

段江涛

计算机导论与程序设计 [CS006001]

机试练习参考程序代码

2020 年 10 月 6 日

Foreword

本机试练习参考程序代码供学习《计算机导论与程序设计》课程的学生使用。希望同学们认真阅读总结代码中体现的课程知识点，并比对自己的程序，建议从以下几个方面总结程序设计要领：

1. 通过大量编程实践，训练自己的逻辑思维能力。
2. 学习结构化、模块化程序设计技术。
3. 重点题型：字符串处理，数组（矩阵）应用，结构体数据组织以及循环迭代。
4. 数组应用注意越界问题，字符串注意‘\0’问题。
5. 循环迭代基本思想是根据变量的上一轮值计算下一轮的值，注意进入内层循环前的初始化问题。
6. 各种类型的排序程序必须熟练掌握，包括整数、字符、字符串、结构体等数组类型的排序，冒泡或选择法排序均可。
7. 课件及本代码集是复习主要参考资料，融汇贯通整个课程内容。
8. 准备自己的函数库或程序片段，做到信手拈来。
9. 学习程序调试技巧，每写好一段程序框架就编译运行一次，排除低级语法错误。千万不要整个程序写完才开始编译调试。
10. 采用模块化分段调试技术，每完成一部分功能，调试该部分功能，保证其正确性。不吝惜使用输出语句，观察程序的执行结果是否与所设想的一致。测试输入变量的正确值，是调试程序的首要点，尤其是数字，字符，字符串混合形式的输入。

Xi'an, China, December, 2019

段江涛

目录

1	第 1 次机试 (课外) 练习	7
1.1	计算球体重量	7
1.2	温度转化	8
1.3	整数简单运算	8
1.4	A+B+C	9
1.5	字符输入输出	10
1.6	数字字符	10
1.7	实数运算	11
1.8	四则运算	11

Chapter 1

第 1 次机试 (课外) 练习

1.1 计算球体重量

已知铁的比重是 7.86(克/立方厘米), 金的比重是 19.3(克/立方厘米)。写一个程序, 分别计算出给定直径的铁球与金球的质量, 假定 $\text{PI}=3.1415926$

输入说明:

输入两个整数, 分别表示铁球与金球的直径 (单位为毫米)

输出说明:

输出两个浮点数, 分别表示铁球与金球的质量 (单位为克), 小数点后保留 3 位小数, 两个浮点数之间用空格分隔

输入样例:

100 100

输出样例:

4115.486 10105.456

提示:

用scanf输入, 用printf输出, 保留 3 位小数的格式控制字符为%.3f

```
#include <stdio.h>
#include <math.h>    // 数学库函数
#define PI 3.1415926
int main()
{
    int a,b;
    scanf("%d%d",&a,&b);
    float v1= 4.0/3.0*pow(a/2.0/10,3)*PI;
    float v2= 4.0/3.0*pow(b/2.0/10,3)*PI;
    printf("%.3f %.3f\n",7.86*v1,19.3*v2);
    return 0;
}
```

Note 1.1 (要点).

1. 整数除以整数, 结果为整数。

4.0/3.0 结果是浮点数, 4/3 结果是整数

2. pow 函数原型: `double pow(double x, double y)`

当形参数整数整数时, 由于精度问题, 推荐不使用此函数计算 x^y . 使用循环语句, 易计算 x^y . 如果必要, 可自定义函数: `int mypow(int x, int y)`。见课件。

1.2 温度转化

已知华氏温度到摄氏温度的转换公式为: 摄氏温度 = (华氏温度 - 32) × 5/9, 写程序将给定的华氏温度转换为摄氏温度输出。

输入说明:

只有一个整数, 表示输入的华氏温度

输出说明:

输出一个表示摄氏温度的实数, 小数点后保留 2 位有效数字, 多余部分四舍五入

输入样例:

50

输出样例:

10.00

提示:

用 scanf 输入, 用 printf 输出, 保留 2 位小数的格式控制字符为

```
#include <stdio.h>

int main()
{
    int f;
    float c;
    scanf("%d",&f);
    c = (f-32)*5.0/9;    // (1)
    //c = (f-32)*5/9;    // (2)
    printf("%.2f\n",c);
    return 0;
}
```

Note 1.2 (思考). 为何语句 (1),(2) 计算结果不一致, 哪一条语句正确?

1.3 整数简单运算

编写程序, 计算用户输入的两个整数的和、差、乘积 (*) 和商 (/)。

输入格式: 输入两个整数, 整数之间用空格分隔。

输出格式：输出四个整数结果，分别表示和、差、积和商，每输出一个结果换行。

输入样例：

3 4

输出样例：

7

-1

12

0

```
#include<stdio.h>
int main()
{
    int a,b;
    scanf("%d%d",&a,&b);
    printf("%d\n%d\n%d\n%d\n",a+b,a-b,a*b,a/b);
    return 0;
}
```

Note 1.3 (思考). b=0 时如何处理？

1.4 A+B+C

通过键盘输入三个整数 a, b, c, 求 3 个整数之和。

输入说明：

三整形数据通过键盘输入，输入的数据介于-100000 和 100000 之间，整数之间以空格、跳格或换行分隔。

输出说明：

输出 3 个数的和。

输入样例：

-6 0 39

输出样例：

33

```
#include<stdio.h>
int main()
{
    int a,b,c;
    scanf("%d%d%d",&a,&b,&c);
    printf("%d\n",a+b+c);
    return 0;
}
```

1.5 字符输入输出

通过键盘输入 5 个大写字母，输出其对应的小写字母，并在末尾加上 “!”。

输入说明：

5 个大写字母通过键盘输入，字母之间以竖线 “|” 分隔。

输出说明：

输出 5 个大写字母对应的小写字母，之间无分隔，并在末尾加上 “!”。

输入样例：

H|E|L|L|O

输出样例：

hello!

```
#include <stdio.h>
int main()
{
    char c1, c2, c3, c4, c5;
    scanf("%c| %c| %c| %c| %c", &c1, &c2, &c3, &c4, &c5);
    c1 += 32; c2 += 32; c3 += 32; c4 += 32; c5 += 32;
    printf("%c%c%c%c%c!", c1, c2, c3, c4, c5);
    return 0;
}
```

Note 1.4 (要点). scanf("原样输入",...);

Note 1.5. (大小写字符转化关系) 小写字符 ASCII 码 = 大写字符 ASCII 码 + 32

1.6 数字字符

通过键盘输入 1 个整数 $a(0 \leq a \leq 4)$, 1 个数字字符 $b('0' \leq b \leq '5')$ 求 $a+b$ 。

输入说明：

整形数据、数字字符通过键盘输入，输入的整形数据介于 0 和 4 之间，输入的数字字符介于 ‘0’ 和 ‘5’ 之间，二个输入数之间用 “,” 分隔。

输出说明：

分别以整数形式及字符形式输出 $a+b$ ，输出的二个数之间用 “,” 分隔。

输入样例：

3,5

输出样例：

56,8

```
#include <stdio.h>
int main()
{
```

```
int a;
char b;
scanf("%d,%c",&a,&b);
printf("%d,%c",a+b,a+b);
return 0;
}
```

Note 1.6. (scanf 函数) scanf("原样输入",...);

1.7 实数运算

通过键盘输入长方体的长、宽、高，求长方体的体积 V(单精度)。

输入说明：

十进制形式输入长、宽、高，输入数据间用空格分隔。

输出说明：

单精度形式输出长方体体积 V，保留小数点后 3 位，左对齐。

输入样例：

15 8.12 6.66

输出样例：

811.188

```
#include <stdio.h>
int main()
{
    float a,b,c;
    scanf("%f%f%f",&a,&b,&c);
    printf("%.3f",a*b*c);
    return 0;
}
```

1.8 四则运算

输入两个整数和一个四则运算符，根据运算符计算并输出其运算结果（和、差、积、商、余之一）。注意做整除及求余运算时，除数不能为零。

输入说明：

使用 scanf() 函数输入两个整数和一个运算符，格式见输入样例。

输出说明：

输出使用 printf() 函数，格式见输出样例。

输入样例：

5%2

输出样例:

5%2=1

```
#include <stdio.h>
int main()
{
    int a,b;
    char op;
    scanf("%d%c%d",&a,&op,&b);
    switch(op)
    {
        case '+': printf("%d%c%d=%d\n",a,op,b,a+b); break;
        case '-': printf("%d%c%d=%d\n",a,op,b,a-b); break;
        case '*': printf("%d%c%d=%d\n",a,op,b,a*b); break;
        case '/': printf("%d%c%d=%d\n",a,op,b,a/b); break;
        case '%': printf("%d%c%d=%d\n",a,op,b,a%b); break;
    }
    return 0;
}
```