

21世纪高等学校计算机规划教材

21st Century University Planned Textbooks of Computer Science

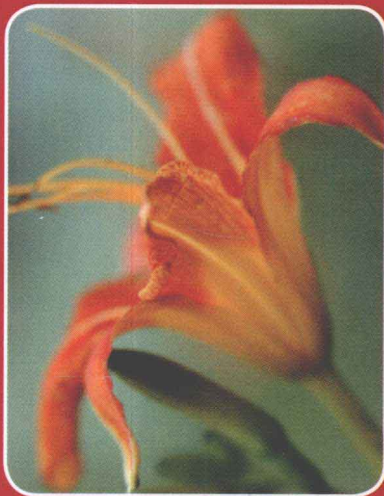
C/C++语言程序设计教程 实验指导与习题解答 (第3版)

The Practice and Answer of The C/C++
Programming Language (3rd Edition)

李丽娟 主编

吴蓉晖 洪跃山 李根强 杜四春 银红霞 副主编

- 核心知识点提示
- 案例引导实践练习
- 习题参考解答



精品系列



人民邮电出版社
POSTS & TELECOM PRESS

21世纪

21st Century

计算机规划教材

Books of Computer Science

C/C++语言程序设计教程 实验指导与习题解答 (第3版)

The Practice and Answer of The C/C++
Programming Language (3rd Edition)

李丽娟 主编

吴蓉晖 洪跃山 李根强 杜四春 银红霞 副主编



精品系列

人民邮电出版社

北京

图书在版编目 (CIP) 数据

C/C++语言程序设计教程实验指导与习题解答 / 李丽娟主编. — 3版. — 北京: 人民邮电出版社, 2012.2
21世纪高等学校计算机规划教材
ISBN 978-7-115-27324-6

I. ①C… II. ①李… 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(2012)第002845号

内 容 提 要

本书是和教材《C/C++语言程序设计教程——从模块化到面向对象(第3版)》配套的实验指导与习题解答。

全书共3部分。第1部分为实验指导,针对教材中每一章的知识点,安排不同的上机实验内容,并给出所有上机实验的参考解答。第2部分为习题解答,包括教材中各章的习题和参考解答,并对部分概念性较强的习题给出注解。第3部分为附录,简要介绍了 Visual C++ 6.0、Turbo C、Borland C++ 5.0 集成环境下程序的编辑、编译和运行的方法。

全书内容紧扣 C 语言程序设计的相关知识点和 C++ 的入门知识,案例丰富,实用性强,可作为学习 C 语言程序设计和 C++ 入门的配套教材。

21 世纪高等学校计算机规划教材

C/C++语言程序设计教程实验指导与习题解答 (第3版)

- ◆ 主 编 李丽娟
- 副 主 编 吴蓉晖 洪跃山 李根强 杜四春 银红霞
- 责任编辑 邹文波
- ◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街 14 号
邮编 100061 电子邮件 315@ptpress.com.cn
网址: <http://www.ptpress.com.cn>,
三河市海波印务有限公司印刷
- ◆ 开本: 787×1092 1/16
印张: 15 75 2012 年 2 月第 3 版
字数: 415 千字 2012 年 2 月河北第 1 次印刷

ISBN 978-7-115-27324-6

定价: 32.00 元

读者服务热线: (010)67170985 印装质量热线: (010)67129223

反盗版热线: (010)67171154

广告经营许可证: 京崇工商广字第 0021 号

第 3 版前言

在学习程序设计语言的过程中, 很多人都有过这样的想法: 概念易懂, 程序难写。其实, 出现这种情况的主要原因是缺少足够的实践与练习。为了帮助读者学好 C 语言, 掌握程序设计的基本方法, 我们结合多年的教学经验, 先后编写了《C 语言程序设计教程》及其配套实践教材、《C 语言程序设计教程(第 2 版)》及其配套实践教材。本书第 2 版出版以来, 受到了广大读者的欢迎, 被全国五十多所各类高等学校选为教材, 在两年多时间里, 先后 10 多次重印, 得到了很多高校的教师和学生的的好评。

本书在继承前两个版次特色的基础上, 结合近几年的教学实践以及广大读者和教师提出的意见和建议, 并特别根据近几年教学改革的实践以及对人才培养的高标准要求, 对其内容做了进一步的优化、补充和完善, 适当增加了 C++ 的基本知识。

本书共分为 3 部分。第 1 部分为实验, 共有 9 章, 每一章都简要地提出了学习目的和要求, 列出了相关知识和实验范例, 并给出了相关的实验练习。我们将为教师提供所有实验练习的程序参考答案。

第 2 部分为习题解答, 是对应教材中所有习题的解答, 并对部分疑难问题给出一些解答提示。读者可以根据参考答案, 进一步掌握 C 语言的基本概念和编程方法, 并初步掌握 C++ 程序设计方法。

第 3 部分为附录, 简单介绍了 Visual C++ 6.0、Turbo C 和 Borland C++ 5.0 集成开发环境的使用, 给出了这些环境下常见的编译错误信息, 帮助读者快速掌握 C/C++ 语言程序的调试。

所有程序的解答有可能不是唯一的, 读者可以进一步思考其他的解题方法, 以拓宽自己的思路。书中所有程序均在 Visual C++ 6.0 环境上调试通过。

本书由李丽娟任主编, 吴蓉晖、洪跃山、李根强、杜四春、银红霞任副主编。

第 1 部分的第 1 章和第 2 部分的第 1 章由吴蓉晖编写; 第 1 部分的第 3 章和第 2 部分的第 3 章由洪跃山编写; 第 1 部分的第 2 章、第 4 章和第 1 部分的第 2 章、第 4 章由李根强编写; 第 1 部分的第 5 章、第 6 章和第 2 部分的第 5 章、第 6 章由杜四春编写; 第 1 部分的第 7 章、第 8 章、第 9 章由银红霞编写, 第 2 部分的第 7 章、第 8 章、第 9 章及第 3 部分的附录 A、附录 B 由李丽娟编写; 第 3 部分的附录 C、附录 D、附录 E 由杨小林编写; 全书由李丽娟统稿。

由于编写时间紧, 编者水平有限, 书中难免存在疏漏之处, 敬请读者批评指正。

编者

2011 年 12 月于岳麓山

目 录

第 1 部分 实 验

第 1 章 函数和宏定义实验	3
1.1 目的和要求	3
1.2 相关知识点	3
1.3 典型案例	4
1.3.1 模块功能设计 1	4
1.3.2 模块功能设计 2	5
1.4 实验内容	5
1.4.1 函数的定义和调用	5
1.4.2 模块化程序设计	11
1.4.3 函数的递归调用	16
1.5 程序案例	19
第 2 章 数组实验	22
2.1 目的和要求	22
2.2 相关知识	22
2.2.1 一维数组	22
2.2.2 多维数组	22
2.2.3 字符数组与字符串	23
2.3 实验内容	23
2.3.1 一维数组的应用	23
2.3.2 二维数组的应用	26
2.3.3 字符数组应用	29
2.4 程序案例	31
第 3 章 指针实验	42
3.1 目的和要求	42
3.2 相关知识点	42
3.3 实验内容	44
3.3.1 指针基础及指针运算	44
3.3.2 数据交换	45
3.3.3 字符串反转及字符串 连接	46
3.3.4 数组元素奇偶排列	48
3.4 综合练习	49
第 4 章 构造数据类型实验	52
4.1 目的和要求	52
4.2 相关知识	52
4.2.1 结构型变量的定义和 引用	52
4.2.2 共用型	53
4.2.3 枚举型	54
4.2.4 用户自定义类型	54
4.3 实验内容	54
4.3.1 结构体变量的应用	54
4.3.2 结构体数组的应用	56
4.3.3 共用体的应用	58
4.3.4 结构体指针的应用	60
4.4 程序案例	62
第 5 章 文件操作实验	74
5.1 目的和要求	74
5.2 相关知识点	74
5.2.1 文件概述	74
5.2.2 文件型指针	75
5.2.3 操作文件的常用函数	75
5.2.4 典型案例	75
5.3 实验内容	78
5.3.1 文本文件的读操作	78
5.3.2 文本文件的读写操作	80
5.3.3 二进制文件的读写操作	81
5.3.4 文件指针的定位操作	84
5.4 程序案例	85
5.4.1 程序案例一	85
5.4.2 程序案例二	89

第 6 章 位运算实验 94	8.1 目的和要求..... 122
6.1 目的和要求..... 94	8.2 相关知识点..... 122
6.2 相关 2 知识点..... 94	8.2.1 类与对象..... 122
6.2.1 位运算的概念..... 94	8.2.2 类成员的访问和作用域..... 123
6.2.2 位运算及位运算赋值操作..... 94	8.2.3 接口和实现分离的设计方法..... 123
6.2.3 位段..... 94	8.2.4 典型案例..... 124
6.2.4 典型案例..... 95	8.3 实验内容..... 126
6.3 实验内容..... 98	8.3.1 类与对象的定义和使用..... 126
6.3.1 按位与运算练习..... 98	8.3.2 通过对象访问类成员..... 127
6.3.2 按位异或运算练习..... 100	8.3.3 通过对象指针访问类成员..... 129
6.3.3 按位或运算练习..... 101	8.3.4 类成员函数重载..... 130
6.3.4 按位取反运算练习..... 102	8.3.5 接口和实现分离的设计方法..... 131
6.4 程序案例..... 103	8.4 程序案例..... 133
6.4.1 程序案例一..... 103	
6.4.2 程序案例二..... 105	
第 7 章 C++ 中新增语法功能实验 107	第 9 章 类与数据抽象 (二) 实验 135
7.1 目的和要求..... 107	9.1 目的和要求..... 135
7.2 相关知识点..... 107	9.2 相关知识点..... 135
7.2.1 新增基本语法..... 107	9.2.1 构造函数和析构函数..... 135
7.2.2 新增函数功能..... 108	9.2.2 对象的赋值..... 136
7.2.3 典型案例..... 109	9.2.3 使用 const 限定对象和成员函数..... 136
7.3 实验内容..... 111	9.2.4 类的静态成员..... 137
7.3.1 引用参数的使用..... 111	9.2.5 典型案例..... 137
7.3.2 函数的默认参数的使用..... 113	9.3 实验内容..... 140
7.3.3 函数重载的使用..... 114	9.3.1 构造函数与析构函数..... 140
7.3.4 函数模板的使用..... 116	9.3.2 友元函数..... 142
7.4 程序案例..... 117	9.3.3 对象的赋值操作..... 143
	9.3.4 类的复合..... 145
	9.3.5 静态成员..... 146
第 8 章 类与数据抽象 (一) 实验 122	9.4 程序案例..... 147

第 2 部分 习题解答

第 1 章 函数与宏定义习题解答 153	第 3 章 指针习题解答 169
第 2 章 数组习题解答 161	第 4 章 构造数据类型习题解答 176

第5章 文件操作习题解答	183	第8章 类与数据抽象(一) 习题解答	202
第6章 位运算习题解答	191	第9章 类与数据抽象(二) 习题解答	213
第7章 C++中新增语法功能习题 解答	196		

第3部分 附 录

附录A Visual C++ 6.0 使用 简介	221	附录D Turbo C 2.0 常见编译 错误	233
附录B Visual C++ 6.0 常见 编译错误	225	D1 致命错误	233
附录C Turbo C 2.0 集成开发 环境简介	227	D2 一般错误	233
		D3 编译警告	242
		附录E Borland C++ 5.0 集成开发 环境简介	245

第 1 部分

实验

- 第 1 章 函数和宏定义实验
- 第 2 章 数组实验
- 第 3 章 指针实验
- 第 4 章 构造数据类型实验
- 第 5 章 文件操作实验
- 第 6 章 位运算实验
- 第 7 章 C++中新增语法功能实验
- 第 8 章 类与数据抽象（一）实验
- 第 9 章 类与数据抽象（二）实验

第 1 章

函数和宏定义实验

1.1 目的和要求

- (1) 掌握函数的定义方法和调用规则。
- (2) 掌握在 C 语言程序中主调函数和被调用函数之间进行数据传递的规则。
- (3) 了解函数的返回值及其类型，并正确使用它。
- (4) 了解局部变量和全局变量的作用域及它们与存储分类的关系，理解变量的存在性和可见性的概念。
- (5) 练习递归函数的使用。
- (6) 理解宏的概念，掌握定义无参宏和带参宏的方法。了解文件包含的概念，掌握其用法。
- (7) 理解内部函数和外部函数，了解外部函数的编译和连接的方法。

1.2 相关知识

1. 函数的概念

函数是一个可以反复使用的程序段，在其他的程序段中均可以通过调用语句来执行这段程序，完成既定的工作。

从程序设计的角度来看，函数可以分为以下两种：

- ① 标准函数，即库函数；
- ② 自定义函数。

从函数形式的角度来看，函数也可分为无参函数和有参函数两种。

2. 函数的定义

一个函数在被调用之前必须先定义，函数定义的一般形式为

```
存储类型 数据类型 函数名(形式参数表)
{ 数据定义语句序列;
  执行语句序列;
}
```

3. 函数的调用

函数调用按是否有返回值分为有返回值的函数调用和无返回值的函数调用。其格式如下：

函数名(实参表);

无返回值的函数调用格式，最后要有一个语句结束符“分号”。

4. 函数的声明

函数声明是指在主调函数中，对在本函数中将要被调用的函数提前做必要的声明。函数声明的一般格式为

存储类型 数据类型 函数名(形式参数表);



① 当函数定义在前，主调函数的定义在后时，可以不需要函数声明。

② 如果被调用的自定义函数和主调函数不在同一文件中，则应在定义函数的文件中将该函数定义成 extern，在主调函数的函数体中将该函数说明为 extern。

5. 函数调用中的数据传递方法

C 语言规定在函数间传递数据有 4 种方式：值传递方式、地址传递方式、返回值方式和全局变量传递方式。

① 值传递方式所传递的是参数值，其特点是“参数值的单向传递”。

② 地址传递方式所传递的是地址，其特点是“参数值的双向传递”。

③ 返回值方式不是在形式参数和实际参数之间传递数据，而是通过函数调用后直接返回一个值到主调函数中。该函数的数据类型不能是 void 类型，且函数体中应有“return <表达式>”语句。

④ 全局变量传递方式不是在形式参数和实际参数之间传递数据，而是利用在主调函数和被调函数中均有有效的全局变量，在主调函数和被调函数之间任意传递数据。

6. 函数的嵌套调用和递归调用

① 在调用一个函数的过程中又调用另一个函数，便形成了函数之间的嵌套调用。

② 一个函数直接或间接地调用其自身，便构成了函数的递归调用。这种函数称为递归函数。

1.3 典型案例

1.3.1 模块功能设计 1

编写求两整数的绝对值的和与差的函数。

分析：在主函数中，输入两个整数，使用 abs()函数求两整数的绝对值。调用 add()函数求两整数的和。调用 sub()函数求两整数的差。在求差的函数中，利用 if 语句找出两数中的较大者并进行减法运算。

程序如下：

```
#include "stdio.h"
#include "math.h"
int add(int c, int d);
int sub(int x, int y)
{   if(x>=y)
    return x-y;
```

```

        else
            return y-x;
    }
void main()
{
    int a,b;
    int sum,minus;
    printf("Input two integers please: ");
    scanf("%d%d",&a,&b);
    a=abs(a);
    b=abs(b);
    sum=add(a,b);
    minus=sub(a,b);
    printf("%d\n",sum);
    printf("%d\n",minus);
}
int add(int c, int d){
return c+d;
}

```

1.3.2 模块功能设计 2

编写一个计算 $f(x) = x^n$ 的递归程序。

(1) 解题思路:

根据函数建立递归式子: $f(x) = x * f(n-1)$ 。

(2) 程序模板:

```

#include <stdio.h>
long int f(int x,int n)
{
    if(n==1) return x;
    return x*f(x,n-1);
}
main()
{
    int x,n;
    printf("Enter x,n please: ");
    scanf("%d%d",&x,&n);
    printf("%ld\n",f(x,n));
}

```

1.4 实验内容

1.4.1 函数的定义和调用

实验练习 1 编写由三角形三边求面积的函数。

1. 实验目的

本实验旨在巩固学生对函数概念的理解,增强程序设计能力。在这个实验中,学生将练习:

- 调用 area()函数求三角形的面积;
- 在求面积函数中运用海伦公式。

2. 问题描述

编写程序，从键盘输入三角形的3条边，调用三角形面积函数求出其面积，并输出结果。程序流程图如图1-1所示。

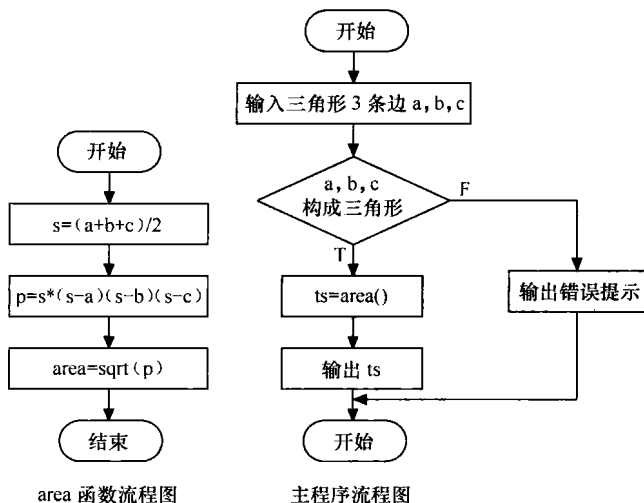


图 1-1 实验练习 1 流程图

3. 示例输出

```
3, 4, 5 (回车)
area=6.000000
```

4. 程序实现 (根据注释完善程序)

```
#include<math.h>
#include<stdio.h>
float area(float a,float b,float c)
{
    float s,p,area;
    _____/*根据海伦公式求三角形面积*/
    return (area);
}
main()
{
    float x,y,z,ts;
    _____ /*键盘输入三角形的3条边*/
    _____ /*判断是否构成三角形*/
    _____ /*调用求三角形面积函数*/

    printf("area=%f\n",ts);
    else printf("data error!");
}
}
```

5. 问题解答提示

(1) 程序要用到开方函数，将头文件 math.h 包含进来。

(2) 要判断输入的3条边是否能构成三角形。如果不能，输出错误提示。

(3) 根据输入的3条边的长度，可由海伦公式求出其面积。设 $p = (a + b + c)/2$ ，三角形的面积等于 $\sqrt{p(p-a)(p-b)(p-c)}$ 。

(4) 求面积函数的类型是浮点型，它是有返回值的，不能写成调用语句。而是把函数调用当做表达式，可把它放在表达式能出现的任何地方。

实验练习 2 编写求 N 阶乘的函数。

1. 实验目的

本实验旨在巩固学生对函数概念的理解，增强程序设计能力。在这个实验中，学生将练习：

- 定义符号常量；
- 使用长整型变量存放累乘积；
- 在函数中，使用局部静态变量存放累乘积；
- 使用全局变量存放累乘积。

2. 问题描述

编写函数，求出从主调函数传来的数值 i 的阶乘值，然后将其传回主调函数并输出。程序流程图如图 1-2 所示。

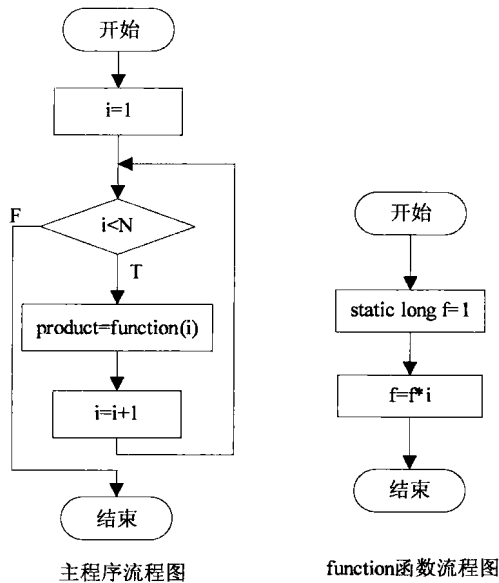


图 1-2 实验练习 2 流程图

3. 示例输出

```

1 的阶乘是: 1
2 的阶乘是: 2
3 的阶乘是: 6
4 的阶乘是: 24
5 的阶乘是: 120
  
```

4. 程序实现 (根据注释完善程序)

```

#include "stdio.h"
_____ /*定义符号常量 N, 代表数字 5*/
long function(int i)
{ _____ /*定义局部静态变量 f 并赋初值 1*/
  _____ /*求形参 i 的阶乘值并存放在 f 中*/
  return f;
}
  
```

```

}
void main()
{   long product;
    int i;
    for(i=1; i<=N; i++)
    {
        _____/*调用函数 function()求阶乘值, 并赋值给 product */
        _____/*输出*/
    }
}

```

5. 问题解答提示

(1) 定义符号常量一方面可以增强程序的可读性, 另一方面可以根据需要修改符号常量的值来求不同数的阶乘值, 使程序具有通用性。

(2) int 型变量占 2 个字节的存储空间, 当求的值太大时就存放不下, 所以要用长整型数来存放, long 型变量占 4 个字节的存储空间。

(3) 使用循环语句依次求出 1 ~ N 的阶乘值。

(4) 在循环体中, 每调用一次求阶乘函数就能求出指定值的阶乘值。

(5) 求阶乘函数的类型是长整型, 它是有返回值的, 不能写成调用语句。而是把函数调用当做表达式, 可把它放在表达式能出现的任何地方。

(6) 局部静态变量有全局的寿命和局部的可见性, 退出 function() 函数它是不可见的, 进入 function() 函数它又可见。这说明该局部静态变量没有被释放。

(7) 局部静态变量的值具有继承性, 利用这一特点可以依次求出 1 ~ N 的阶乘值。

实验练习 3 求两个整数的最大公约数。

1. 实验目的

本实验旨在巩固学生对函数概念的理解, 增强程序设计能力。在这个实验中, 学生将练习:

- 调用 gcd() 函数求两个整数的最大公约数;
- 掌握辗转相除法求两个整数的最大公约数。

2. 问题描述

编写程序, 从键盘输入两个整数, 调用 gcd 函数求它们的最大公约数并输出结果。程序流程图如图 1-3 所示。

3. 示例输出

```

please input two integers:64,72
The great common divisor is:8

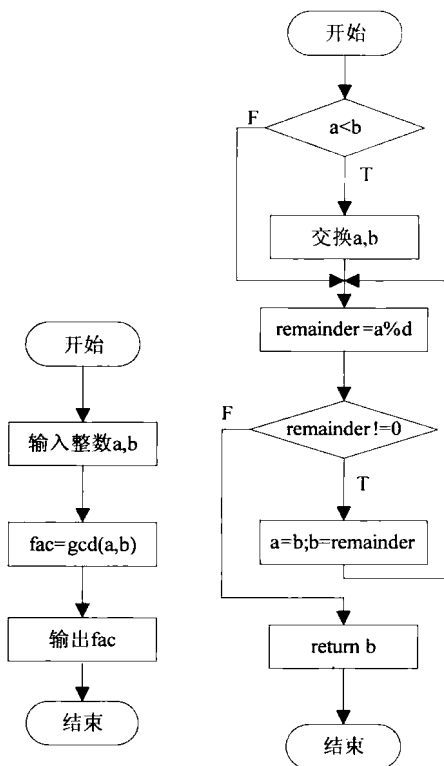
```

4. 程序实现 (根据注释完善程序)

```

#include<stdio.h>
int gcd(int a,int b)
{ int temp;

```



主程序流程图

area函数流程图

图 1-3 实验练习 3 流程图

```

int remainder;
if(a<b){
    _____/*交换 a 和 b 的值*/
}
remainder=a%b;
while(remainder!=0)
{
    _____/*辗转相除求最大公约数*/
}
return b;
}
main()
{int x,y;
int fac;
    _____/*提示输入两个整数*/
    _____/*输入两个整数*/
    _____/*用输入的两个数调用求最大公约数的函数*/
printf("The great common divisor is:%d",fac);
}

```

5. 问题解答提示

(1) 程序的关键点是用辗转相除法求两个整数的最大公约数。例如, $a > b$, 如果 a 能被 b 整除, 最大公约数就是 b 。如果 a 除 b 的余数为 c , 则继续用 b 除 c , 如此反复, 直到余数为 0, 则最后一个非 0 除数就为 a 、 b 的最大公约数。

(2) 求最大公约数函数的类型是整型, 它是有返回值的, 不能写成调用语句, 而是把函数调用当做表达式, 可把它放在表达式能出现的任何地方。

实验练习 4 打印输出指定图形。

1. 实验目的

本实验旨在巩固学生对函数概念的理解, 增强程序设计能力。在这个实验中, 学生将练习:

- 调用 `trangle()` 函数输出三角形;
- 在 `trangle()` 函数中用 `for` 循环的嵌套输出指定的结果。

2. 问题描述

输入整数 n , 输出高度为 n 的等边三角形。当 $n = 5$ 时的等边三角形如下:

```

*
***
*****
*****
*****

```

程序流程图如图 1-4 所示。

3. 程序实现 (根据注释完善程序)

```

#include <stdio.h>
void trangle(int n)
{ int i,j;
  for (i=0;i<n;i++)
  {
    _____/*打印每一行的空格*/

```



```

        _____ /*打印每一行的*号*/
        putchar('\n');
    }
}
main()
{ int n;
  _____ /*提示输入一个整数 n*/
  _____ /*输入整数 n*/
  printf("\n");
  _____ /*调用函数打印出等边三角形*/
}

```

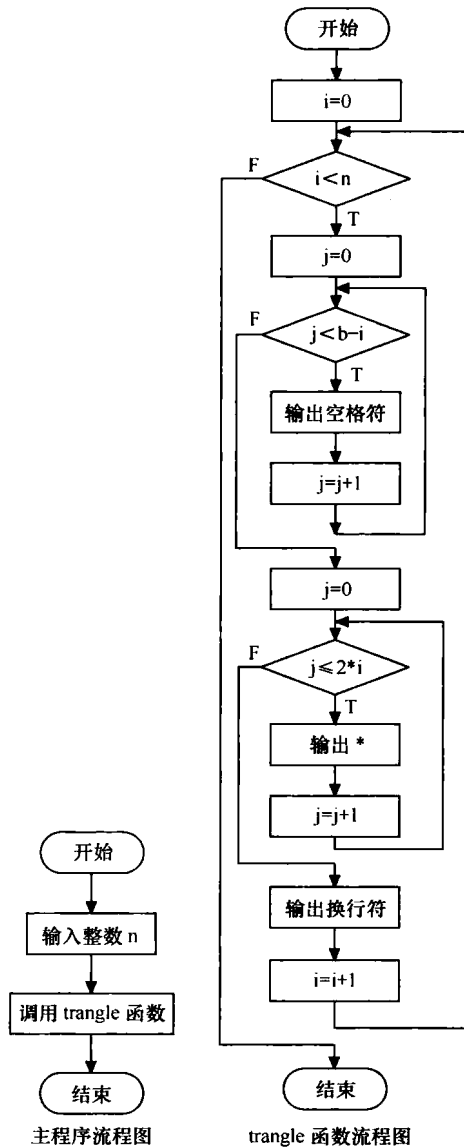


图 1-4 实验练习 4 流程图

4. 问题解答提示

由给出的示例可以看出等边三角形的组成规律，每一行*号前的空格数都比上一行少一个，每