的结果变为：9,11,8,10。

③ 将程序进行如下修改，运行并观察结果。

```
#include <stdio.h>
void main()
{
    int i,j,m=0,n=0;
    i=8;
    j=10;
    m+=i++;n--=-j;
    printf("i=%d,j=%d,m=%d,n=%d\n",i,j,m,n);
}
```

【运行结果】

i=9,j=9,m=8,n=-9

【程序分析】 这是一个含有自增自减的复合赋值运算，执行m+=i++时，m和i相加（0+8=8）后再赋给m，m=8，然后i自增i=9；执行n--=-j时，j先自减j=9，然后与n相减（0-9=-9）后再赋给n，n=-9。

2.1.2 拓展实验

(1) 输入一个256以内的整数，输出其对应的ASCII码字符（注意有些字符为不可显