

普.通.高.等.学.校
计算机教育“十三五”规划教材

C 语言程序设计教程 实验指导与习题解答

(第5版)

*THE ANSWER AND PRACTICE OF
THE C PROGRAMMING LANGUAGE
(5th edition)*

李丽娟 ◆ 主编

吴蓉晖 杨小林 谷长龙 洪跃山 李根强 杜四春 ◆ 副主编



普通高等院校
计算机教育“十三五”规划教材

C 语言程序设计教程 实验指导与习题解答

(第 5 版)

THE ANSWER AND PRACTICE OF
THE C PROGRAMMING LANGUAGE
(5th edition)

李丽娟 ◆ 主编

吴蓉晖 杨小林 谷长龙 洪跃山 李根强 杜四春 ◆ 副主编



人民邮电出版社

北京

图书在版编目(CIP)数据

C语言程序设计教程实验指导与习题解答 / 李丽娟主编. — 5版. — 北京: 人民邮电出版社, 2019.7
普通高等学校计算机教育“十三五”规划教材
ISBN 978-7-115-49580-8

I. ①C… II. ①李… III. ①C语言—程序设计—高等学校—教学参考资料 IV. ①TP312.8

中国版本图书馆CIP数据核字(2018)第227932号

内 容 提 要

本书是《C语言程序设计教程(第5版)》配套的实验指导与习题解答教程。

全书分为3个部分。第1部分为实验指导,针对《C语言程序设计教程(第5版)》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分析与提示。第2部分为习题解答,包括《C语言程序设计教程(第5版)》各章的习题和参考答案,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 Studio 2010、Turbo C 2.0、Dev C++ 5.11集成环境下程序的编辑、编译连接和运行的方法。

全书内容紧扣C语言程序设计的相关知识点,案例丰富,实用性强,可作为学习C语言程序设计的配套教程,也可供一般工程技术人员参考。

-
- ◆ 主 编 李丽娟
 - 副 主 编 吴蓉晖 杨小林 谷长龙 洪跃山 李根强 杜四春
 - 责任编辑 邹文波
 - 责任印制 陈 犇
 - ◆ 人民邮电出版社出版发行 北京市丰台区成寿寺路11号
邮编 100164 电子邮件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
三河市中晟雅豪印务有限公司印刷
 - ◆ 开本: 787×1092 1/16
印张: 16.25 2019年7月第5版
字数: 425千字 2019年7月河北第1次印刷
-

定价: 42.00元

读者服务热线: (010)81055256 印装质量热线: (010)81055316

防盗版热线: (010)81055315

广告经营许可证: 京东工商广登字 20170147号

在学习 C 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，出现这种想法的主要原因是缺乏足够的实践与练习。为了帮助读者学好 C 语言，真正掌握用 C 语言进行程序设计的方法，我们特别在《C 语言程序设计实验指导与习题解答（第 4 版）》的基础上，结合多年的教学经验，特别是多年教学改革的实践，优化、补充和完善内容，编写了《C 语言程序设计教程实验指导与习题解答（第 5 版）》，希望能使读者在学习 C 语言时有更好的用户体验。

本书分为 3 个部分。第 1 部分为实验指导，共有 11 章，与《C 语言程序设计教程（第 5 版）》中的章节相对应，每一章都简要地提出学习目的和要求，列出相关知识点和案例，并安排了相关的实验内容。

第 2 部分为习题解答，是对《C 语言程序设计教程（第 5 版）》中所有习题的解答。习题覆盖《全国计算机等级考试大纲》（C 语言部分）的所有内容，共有 11 章，与《C 语言程序设计教程（第 5 版）》中的章节完全对应，并对部分疑难问题给出一些解答提示。对于编程题，参考答案并不是唯一的，给出参考答案仅仅是为了起到抛砖引玉的作用，希望读者独立思考，将计算思维的方法应用于程序设计，设计出不同的算法实现，以拓宽自己的思路，掌握 C 语言的基本概念和编程方法。

第 3 部分为附录，简单介绍了 Visual Studio 2010、Turbo C 2.0 和 Dev C++ 5.11 集成开发环境的使用，帮助读者快速掌握 C 语言程序的调试方法。

书中所有程序均在 Visual Studio 2010 环境和 Dev C++ 5.11 环境下调试并编译通过。

本书由李丽娟担任主编，吴蓉晖、杨小林、谷长龙、洪跃山、李根强、杜四春担任副主编。编写分工如下：第 1 部分的第 1 章、第 3 章和第 2 部分的第 1 章、第 3 章、附录 A、附录 B 由李丽娟编写；第 1 部分的第 2 章、第 4 章和第 2 部分的第 2 章、第 4 章由杨小林和谷长龙编写；附录 C 由杨小林编写；第 1 部分的第 5 章、第 6 章和第 2 部分的第 5 章、第 6 章由吴蓉晖编写；第 1 部分的第 8 章和第 2 部分的第 8 章由洪跃山编写；第 1 部分的第 7 章、第 9 章和第 2 部分的第 7 章、第 9 章由李根强编写；第 1 部分的第 10 章、第 11 章和第 2 部分的第 10 章、第 11 章由杜四春编写。全书由李丽娟统稿。

由于编者水平有限，书中难免存在不足之处，敬请读者批评指正。

编者

2019 年 3 月于岳麓山

目 录

第 1 部分 实验指导

第 1 章 实验引言 2

- 1.1 学习目的和要求 2
- 1.2 相关知识点 2
- 1.3 实验内容 2
 - 1.3.1 熟悉 Turbo C 环境 2
 - 1.3.2 熟悉 Visual Studio 2010 环境 4
 - 1.3.3 熟悉 Dev C++ 环境 5
 - 1.3.4 熟悉 Borland C++ 环境 6
- 1.4 C 语言开发环境综合练习 8

第 2 章 基本的程序语句实验 9

- 2.1 学习目的和要求 9
- 2.2 相关知识点 9
 - 2.2.1 常量和变量 9
 - 2.2.2 算术表达式 9
 - 2.2.3 运算 10
 - 2.2.4 编程中应注意的事项 10
 - 2.2.5 典型案例 10
- 2.3 实验内容 12
 - 2.3.1 表达式的计算 12
 - 2.3.2 表达式的应用 13
 - 2.3.3 字符与 ASCII 13
 - 2.3.4 运算符与表达式的应用 14
 - 2.3.5 顺序结构应用程序 15
 - 2.3.6 分析程序 16
- 2.4 程序案例 17
 - 2.4.1 程序案例一 17
 - 2.4.2 程序案例二 18

第 3 章 程序的简单算法设计实验 20

- 3.1 学习目的和要求 20
- 3.2 相关知识点 20

3.3 实验内容 20

- 3.3.1 数学函数的算法描述 20
- 3.3.2 鸡兔同笼的算法描述 21
- 3.3.3 确定坐标的算法描述 22

第 4 章 分支结构实验 24

- 4.1 学习目的和要求 24
- 4.2 相关知识点 24
 - 4.2.1 if 语句 24
 - 4.2.2 switch 语句 24
 - 4.2.3 编程中应注意的事项 25
 - 4.2.4 典型案例 25
- 4.3 实验内容 28
 - 4.3.1 if 语句的应用 28
 - 4.3.2 switch···case 语句的应用 28
 - 4.3.3 switch···case 语句嵌套 if 语句的应用 29
 - 4.3.4 switch···case 结构嵌套的应用 30
 - 4.3.5 分析程序 32
- 4.4 程序案例 32
 - 4.4.1 程序案例一 32
 - 4.4.2 程序案例二 33

第 5 章 循环结构实验 36

- 5.1 学习目的和要求 36
- 5.2 相关知识点 36
 - 5.2.1 while 语句 36
 - 5.2.2 do···while 语句 37
 - 5.2.3 for 语句 37
 - 5.2.4 典型案例 38
- 5.3 实验内容 40
 - 5.3.1 用 for 语句实现循环 40
 - 5.3.2 用 while 语句实现循环 43

5.3.3 用 do···while 语句实现循环	44	7.4.3 程序案例三	85
5.3.4 用 while 语句和 for 语句配合 实现循环	45	7.4.4 程序案例四	87
5.3.5 用 for 语句嵌套实现循环	47	7.4.5 程序案例五	87
5.4 程序案例	51	第 8 章 指针实验	91
5.4.1 程序案例一	51	8.1 学习目的和要求	91
5.4.2 程序案例二	52	8.2 相关知识点	91
第 6 章 函数与宏定义实验	55	8.3 实验内容	93
6.1 学习目的和要求	55	8.3.1 指针基础及指针运算	93
6.2 相关知识点	55	8.3.2 数据交换	94
6.2.1 函数的概念	55	8.3.3 字符串反转及字符串连接	95
6.2.2 函数的定义形式	55	8.3.4 数组元素奇偶排列	97
6.2.3 函数的调用	56	8.4 程序案例	98
6.2.4 函数的声明	56	8.4.1 程序案例一	98
6.2.5 函数调用中的数据传递方法	56	8.4.2 程序案例二	99
6.2.6 函数的嵌套调用和递归调用	56	8.4.3 程序案例三	99
6.2.7 典型案例	56	第 9 章 构造数据类型实验	101
6.3 实验内容	57	9.1 学习目的和要求	101
6.3.1 函数的定义和调用	57	9.2 相关知识点	101
6.3.2 模块化程序设计	62	9.2.1 结构型变量的定义和引用	101
6.3.3 函数的递归调用	68	9.2.2 共用型变量的定义和引用	102
6.4 程序案例	71	9.2.3 枚举型变量的定义和引用	102
6.4.1 程序案例一	71	9.3 实验内容	103
6.4.2 程序案例二	73	9.3.1 结构型变量的应用	103
6.4.3 程序案例三	74	9.3.2 结构型数组的应用	105
第 7 章 数组实验	75	9.3.3 共用体的应用	106
7.1 学习目的和要求	75	9.3.4 结构体指针的应用	108
7.2 相关知识点	75	9.4 程序案例	110
7.2.1 一维数组	75	9.4.1 程序案例一	110
7.2.2 多维数组	75	9.4.2 程序案例二	113
7.2.3 字符数组与字符串	76	9.4.3 程序案例三	116
7.3 实验内容	76	9.4.4 程序案例四	117
7.3.1 一维数组的应用	76	9.4.5 程序案例五	118
7.3.2 二维数组的应用	78	第 10 章 文件操作实验	123
7.3.3 字符数组的应用	80	10.1 学习目的和要求	123
7.4 程序案例	82	10.2 相关知识点	123
7.4.1 程序案例一	82	10.2.1 文件概述	123
7.4.2 程序案例二	84	10.2.2 文件型指针	124

10.2.3 操作文件的常用函数.....	124	11.2 相关知识点.....	144
10.2.4 典型案例.....	124	11.2.1 位运算的概念.....	144
10.3 实验内容.....	127	11.2.2 位运算及位运算赋值操作.....	144
10.3.1 文本文件的读操作.....	127	11.2.3 位段.....	144
10.3.2 文本文件的读/写操作.....	129	11.2.4 典型案例.....	145
10.3.3 二进制文件的读/写操作.....	131	11.3 实验内容.....	148
10.3.4 文件指针的定位操作.....	133	11.3.1 按位与运算练习.....	148
10.3.5 文件的加密操作.....	134	11.3.2 按位异或运算练习.....	149
10.4 程序案例.....	136	11.3.3 按位或运算练习.....	149
10.4.1 程序案例一.....	136	11.3.4 按位取反运算练习.....	150
10.4.2 程序案例二.....	139	11.4 程序案例.....	151
第 11 章 位运算实验	144	11.4.1 程序案例一.....	151
11.1 学习目的和要求.....	144	11.4.2 程序案例二.....	152

第 2 部分 习题解答

第 1 章 引言习题解答.....	155	第 6 章 函数与宏定义习题解答.....	186
第 2 章 基本的程序语句习题解答.....	157	第 7 章 数组习题解答.....	195
第 3 章 程序的简单算法设计习题 解答.....	161	第 8 章 指针习题解答.....	209
第 4 章 分支结构习题解答.....	166	第 9 章 构造数据类型习题解答.....	216
第 5 章 循环结构习题解答.....	172	第 10 章 文件操作习题解答.....	223
		第 11 章 位运算习题解答.....	232

第 3 部分 附 录

附录 A Visual Studio 2010 使用 简介.....	238	附录 C Dev C++ 5.11 集成开发 环境简介.....	250
附录 B Turbo C 2.0 集成开发环境 简介.....	244		

第 1 部分

实验指导

- 第 1 章 实验引言
- 第 2 章 基本的程序语句实验
- 第 3 章 程序的简单算法设计实验
- 第 4 章 分支结构实验
- 第 5 章 循环结构实验
- 第 6 章 函数与宏定义实验
- 第 7 章 数组实验
- 第 8 章 指针实验
- 第 9 章 构造数据类型实验
- 第 10 章 文件操作实验
- 第 11 章 位运算实验

第 1 章

实验引言

1.1 学习目的和要求

(1) 了解不同的 C 语言程序的开发环境，并能在不同的环境下编辑、编译、连接和运行 C 语言程序。

(2) 通过运行简单的 C 语言程序，初步了解 C 语言源程序的特点。

1.2 相关知识点

1. C 语言程序要在特定的开发环境下才能生成可执行的程序，适合 C 语言的开发环境有 Turbo C 2.0、Borland C++、Visual Studio 2010、Dev C++ 5.11、GCC 等。

2. C 语言程序的基本结构由以下 6 部分组成：

- (1) 文件包含部分；
- (2) 预处理部分；
- (3) 变量说明部分；
- (4) 函数原型声明部分；
- (5) 主函数部分；
- (6) 自定义函数部分。

若要生成可执行的程序，其中文件包含部分和主函数部分一般不可以缺少，其他部分可根据程序的需要而定。

3. C 语言源程序是不能直接运行的，必须对源程序进行编译、链接后，才能生成可执行的程序。

1.3 实验内容

1.3.1 熟悉 Turbo C 环境

Turbo C 是 Borland 公司于 1989 年推出的一种 C 语言程序集成开发环境，它将编辑、编译、链接和运行集成在一起，使用起来方便简单，具有效率高、功能强等优点，其中的 Turbo C 2.0 要

求 DOS (磁盘操作系统) 为 2.0 或更高版本。

实验练习 安装 Turbo C 2.0, 并在该环境下编辑如下简单的 C 语言程序:

```
#include <stdio.h>
main()
{
    printf("Hello,World!\n");
}
```

对其进行编译, 生成可执行的程序。执行该程序, 观察程序的运行结果。

1. 实验目的

本实验旨在巩固学生对编程基本概念的理解。在这个实验中, 学生将学习:

- Turbo C 环境下 C 语言程序的开发;
- printf()函数的基本用法。

在强化练习中, 学生将练习如何输出指定的信息。

2. 问题描述

请对上面的程序进行修改, 要求输出如下信息:

```
I am learning The C program.
Very good!
```

3. 输出示例

```
I am learning The C program.
Very good!
```

4. 程序样板

请在以下程序的空白处填入合适的内容。

```
#include <stdio.h>
main()
{
    printf("_____ ① _____");
    printf("_____ ② _____");
}
```

5. 问题解答与提示

(1) 启动 Turbo C 系统, 进入 Turbo C 集成开发环境。

(2) 在主菜单下, 选择“File”菜单下的“New”命令(建立新文件), 进入 Turbo C 编辑状态。

(3) 在编辑窗口输入上述程序。

6. 强化练习

根据程序模板, 现要求输出如下信息:

```
*****
                          Very good
*****
```

请写出相应的源程序。

7. 实验结果小结

(1) 根据使用过程, 回答下面的问题:

- ① 怎样进入程序的编辑状态?
- ② 使用什么命令对源程序进行编译?
- ③ 怎样生成可执行的文件?

④ 怎样查看程序的运行结果?

(2) 给出 Turbo C 2.0 两个不同状态下的界面图, 如编辑状态、编译状态、环境设置状态、文件打开状态等。

1.3.2 熟悉 Visual Studio 2010 环境

Visual Studio 2010 (简称 VS 2010), 是微软公司推出的一款集成开发平台。该编译器同样也适合编写 C 语言, 它是一个基于 Windows 操作系统的可视化集成开发环境 (Integrated Development Environment, IDE)。

实验练习 1 安装 Visual Studio 2010, 并在该环境下重做 1.3.1 小节的练习。

实验练习 2 在 Visual Studio 2010 环境下, 编辑如下简单的 C 语言程序:

```
#include <stdio.h>
void main()
{
    int a,b,c;
    printf("Please enter a,b:");
    scanf("%d,%d",&a,&b);
    c=a+b;
    printf("c=%d\n",c);
}
```

1. 实验目的

本实验旨在巩固学生对编程基本概念的理解。在这个实验中, 学生将学习:

- 算术运算表达式的写法;
- printf()函数的基本用法;
- scanf()函数的基本用法。

在强化练习中, 学生将练习整数的减法运算和乘法运算。

2. 问题描述

编写程序, 计算两个整数的和与积。

3. 输入/输出示例

```
Please enter a,b: 3,5↵
3 + 5 = 8
3*5 = 15
```

注: “↵”表示输入一个换行符(回车), 为不可见字符。

4. 程序样板

以下程序需要输出表达式的计算结果, 请在以下程序的空白处填入合适的内容, 将表达式的计算结果输出到屏幕。运行程序, 观察程序的运行结果。

```
#include <stdio.h>
void main()
{
    int a,b,c,d;
    printf("Please enter a,b: ");
    scanf("%d,%d",&a,&b);
    c=a+b;
    d=a*b;
    printf("____ ① ____");
```

```
printf("_____②");
}
```

5. 问题解答与提示

(1) 程序的功能是进行加法运算和乘法运算, 因此需要从键盘输入两个变量的值。

(2) 观察示例输出 $a+b$ 和 $a*b$ 的结果可知, 程序进行的是整型数据的计算, 因此变量类型均为整型。

(3) 用 `scanf` 语句接收数值, 并存储在变量 a 、 b 中。

(4) 分别将加法运算和乘法运算的结果赋予不同的变量保存起来。

(5) 输出计算结果。

6. 强化练习

如果要计算两个整数的差, 上述程序又将如何修改呢? 请写出正确程序。

7. 实验结果小结

(1) 根据使用过程, 回答下面的问题:

- ① 怎样进入程序的编辑状态?
- ② 使用什么命令对源程序进行编译?
- ③ 怎样生成可执行的文件?
- ④ 怎样查看程序的运行结果?

(2) 给出 Visual Studio 2010 两个不同状态下的界面图, 如编辑状态、编译状态、环境设置状态、文件打开状态等。

1.3.3 熟悉 Dev C++ 环境

Dev C++ 是一个 Windows 环境下适合 C/C++ 语言开发的工具。它是一款自由软件, 非常实用, 能满足初学者与编程高手的不同需求。

实验练习 1 安装 Dev C++ 5.11 (或者 Dev C++ 5.0 以上的其他版本), 在该环境下, 重做 1.3.1 小节的练习。

实验练习 2 在 Dev C++ 环境下编辑如下简单的 C 语言程序:

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

对其进行编译、生成可执行的程序, 执行该程序, 观察程序的运行结果。

1. 实验目的

本实验旨在巩固学生对编程基本概念的理解。在这个实验中, 学生将学习:

- Dev C++ 环境下 C 语言程序的开发;
- 运行简单的 C 程序, 初步了解 C 语言源程序的特点;
- 改变 `printf` 中的符号, 输出其他图案。

在强化练习中, 练习用 C 语言的 `printf` 语句输出各种不同的图案。

2. 问题描述

编写程序, 输出一个由字母 A 组成的等腰三角形图案, 三角形的两边空白处用字母 O 填充。

3. 输出示例

```

O O O O A O O O O
O O O O A A O O O
O O O A A A A O O
O O A A A A A A O
O A A A A A A A O
A A A A A A A A A

```

4. 程序样板

请在以下程序的空白处填写你喜欢的字母符号, 使程序输出不同的几何图案, 然后运行程序, 观察程序的运行结果, 了解用 printf 语句输出几何图案的特点。

```

#include <stdio.h>
void main()
{
    printf("____ (1) ____");
    printf("____ (2) ____");
    printf("____ (3) ____");
    printf("____ (4) ____");
    ...
    printf("____ (n) ____");
}

```

5. 问题解答与提示

(1) 根据问题描述进行分析, 程序的主要任务是输出有规律的几何图案。因此, 关键在于要给出图案的形状, 然后用 printf 语句逐行将图案输出。

(2) 通过这种固定方式来输出几何图案, 大小是固定不变的, 除非修改程序的源代码。

6. 强化练习

请在以上程序的基础上, 输出下列几种几何图案: 圆形、五角星、倒三角形。

7. 实验结果小结

(1) 根据使用过程, 回答下面的问题:

- ① 如何进入程序的编辑状态?
- ② 使用什么命令对源程序进行编译?
- ③ 如何生成可执行的文件?
- ④ 如何查看程序的运行结果?

(2) 给出 Dev C++ 两个不同状态下的界面图, 如编辑状态、编译状态、环境设置状态、文件打开状态等。

1.3.4 熟悉 Borland C++ 环境

Borland C++ 使用方便, 是一个 Windows 环境下适合 C/ C++ 语言开发的工具。

实验练习 1 安装 Borland C++ 5.0 (也可以是 Borland C++ 的其他版本), 在该环境下, 重做 1.3.1 小节的练习。

实验练习 2 在 Borland C++ 环境下编辑如下简单的 C 语言程序:

```
#include <stdio.h>
void main()
{
    printf("Data Types and Sizes :\n");
    printf("char: %d\n",sizeof(char));
    printf("float: %d\n",sizeof(float));
    printf("double: %d\n",sizeof(double));
}
```

对其进行编译，生成可执行的程序，执行该程序，观察程序的运行结果。

1. 实验目的

本实验旨在巩固学生对编程基本概念的理解。在这个实验中，学生将学习：

- Borland C++环境下 C 语言程序的开发；
- 运行简单的 C 程序，以便初步了解 C 语言源程序的特点；
- sizeof 运算符的特殊用法。

在强化练习中，学生将了解 C 语言的各种数据类型是否固定不变。

2. 问题描述

编写程序，第 1 行输出字符串“Data Types and Sizes:”，从第 2 行开始打印输出 Borland C++ 环境下 C 语言中各种数据类型存储空间的大小。

3. 输出示例

```
Data Types and Sizes:
long: 4
unsigned int: 2
double: 8
```

4. 程序样板

请在以下程序的空白处随意填写文字或者表达式，然后运行程序，观察程序的运行结果，了解 printf 语句的特点。

```
#include <stdio.h>
void main()
{
    printf("____ ① ____");
    printf("____ ② ____");
    printf("____ ③ ____");
    printf("____ ④ ____");
}
```

5. 问题解答与提示

(1) 根据问题描述进行分析，程序分为两项任务，第 1 项要求输出指定的字符串，第 2 项要求输出 C 语言中各种数据类型存储空间的大小。

(2) 对于第 1 项任务，可以采用 printf() 语句进行编写。

(3) 对于第 2 项任务，首先分析 C 语言中有哪些主要的数据类型 (int 型、char 型、float 型、double 型等)；其次分析如何通过程序计算出这些类型所占用的存储空间 (使用系统提供的求某种数据类型存储空间字节数的函数 sizeof())；最后分析如何将各数据类型所占用的存储空间大小输出。

6. 强化练习

(1) 思考上述类型所占用的存储空间大小是否是固定的？如果不是，你能说出原因吗？

(2) 如果在输出字符串下加一行“-----”，应怎么修改程序？请写出正确程序。

7. 实验结果小结

(1) 根据使用过程，回答下面的问题：

- ① 如何进入程序的编辑状态？
- ② 使用什么命令对源程序进行编译？
- ③ 如何生成可执行的文件？
- ④ 如何查看程序的运行结果？

(2) 给出 Borland C++ 两个不同状态下的界面图，如编辑状态、编译状态、环境设置状态、文件打开状态等。

1.4 C 语言开发环境综合练习

1. 练习安装几种不同的适合 C 语言的集成开发环境，如 Dev C++、Borland C++、VS 2010 等。要求了解和掌握以下内容：

(1) 系统的目录结构；

(2) 系统每一个目录下的文件形式；

(3) 运行该集成编辑环境，调入系统提供的程序范例，练习对该程序的编辑、运行，了解程序的运行机制。

(4) 自己动手编辑一个 C 语言程序并调试运行，掌握基本的程序调试过程。

2. 对一个已经安装好的 Turbo C 2.0 集成编辑环境，完成以下任务：

(1) 改变系统的根目录名；

(2) 运行该集成编辑环境，调入系统提供的程序范例，对该程序进行编译，看看会发生什么问题，寻找解决方案；

(3) 了解集成环境下的其他功能。

第 2 章

基本的程序语句实验

2.1 学习目的和要求

- (1) 掌握 C 语言的基本数据类型, 如整型 (int)、字符型 (char)、单精度型 (float)、双精度型 (double), 以及由这些基本类型构成的常量和变量的使用方法。
- (2) 掌握 C 语言的运算符和表达式的正确使用。
- (3) 掌握自增 (++) 和自减 (--) 运算符的正确使用。
- (4) 掌握常见输入/输出函数 scanf()、getchar()、printf() 和 putchar() 的基本功能、调用方法、输入/输出格式控制规定等。
- (5) 能独立编写顺序结构的程序并调试通过。
- (6) 用表达式语句、输入/输出语句解决简单的应用问题并上机实现。

2.2 相关知识点

2.2.1 常量和变量

常量和变量是 C 语言数据处理的基本对象。

1. 常量分为数值常量、字符常量和字符串常量

数值常量分为整型常量和实型常量两种, 分别表示整数和实数。整数可以使用十进制数、八进制数和十六进制数表示; 实数只使用十进制数的小数形式和指数形式具体表示。

字符常量由一对单引号括起来的一个字符表示。字符常量的值是该字符所对应的 ASCII 值。

字符串常量由一对双引号括起来的一串字符表示。字符串在内存存储时, 自动在其尾部加上一个空 (NULL) 字符。

2. 变量的基本数据类型

变量的基本数据类型有字符型、整型、单精度浮点型和双精度浮点型, 根据基本数据类型的数据长度及有无符号又可派生出多种类型。

2.2.2 算术表达式

算术表达式是用算术运算符和括号将运算对象连接起来, 且符合 C 语言语法规则的式子。

C 语言在进行不同数据类型的混合运算时, 会按规则将不同数据类型的数据转换成相同的数据类型后再进行运算, 即 C 语言在编译时会自动把参与运算的各种常量和变量转换为其中间数据长度最长的数据类型, 但赋值操作按把赋值号右边类型转换成赋值号左边类型的原则进行。如果希望在一个算术表达式中按用户的要求确定结果的类型, 就要采用强制类型转换符号进行操作。

实际应用中, 应尽可能避免出现数据类型转换的表达式, 设计的表达式应简洁明了。

2.2.3 运算

1. 算术运算

单目运算: $-$, $++$, $--$ 。

双目运算: $+$, $-$, $*$, $/$, $\%$ 。

2. 关系与逻辑运算

关系运算的结果是 1 或 0, 若关系表达成立, 则其值为 1, 否则为 0。

逻辑运算的结果是真或假, 如果为真, 定义其值为 1, 如果为假, 定义其值为 0。

2.2.4 编程中应注意的事项

1. 用整型 (int、short、long) 说明变量时, 要注意变量的取值范围, 否则将出现意想不到的错误。

2. 运算符的优先级: 关系运算符中的等号 ($=$) 不要写成赋值号; 逻辑运算中, 当有 $\&\&$ 和 $\|\|$ 符号连接的表达式时, 应按从左至右的顺序进行运算, 且一旦能确定结果, 便终止后面的运算。

2.2.5 典型案例

案例一 面积的计算

1. 问题描述

输入 r_1 、 r_2 , 求出图 2-1 中圆形垫片的面积 (网格部分)。

2. 具体分析

(1) 所求面积 s = 大圆面积 s_2 (半径 r_2) - 小圆面积 s_1 (半径 r_1)。

(2) 圆面积 = 半径 \times 半径 $\times 3.14$ 。

(3) 程序接收 r_1 、 r_2 , 定义为 float 型。

其算法流程图如图 2-2 所示。

3. 程序样板

```

/*sy2_1.c*/
#include <stdio.h>
#define P 3.14
int main()
{
    float r1,r2;
    double s1,s2,s;
    printf("Please enter r1,r2:\n" );
    scanf("%f%f",&r1,&r2);
    s2=r2*r2*P;
    s1=r1*r1*P;
    s=s2-s1;
}

```

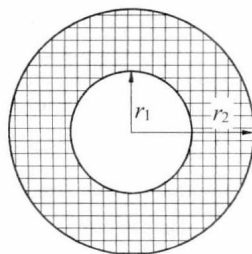


图 2-1 圆形垫片

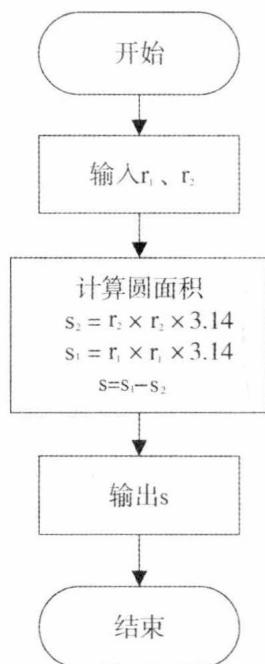


图 2-2 案例一的算法流程图