第 2 章 用 C 语言编写程序

2.1 教学要点

本章从案例出发，概要介绍顺序、分支和循环 3 种基本控制结构和函数的使用，以及在案例程序中用到的语言知识，使学生对 C 语言有一个总体的了解，并能从模仿入手，学习编写简单的程序，培养学习兴趣。

2.1 节通过案例“在屏幕上显示一个短句“Hello World!”及相关知识的学习，使学生能编程实现在屏幕上显示若干信息。教师在讲解程序的过程中，介绍涉及到的主函数 main()、语句、注释和编译预处理等语言知识。本小节应与第 1 章一起在第 1 次课讲授，使学生能尽快上机练习。

2.2 节通过案例“求华氏温度 100°F 对应的摄氏温度”及相关知识的学习，使学生能编程进行简单的数据处理，并掌握涉及到的语言知识，包括常量、变量、数据类型、算术运算和赋值运算，以及格式化输出函数 printf()。

2.3 节通过案例“根据居民的月用水量分段计算水费”及相关知识的学习，使学生能编程计算二分段函数，并掌握涉及到的语言知识，包括关系运算、if-else 语句、格式化输入函数 scanf()，以及常用的数学库函数。

2.4 节通过案例“输出华氏－摄氏温度转换表”及相关知识的学习，使学生能使用 for 循环语句实现指定次数的循环程序设计。

2.5 节通过案例“生成乘方表与阶乘表”及相关知识的学习，使学生对函数的定义和调用有初步的认识，能模仿编程。

作为学生课程学习的起始点，本章的学习效果至关重要。教师讲授本章时，应处理好学生程序设计能力培养与语言知识学习的关系，以程序设计为主线开展教学，只介绍在示例程序中用到的语言知识，不要展开，避免学生面对庞杂的语言体系不知所措，产生畏难情绪。在学生初学编程的过程中，应多鼓励，多赞扬，不忽视学生的点滴进步，不吝啬赞美之词，使学生切实感受到每节课都有进步、有收获，在上机编程中享受学习的乐趣和成就感。

讲授学时：8 学时，实验学时同讲授学时。

本章的知识能力结构图见图 2.1。

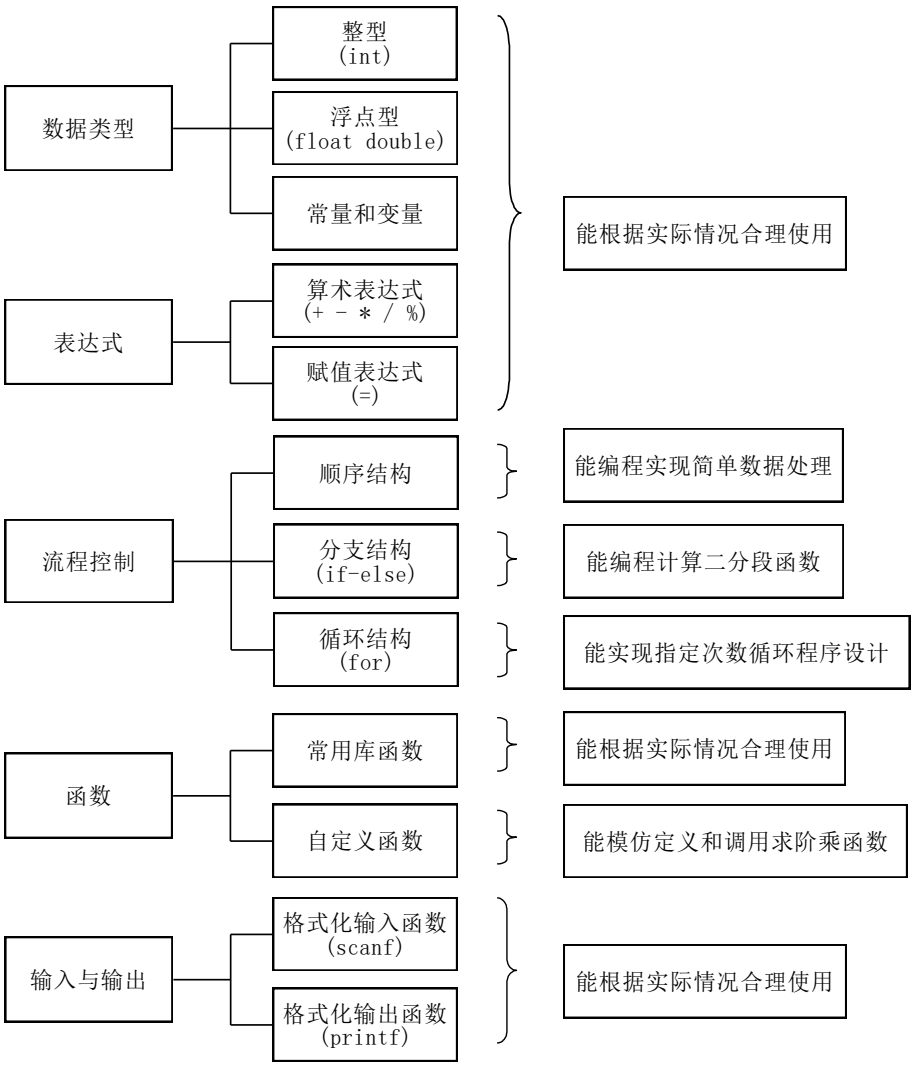


图 2.1 知识能力结构图

2.2 讲稿

1	**Chap 2　用C语言编写程序** **2.1　在屏幕上显示 Hello World!** **2.2　求华氏温度 100°F 对应的摄氏温度** **2.3　计算分段函数** **2.4　输出华氏—摄氏温度转换表** **2.5　生成乘方表与阶乘表**	本章分 5 节。

2	**本章要点** ■ 怎样编写程序，在屏幕上显示一些信息？ ■ 怎样编写程序，实现简单的数据处理，例如将华氏温度转换为摄氏温度? ■ 怎样使用 **if** 语句计算分段函数? ■ 怎样用 **for** 语句求 **1+2+…+100?** ■ 如何定义和调用函数生成一张乘方表?	提出本章的学习要点。
3	**2.1 在屏幕上显示Hello World!** **例2-1 在屏幕上显示一个短句:** **Hello World!**	提出问题：如何在屏幕上显示一个短句？

4	**在屏幕上显示Hello World!**
	/* 显示 “Hello World!” */ ← 注释文本
	# include <stdio.h>
	int main (void) ← 主函数
	{
	printf ("Hello World! \n"); ← 语句结束
	return 0;
	}
	输出函数 换行符
	1. 任何程序都有主函数
	2. 程序由若干语句组成
	3. 语句由；结束
		先展示例 2-1 程序，然后运行程序，查看结果。
		按照 PPT 动画的顺序依次介绍 main 函数、输出函数、分号、换行符和注释文本。
		设问：如何在屏幕上显示 How are you?
		解答：
		将 printf(“Hello World!\n”);
		改为：
		printf(“How are you?\n”);
		修改程序后，运行程序。

5	**在屏幕上显示一些信息**
	例2-2 在屏幕上显示:
	Programming is fun!
	And Programming in C is even more fun!
	# include <stdio.h> ← 编译预处理命令
	int main (void)
	{
	printf ("Programming is fun! \n");
	printf ("And Programming in C is even more fun! \n");
	return 0;
	}
		提出问题：如何在屏幕上显示 2 句话？
		展示、运行例 2-2 程序。
		设问 1：可以用一条语句完成吗？
		解答：
		`printf("Programming is fun.\nAnd Programming in C is even more fun!\n");`
		设问 2：如何打印一个简单图案？
		请学生提出要打印的图案，编程实现。

6	**2.2 求华氏温度 100° F 对应的摄氏温度** 摄氏温度 c = (5/9)(f-32) 2.2.1 程序解析 2.2.2 常量、变量和数据类型 2.2.3 算术运算和赋值运算 2.2.4 格式化输出函数 printf()	提出问题：怎样根据计算公式实现温度转换？ 本节介绍为解决这个问题所编写的程序和涉及到的语言知识。

7	**2.2.1 程序解析**
	例2-3 求华氏温度 100° F 对应的摄氏温度。
	摄氏温度 c=5*(f-32)/9
	#include <stdio.h>
	int main (void)
	{
	int celsius, fahr; 变量定义
	fahr = 100;
	celsius = 5 * (fahr - 32) / 9; } 变量使用
	printf ("fahr = %d, celsius = %d\n", fahr, celsius);输出结果
	return 0;
	}
	输出:fahr =100, celsius = 37
		展示、运行例 2-3 程序。
		解读程序的过程中顺带提一下定义了 2 个变量存放数据，可以详细解释输出结果，并与 2.1 节对比。
		指出：不同之处在于 printf 中新出现的“%d”和变量（紫色字体）。

8	**2.2.2 常量、变量和数据类型** int celsius, fahr; celsius = 5 * (fahr - 32) / 9; ■ 数据 □ 常量：在程序运行过程中，其值不能被改变 □ 变量：在程序运行过程中，其值可以被改变 ■ 数据类型 □ 常量：5 和 9 是整型常量（整数） □ 变量：在定义时指定	由例 2-3 程序引出常量、变量和数据类型的概念。
9	**变量的定义** 变量名：小写字母；见名知义 变量定义的一般形式： 类型名 变量名表； 例如： int celsius, fahr; 定义整型变量 float x; 定义单精度浮点型变量 double area, length; 定义双精度浮点型变量 double型数据比float精度高，取值范围大	说明变量定义的一般形式。 提示 1：变量名应“见名知义”。 提示 2：变量名中的英文字母习惯用小写字母。

10	**变量的定义** ■ 定义变量时要指定变量名和数据类型 类型名　　变量名表； int celsius, fahr; float x; double area, length; ■ 变量名代表内存中的一个存储单元 存放该变量的值 ■ 该存储单元的大小由变量的数据类型决定 ■ C语言中的变量代表保存数据的存储单元 ■ 数学中的变量代表未知数 x = x+1	结合存储单元解释变量定义的作用，为什么定义变量时要定名字和类型？ 比较C语言中的变量与数学中的变量的不同含义。

11	**变量的定义与使用**
	变量必须先定义，后使用。
	应该先赋值，后引用
	#include <stdio.h>
	int main(void)
	{
	int celsius, fahr;
	fahr = 100;
	celsius = 5 * (fahr - 32) / 9;
	printf("fahr = %d, celsius = %d\n", fahr, celsius);
	return 0;
	}
	• 一个变量名只能定义一次
	• 变量定义一般在程序首部
		定义变量的目的是为了使用变量。
		两条原则：
		（1）先定义，后使用；
		（2）先赋值，后引用。

12	**2.2.3　算术运算和赋值运算**
	fahr = 100;
	celsius = 5 * (fahr - 32) / 9;
	1. 算术运算
	□双目算术运算符：+ - * / %
	□算术表达式：用算术运算符将运算对象连接起来的符合C语言语法规则的式子
	数学式：5(f-32) /9
	C表达式：5＊(fahr - 32) / 9 或者：
	数学式：s(s-a)(s-b)(s-c)
	C表达式：
		由例 2-3 程序引出算术运算和赋值运算的概念。
		关于算术运算：
		（1）只介绍双目算术运算符；
		（2）C 表达式：用运算符将运算对象连接起来。
		讲解并现场练习将数学式正确表示为C表达式，将例 2-3 中的温度转换公式表示为合法的 C 表达式（不限一种）。

13	**算术运算** □双目算术运算符：+ - * / % □算术表达式：用算术运算符将运算对象连接起来的符合C语言语法规则的式子 注意 ■ / 整数除整数，得整数 1/2 = 0，9/4 = 2 5＊(fahr - 32) / 9 和 5 / 9 ＊(fahr - 32) 等价吗? ■ % 针对整型数据 5%6=5，9%4=1，100%4=0 ■ 双目运算符两侧操作数的类型要相同	对除法（/）和求余（%）运算的说明。 针对前面写出的对应温度转换公式的所有 C 表达式，讨论哪个表达式是适合的？为什么？

14	**赋值运算**
	■ 赋值运算符 =
	■ 赋值表达式：用 = 将一个变量和一个表达式连接起来的式子
	变量 = 表达式　　=的左边必须是一个变量
	例如：
	fahr = 100;
	celsius = 5 * (fahr - 32) / 9;
	□计算赋值运算符右侧表达式的值
	□将赋值运算符右侧表达式的值赋给左侧的变量
		关于赋值运算：
		（1）明确基本运算过程；
		（2）提醒赋值号的左边必须是变量。

15	**2.2.4　格式化输出函数printf()** 数据输出：格式化输出函数 printf() #include <stdio.h> printf ("Hello World! \n"); printf ("fahr = %d, celsius = %d\n", fahr, celsius); printf (格式控制字符串, 输出参数1, …, 输出参数n); 要输出的数据 用双引号括起来，表示输出的格式	由本章的前 3 个例子引出 printf()函数，说明该函数的一般调用格式，即格式控制字符串和输出参数。

16	**printf—格式控制字符串**
	printf ("Hello World! \n");
	printf ("fahr = %d, celsius = %d\n", fahr, celsius);
	printf ("Hi\n", fahr);
	格式控制字符串：
	□普通字符：原样输出
	□格式控制说明：按指定的格式输出数据，%…
	与数据类型有关
	■ int型 ：%d
	■ float double型：%f
	printf ("fahr = %d, celsius = %d\n", fahr, celsius);
	输出:fahr =100, celsius = 37
		重点解释格式控制字符串中的普通字符和格式控制说明的作用，说明输出参数的类型决定选用的格式控制符(只介绍%d和%f)。
		设问：若 a=3，b=5，求 s=a+b 的值，并按形如“3+5=8”或“8=3+5”的形式输出。
		解答：
		a=3; b=5;
		s=a+b;
		printf(“%d+%d=%d\n”, a, b, s);
		或 printf(“%d=%d+%d\n”, s, a, b);

17	**2.3 计算分段函数** 2.3.1　程序解析 2.3.2　关系运算 2.3.3　if-else语句 2.3.4　格式化输入函数scanf 2.3.5　常用数学库函数	提出问题：怎样计算分段函数？ 本节介绍为解决这个问题所编写的程序和涉及到的语言知识。

18	**2.3.1 程序解析** 例**2-4** 分段计算水费 输入用户的月用水量 **x**（吨），计算并输出该用户应支付的水费 **y**（元）（保留**2**位小数） $y = f(x) = \begin{cases} \frac{4x}{3} & x \le 15 \\ 2.5x - 10.5 & x > 15 \end{cases}$ 要解决的问题： □输入 □计算分段函数 □输出，并保留2位小数	例 2-4 要求：按月用水量分段计算水费。 问题分析： 有 3 个用已有的知识无法解决的新问题： （1）输入 x； （2）计算分段函数； （3）输出（可以实现），保留 2 位小数（新问题）。

19	**2.3.1 程序解析—求分段函数**
	`#include <stdio.h>`
	`int main (void)`
	`{`
	`double x, y;`
	`printf ("Enter x (x>=0):\n");` /* 输入提示 */
	`scanf ("%lf", &x);` /* 调用scanf函数输入数据 */
	`if(x <= 15){` /* if – else语句 */
	`y = 4 * x / 3;`
	`}`
	`else{`
	`y = 2.5 * x - 10.5;`
	`}`
	`printf ("f(%f) = %.2f\n", x, y);`
	`return 0;`
	`} }`
	Enter x (x>=0):
	9.5
	f(9.500000)=12.67
	Enter x (x>=0):
	15
	f(15.000000)=20.00
	数据必须输入吗？
	Enter x (x>=0):
	21.3
	f(21.300000)=42.75
		展示例 2-4 程序，并连续运行程序 3 次。
		解读程序的过程中回答上页的 3 个问题（紫色字体），不需展开。
		设问：为什么程序运行 3 次？
		解答：计算水费的分段函数中，x 有 3 个取值区间：小于 15、15、大于 15，故需运行 3 次，每个区间取一个值。
		再次连续运行程序 3 次，由学生定输入数据。
		提示：强烈推荐养成“{语句}”的好习惯，即使只有一条语句。如本例中 if-else 语句。在 Dev-C 环境下，这样写能自动调整对齐为多层缩进的格式，使程序层次分明。

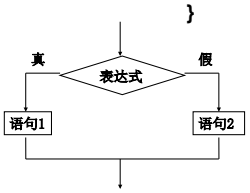
20	**2.3.2 关系运算** **x <= 15** 比较 **x** 和 **15** 的大小 比较的结果：真 假 当 **x** 取值**9.5**时， **x <= 15**的结果是：？ 当 **x** 取值**21.3**时，**x <= 15**的结果是：？ 关系运算 — 比较运算，比较两个操作数 ■ 关系运算符 **x < y x <= y x == y** **x > y x >= y x != y** 区分**=** 和**==** ■ 关系表达式：用关系运算符将2个表达式连接起来的式子。 如：**x <= 1**	由例 2-4 程序引出关系运算的概念。 说明 3 点： （1）关系运算就是比较运算； （2）关系运算的结果是“真”或“假”，没有第三种情况。建议不提 C 语言中分别用 1 和 0 代表“真”和“假”（第 6 章讲）。 （3）顺带提醒学生注意“=”和“==”的区别，前者是赋值运算符，后者是关系运算符。
21	**运用关系表达式** 表示比较的数学式 **C**关系表达式 **x≤10** x <= 10 **x≥10** x >= 10 **x≠10** x != 10 **x = 10** x == 10 用关系表达式描述条件 □判断 **x** 是否为负数 **x < 0** □判断 **x** 是否不为零 **x != 0**	学习关系表达式的目的是合理使用之。 讲解并现场练习将数学式表示为C关系表达式，以及用关系表达式描述条件。

<table>
<tr><td>22</td><td>

2.3.3 if - else语句

```
if (表达式)
    语句1
else
    语句2
```

```
if(x <= 15) {
    y = 4 * x / 3;
}
else{
    y = 2.5 * x - 10.5;
}
```



</td><td>

由例 2-4 程序引出 if-else 语句。

说明 3 点：

（1）使用 if-else 语句求分段函数；

（2）if-else 语句的一般形式；

（3）if-else 语句的执行流程。

</td></tr>
<tr><td>23</td><td>

计算二分段函数

$$f(x)=\begin{cases}\frac{1}{x} & x \neq 0\\ 0 & x=0\end{cases}$$

```
if (表达式)
    语句1
else
    语句2
```

```
if(x != 0){
    y = 1/x;
}
else{
    y = 0;
}
```

</td><td>

练习用 if-else 语句求分段函数。

</td></tr>
<tr><td>24</td><td>

源程序

```
#include <stdio.h>
int main (void)
{
    double x, y;

    printf ("Enter x:\n");
    scanf ("%lf", &x);
    if(x != 0){
        y = 1/x;
    }
    else{
        y = 0;
    }
    printf ("f(%.2f) = %.1f\n", x, y);
    return 0;
}
```

```
#include <stdio.h> /* 例2-4 */
int main (void)
{
    double x, y;

    printf ("Enter x (x>=0):\n");
    scanf ("%lf", &x);
    if(x <= 15){
        y = 4 * x / 3;
    }
    else{
        y = 2.5 * x - 10.5;
    }
    printf ("f(%f) = %.2f\n", x, y);
    return 0;
}
```

</td><td>

展示完整的源程序。

</td></tr>
<tr><td>25</td><td>

```
#include <stdio.h>
int main (void)
{
    double x, y;

    printf ("Enter x:\n");
    scanf ("%lf", &x);
    if(x != 0){
        y = 1/x;
    }
    else{
        y = 0;
    }
    printf ("f(%.2f) = %.1f\n", x, y);
    return 0;
}
```

运行结果

```
input x:
2.5
f(2.50)=0.4
```

```
input x:
0
f(0.00)=0.0
```

</td><td>

单步运行程序 2 次，观察语句执行顺序。

设问：为什么程序运行 2 次？

解答：求解的分段函数中，x 有 2 个取值区间：0 和非 0，故需运行 2 次，每个区间取一个值。

再次连续运行程序 2 次，由学生定输入数据。

</td></tr>
</table>

26	**软件测试的基本思想**
	if(x != 0)
	y = 1/x;
	else
	y = 0;
	input x:
	2.5
	f(2.50)=0.4
	input x:
	0
	f(0.00)=0.0
	软件测试
	精心设计一批测试用例［输入数据，预期输出结果］，然后分别用这些测试用例运行程序，看程序的实际运行结果与预期输出结果是否一致。
	if(x <= 15)
	y = 4 * x / 3;
	else
	y = 2.5 * x - 10.5;
	Enter x (x>=0):
	9.5
	f(9.500000)=12.67
	Enter x (x>=0):
	15
	f(15.000000)=20.00
	Enter x (x>=0):
	21.3
	f(21.300000)=42.75
		介绍软件测试的基本思想，说明 4 点：
		（1）测试：用测试用例运行程序，若实际运行结果与预期结果不一致，说明程序中存在错误；
		（2）调试：查找并改正程序中的错误；
		（3）测试和调试需要反复进行；
		（4）穷举测试用例不可行。可根据程序的逻辑结构和功能，设计有代表性的测试用例。
		教师可以再列举几个分段函数，让学生练习写出相应的 if-else 语句，以及测试用例。

27	**2.3.4 格式化输入函数scanf()** 数据输入：格式化输入函数 scanf() #include <stdio.h> scanf ("%lf", &x); scanf (格式控制字符串, 输入参数1, …, 输入参数n); 变量地址 用双引号括起来，表示输入的格式	由例 2-4 程序引出 scanf()函数，说明该函数的一般调用格式，即格式控制字符串和输入参数，讲解时可与 printf()对照。 提醒学生注意输入参数的形式，在变量名前面加&，如&x。
28	**scanf—格式控制字符串** 格式控制字符串： □格式控制说明：按指定的格式输入数据，%… 与数据类型有关 ▪ int型 ：%d ▪ float型：%f ▪ double型：%lf □普通字符：原样输入 尽量不要出现普通字符 例如： scanf ("%lf", &x); scanf ("x=%lf", &x); 输入：9.5 输入:x=9.5	重点解释格式控制字符串中的普通字符和格式控制说明的作用，说明 3 点： （1）输入数据的类型决定选用的格式控制符（介绍%d、%f 和%lf），提醒 double 型数据用%lf。 （2）普通字符原样输入（与 printf()对比），举例说明。 （3）需要输入数据时，请注意 scanf()函数的输入参数和格式控制说明、以及程序运行时实际输入的数据这三者之间的协调。

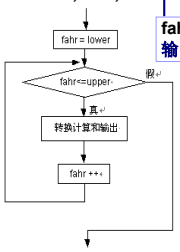
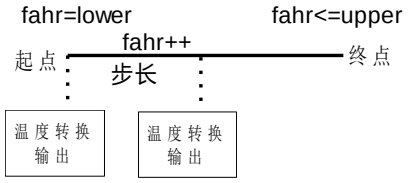
29	**改进例2-3的程序**
	例2-3 求华氏温度 100° F 对应的摄氏温度。
	摄氏温度 C=5*(F-32)/9
	#include <stdio.h>
	int main (void)
	{
	int celsius, fahr;
	printf ("Enter fahr: \n");
	fahr = 100;
	scanf ("%d", &fahr);
	celsius = 5 * (fahr - 32) / 9;
	printf ("fahr = %d, celsius = %d\n", fahr, celsius);
	return 0;
	}
	Enter fahr:
	100
	fahr =100, celsius = 37
		回顾例 2-3，将变量赋值改为输入。以避免由于数据的改变而修改程序的情况。

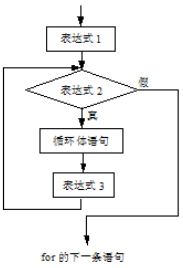
30	**2.3.5 常用数学库函数** ■ 库函数 □C语言处理系统提供事先编好的函数，供用户在编程时调用。scanf(), printf(), exp() □在相应的系统文件（头文件）中定义一些必需的信息。 ■ #include命令 □用户调用库函数时，将相应的头文件包含到源程序中。 例如 □调用scanf, printf，需要 #include <stdio.h> □调用sqrt，需要 #include <math.h>	讲解库函数的使用，说明 2 点： （1）编程时可根据需要调用 C 库函数； （2）调用时，必须包含相应的头文件。
31	**常用数学库函数** ■ 平方根函数 sqrt (x) ■ 绝对值函数 fabs (x) fabs(-3.56) 的值为3.56 ■ 幂函数 pow (x, n)： x^n pow(1.1, 2) 的值为1.21（即1.12） ■ 指数函数 exp (x)： e^x exp(2.3) 的值为$e^{2.3}$ ■ 以e为底的对数函数 log (x)： ln x log(123.45) 的值为4.815836 ■ 以10为底的对数函数 log10 (x)： $\log_{10}x$ log10(123.45) 的值为2.091491。	使学生了解常用的数学库函数，以备需要时调用。
32	**例2-5 坚持的力量** 以第一天的能力值为基数，用initial表示，能力值相比前一天提高的值factor就是努力参数，坚持天数为day，让我们一起来看看坚持的力量。输入能力的初始值 initial、努力参数factor和坚持天数day，根据下列公式计算出坚持努力后达到的能力值，输出时保留两位小数。 $result = initial(1 + factor)^{day}$	由例 2-5 引出库函数的调用，以及多个数据的输入。

33	**例2-5 程序**
	Enter initial:1.0
	Enter factor:0.01
	Enter day:365
	result = 37.78
	`# include <stdio.h>`
	`# include <math.h>`
	`int main(void)`
	`{`
	`int day;`
	`double factor, initial, result;`
	`printf("Enter initial:");`
	`scanf("%lf", &initial);  /* 调用scanf()函数输入initial */`
	`printf("Enter factor:");`
	`scanf("%lf", &factor);  /* 调用scanf()函数输入factor */`
	`printf("Enter day:");`
	`scanf("%d", &day);  /* 调用scanf()函数输入day */`
	`result = initial * pow(1 + factor, day);  /* 调用幂函数pow() */`
	`printf("result = %.2f\n", result);`
	`return 0;`
	`}`
		展示、运行例 2-5 程序。
		
		调用 3 次 scanf 函数输入 3 个数据。
		设问：可以只调用一次 scanf 函数吗？

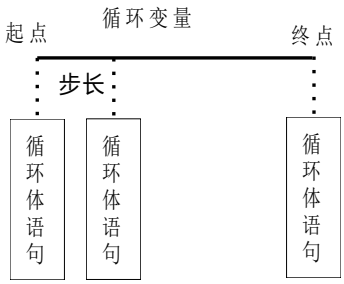
34	**调用scanf函数输入多个数据** scanf("%lf%lf%d", &initial, &factor, &day); 输入：1.0 0.01 365 ■ scanf需要多个输入参数和多个格式控制说明 输入参数的类型、个数和位置要与格式控制说明一一对应 ? scanf("%lf%d%lf", &initial, &factor, &day); ■ 程序运行时，输入的多个数据之间必须有间隔。 数值型数据之间的间隔可以是空格或换行	由例 2-5 的设问引出调用 scanf()函数输入多个数据。 编程解决实际问题时，一般都需要输入数据，此时应注意程序中 scanf()函数的输入参数、scanf()函数的格式控制说明、程序运行时实际输入的数据这三者之间的协调。
35	**2.4 输出华氏—摄氏温度转换表** 2.4.1 程序解析 2.4.2 for语句 2.4.3 指定次数的循环程序设计	提出问题：2.2 节将华氏温度 100°F 转换为相应的摄氏温度，如何将华氏温度 30~35°F 都转换为相应的摄氏温度，即如何输出一张温度转换表？ 本节介绍为解决这个问题所编写的程序和涉及到的语言知识。
36	**2.4.1 程序解析** 例2-6输入2个整数lower和 upper，输出一张华氏—摄氏温度转换表，华氏温度的取值范围是[lower, upper]，每次增加1°F。 fahr celsius 30 -1.1 31 -0.6 32 0.0 33 0.6 34 1.1 35 1.7	问题的具体描述。

37	**2.4.1 程序解析-温度转换表**
	#include <stdio.h>
	int main (void)
	{ int fahr, lower, upper;
	double celsius;
	printf ("Enter lower:"); scanf ("%d", &lower);
	printf ("Enter upper:"); scanf ("%d", &upper);
	if (lower <= upper){ /* 判断输入数据的合法性 */
	printf ("fahr celsius\n");
	//温度重复转换，华氏温度从lower开始，到upper结束，每次增加1°F
	for (fahr = lower ; fahr<= upper; fahr ++){
	celsius = (5.0 / 9.0) * (fahr - 32);
	printf ("%4d%6.1f\n", fahr, celsius);
	}
	}else
	printf ("Invalid Value!\n");
	return 0;
	}
	fahr = fahr+1
	Enter lower: 30
	Enter upper: 35
	fahr celsius
	30 -1.1
	31 -0.6
	32 0.0
	33 0.6
	34 1.1
	35 1.7
		展示、运行例 2-6 程序。
		华氏温度 fahr 取值范围为[low, upper]时，对 fahr 的每个值，重复以下 2 个操作：
		(1)温度转换：celsius = (5.0/9.0) * (fahr-32);
		(2)输出：printf("%d%6.1f\n", fahr, celsius);
		这就是循环，用 for 语句实现（紫色字体）。顺带说明 fahr++等价于 fahr = fahr + 1。

38	**例2-6中for语句的流程**
	for (fahr = lower; fahr <= upper; fahr ++) {
	celsius = (5.0 / 9.0) * (fahr - 32)
	printf("%d %6.1f\n", fahr, celsius);
	}
	fahr = fahr+2
	输出？
	Enter lower: 30
	Enter upper: 35
	fahr celsius
	30 -1.1
	31 -0.6
	32 0.0
	33 0.6
	34 1.1
	35 1.7
	 
		介绍例 2-6 中 for 语句的执行流程，单步运行，观察语句被反复执行的过程。
		 
		设问：将 fahr++改为 fahr = fahr + 2，程序的功能有什么变化？
		解答：修改并运行程序，让学生观察运行结果。fahr = fahr + 2 意味华氏温度 fahr 在取值范围内，每次增加 2°F，而不是 1°F。

39	**2.4.2 for语句—循环语句** for (表达式1；表达式2；表达式3) 循环体语句 实现C语句的重复执行 3个表达式、循环体语句 ! 书写顺序和执行顺序不同 ! 表达式1只执行一次 	讲解 for 语句的一般形式与执行流程。 提醒：计算表达式 3 在执行循环体语句之后。

40	**for语句中的循环变量**
	循环(控制)变量：for语句中，通过改变或判断某个变量的值来控制循环的执行
	for (fahr = lower; fahr <= upper; fahr ++) {
	celsius = (5.0 / 9.0) * (fahr - 32.0);
	printf("%d %6.1f\n", fahr, celsius);
	}
	赋初值 判断其值 改变其值
		引出循环变量的概念。

41	**for语句的说明**
	for (fahr = lower; fahr <= upper; fahr ++) {
	celsius = (5.0 / 9.0) * (fahr - 32.0);
	printf("%d %6.1f\n", fahr, celsius);
	}
	表达式1：给循环变量赋初值，指定循环的起点。
	fahr = lower
	表达式2：给出循环的条件，判断循环是否达到终点？
	fahr <= upper
	表达式3：设置循环的步长，改变循环变量的值，从而可改变表达式2的真假性。
	fahr++
	循环体语句：被反复执行的语句，一条语句。
		讨论 for 语句中 3 个表达式和循环体语句的含义与功能。
		 

42	**复合语句{ }和空语句 ；**
	for (fahr = lower; fahr <= upper; fahr ++) {
	celsius = (5.0 / 9.0) * (fahr - 32.0);
	printf("%d %6.1f\n", fahr, celsius);
	}
	for (fahr = lower ; fahr <= upper; fahr = fahr + 1)
	celsius = (5.0 / 9.0) * (fahr - 32.0);
	printf("%d %6.1f\n", fahr, celsius);
	for (fahr = lower ; fahr <= upper; fahr = fahr + 1);
	celsius = (5.0 / 9.0) * (fahr - 32.0);
	printf("%d %6.1f\n", fahr, celsius);
	！不要在for语句中随意加分号
		讨论复合语句和空语句，建议现场修改并运行程序，比较不同的结果。
		设问：若循环条件为 fahr <= upper，循环结束时，fahr 的值是什么？
		解答：循环结束，说明循环条件不满足，此时 fahr > upper，但 fahr 的具体值与 for 语句的其他内容有关。
		可结合单步运行，观察循环结束时 fahr 的值，特别注意表达式 3 分别为 fahr++、fahr = fahr + 2 的不同。

43	**2.4.3 指定次数的循环程序设计**
	求 1+2+……+100
	抽取具有共性的算式：sum = sum + i
	sum初值为0，该算式重复100次，i从1变到100
	设i为循环变量，则：
	指定循环起点的表达式1：i = 1
	给出循环条件的表达式2：i <= 100
	设置循环步长的表达式3：i++
	循环体语句：sum = sum + i;
	for (i = 1; i <= 100; i++){
	sum = sum + i;
	}
		通过以下 8 个例子（求累加和、求连乘），讲解指定次数的循环程序设计。
		（1）求 1+2+……+100
		（2）求 1+1/2+1/3+……+ 1/100
		（3）例 2-7，求 1+2+3+……+n
		（4）求 1+1/2+1/3+……+ 1/n
		（5）求 1+1/3+1/5+… 的前 n 项和
		（6）例 2-8，求 1-1/3+1/5-… 的前 n 项和
		（7）例 2-9，求 n!
		（8）求 x^n

44	**源程序 求 1+2+……+100**
	sum=5050
	/* 计算1 + 2 + 3 + …… + 100 */
	#include <stdio.h>
	int main (void)
	{
	int i, sum;
	sum = 0; /* 置累加和sum的初值为0 */
	for (i = 1; i <= 100; i++){ /* 循环重复100次 */
	sum = sum + i; /* 反复累加 */
	}
	printf ("sum = %d\n", sum); /* 输出累加和 */
	return 0;
	}
	for(i = 1; i <= 100; i++){
	sum = 0;
	sum = sum + i;
	}
		第 1 个例子：求 1+2+……+100。
		单步运行若干次，观察变量 i, sum 的变化，然后运行到最后，查看结果。
		
		提示：再次强烈推荐养成“{语句}”的好习惯，即使只有一条语句。如本例中 for 语句。在 Dev-C 环境下，这样写能自动调整对齐为多层缩进的格式，使程序层次分明。

45	**求 1+1/2+1/3+……+ 1/100**
	#include <stdio.h>
	int main (void)
	{
	int i, sum;
	sum = 0;
	for (i = 1; i <= 100; i++){
	sum = sum + i;
	}
	printf ("sum = %d\n", sum);
	return 0;
	}
	int i;
	double sum;
	for(i = 1; i <= 100; i++){
	sum = sum + 1.0/i;
	}
	printf ("sum = %f\n", sum);
		第 2 个例子：求 1+1/2+1/3+……+ 1/100。
		在第 1 个例子的基础上修改。
		说明 2 点：
		（1）sum 的类型改为 double，故 printf 中的格式控制符随之修改为%f；
		（2）1/i 与 1.0/i 的区别，前者运算结果为整型，后者是浮点型。

<table>
<tr>
<td>46</td>
<td>

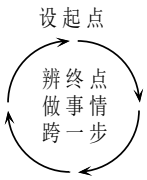
指定次数的循环程序设计

一般包含四个部分：

- 初始化：指定循环起点
  - 给循环变量赋初值，如**i = 1;**
  - 进入循环之前，设置相关变量的初值，如**sum = 0**。
- 条件控制：
  - 只要**i <= 100**，循环就继续
- 工作：指重复执行的语句（循环体）。
  - 一条语句，可以是复合语句或空语句。如**sum = sum + i**。
- 改变循环变量：在每次循环中改变循环变量的值
  - 如 **i++**，以改变循环条件的真假。一旦**i > 100**，循环结束。

</td>
<td>

通过前面 3 个例子，归纳指定次数循环程序设计的 4 个部分，简而言之：



</td>
</tr>
<tr>
<td>47</td>
<td>

例2-7 求 1+2+3+……+n

输入一个正整数n，求前n项和，即循环n次

```
#include <stdio.h>
int main (void)
{
   int i, sum;

   sum = 0;
   for ( i = 1; i <= 100; i++ ){
      sum = sum + i;
   }
   printf ("sum = %d\n", sum);
   return 0;
}
```

```
printf ("Enter n:");
scanf ("%d", &n);
```

```
for (i =1; i <= n; i++){
    sum = sum + i;
}
```

```
Enter n: 100
Sum = 5050
```

</td>
<td>

第 3 个例子：求 1+2+3+……+n（例 2-7）。在第 1 个例子的基础上修改。

</td>
</tr>
<tr>
<td>48</td>
<td>

求1+1/2+1/3+……+ 1/n

```
#include <stdio.h>
int main (void)
{
   int i, sum;

   printf ("Enter n: ");
   scanf ("%d", &n);
   sum = 0;
   for ( i = 1; i <= n; i++ ){
      sum = sum + i;
   }
  printf ("sum = %d\n", sum);
  return 0;
}
```

```
#include <stdio.h>
int main (void)
{ int i;
   double sum;

   printf ("Enter n: ");
   scanf ("%d", &n);
   sum = 0;
   for (i = 1; i <= n; i++){
      sum = sum + 1.0/i;
   }
   printf ("sum = %f\n", sum);
   return 0;
}
```

</td>
<td>

第 4 个例子：求 1+1/2+1/3+……+ 1/n。在第 3 个例子的基础上修改。

类似于第 2 个例子在第 1 个例子的基础上修改。

</td>
</tr>
<tr>
<td>49</td>
<td>

求 1+1/3+1/5+… 的前n项和

求前n项和，即循环n次，每次累加1项。

```
for (i = 1; i <= n ; i++){
    sum = sum + item (第i项)
}

item = 1.0 / (2 * i - 1)
```

</td>
<td>

第 5 个例子：求 1+1/3+1/5+… 的前 n 项和。引入循环不变式的概念。

循环不变式：表示一种在循环过程进行时不变的性质，不依赖于前面所执行过程的重复次数的断言。可直观理解为具有共性的式子。例 2-6 中循环不变式为：

{celsius = (5.0 / 9.0) * (fahr - 32);

printf("%d %6.1f\n", fahr, celsius);}

而求累加和，循环不变式一般为：

sum = sum + item

</td>
</tr>
</table>

50	**源程序 求 1+1/3+1/5+…**
	#include <stdio.h>
	int main (void)
	{ int i, n;
	double item, sum;
	
	printf ("Enter n: ");
	scanf ("%d", &n);
	sum = 0 ;
	for (i = 1; i <= n; i++) {
	item = 1.0 / (2 * i - 1); /* 计算第i项的值 */
	sum = sum + item ; /* 累加第i项的值 */
	}
	printf ("sum = %f\n", sum);
	return 0;
	}
		展示、运行程序。
		问题分析（逐步细化）：
		求累加和的循环不变式为：
		sum = sum + item
		根据具体问题（1+1/3+1/5+…）分析 item 的构成：
		item = 1.0 / (2 * i - 1)

51	**例2-8 求 1-1/3+1/5-… 的前n项和** 求前n项和，即循环n次，每次累加1项。 for (i = 1; i <= n ; i++) sum = sum + item (第i项) item = flag*1.0/(2 * i - 1) item = flag * 1.0 / denominator denominator = denominator + 2 flag = -flag	第 6 个例子：求 1-1/3+1/5-… 的前 n 项和（例 2-8）。 问题分析（逐步细化）： sum = sum + item item = flag * 1.0 / denominator denominator = denominator + 2 flag = -flag

52	**例2-8 源程序**
	#include <stdio.h>
	int main(void)
	{ int denominator, flag, i, n;
	double item, sum;
	
	printf("Enter n: ");
	scanf ("%d", &n);
	flag=1;
	denominator =1;
	item=1;
	sum = 0 ;
	for (i = 1; i <= n; i++) {
	sum = sum + item ; /* 累加第i项的值 */
	flag = -flag; /*改变符号，准备下一次循环 */
	denominator = denominator +2; /* 分母递增2 */
	item = flag * 1.0/ denominator; /* 计算item的值 */
	}
	printf ("sum = %f\n", sum);
	}
		展示、运行程序。
		说明 2 点：
		（1）for 循环体中语句的顺序与问题分解的顺序并不一定一致；
		（2）在 for 语句前对变量赋的初值与 for 循环体中语句的顺序有关。

53	**例2-9 求n!** n! = 1*2*…*n product = ? for (i = 1; i <= n ; i++) product = product * item (第i项) item=i	第 7 个例子：求 n! （例 2-9）。 问题分析（逐步细化）： 求连乘的循环不变式一般为： product = product * item 根据具体问题（求 n!）分析 item 的构成： item = i

<table>
<tr><td>54</td><td>

例2-9 源程序

```
#include <stdio.h>
int main (void)
{
   int i, n;
   double product;

   printf ("input n: \n");
   scanf ("%d", &n) ;
   product = 1;              /* 置阶乘product的初值为1 */
   for (i = 1; i <= n; i++){   /* 循环重复n次，计算n! */
      product = product * i ;
   }
   printf ("product = %.0f\n", product);
   return 0;
}
```

</td><td>

展示、运行程序。

设问：product 的初值为何赋 1?

解答：

product = product * item;

求连乘，初值赋 1。

</td></tr>
<tr><td>55</td><td>

求 x^n

输入实数x和正整数n，

计算x^n = x * x * … * x

```
for (i = 1; i <= n ; i++)
   power = power * item (第i项)
```

item=?

</td><td>

第 8 个例子：求 x^n。

问题分析（逐步细化）：

product = product * item

item = x

</td></tr>
<tr><td>56</td><td>

源程序 求 x^n

```
#include <stdio.h>
int main (void)
{
   int i, n;
   double x, power ;

   printf ("Enter x, n: \n");
   scanf ("%lf%d", &x, &n) ;
   power = 1;              /* 置power的初值为1 */
   for (i = 1; i <= n; i++){   /* 循环重复n次，计算x的n次幂 */
      power = power * x;
   }
   printf ("%0.f\n", power);
}
```

</td><td>

展示、运行程序。

</td></tr>
<tr><td>57</td><td>

2.5 生成乘方表和阶乘表

例2-10 生成乘方表

输入一个正整数n，生成一张2的乘方表，输出2^0到2^n的值，可以调用幂函数计算2的乘方。

```
for (i = 0; i <= n ; i++){
   power = pow(2, i);   /*调用幂函数pow(2,i)计算2的i次方*/
   输出power的值;
}
```

</td><td>

提出问题：如何生成乘方表和阶乘表？

本节介绍为解决这个问题所编写的程序和涉及到的语言知识。

问题分析：

根据本章已介绍内容，生成一张表格用 for 循环，求 2 的乘方则调用库函数。

（1）循环变量 i：起点 0，终点 n；

（2）pow(2, i)计算 2^i。

</td></tr>
</table>

<table>
<tr><td>58</td><td>

源程序：生成乘方表

```
#include <stdio.h>
#include <math.h>
int main (void)
{
   int i, n;
   double power;
   printf ("Enter n:");
   scanf ("%d", &n);
   for (i = 0; i <= n ; i++){
      power = pow(2, i);  /* 调用幂函数pow(2,i)计算2的i次方 */
      printf ("pow(2,%d)= %.0f\n", i, power);
   }
  return 0;
}
```

```
Enter n: 4
pow(2,0)= 1
pow(2,1)= 2
pow(2,2)= 4
pow(2,3)= 8
pow(2,4)= 16
```

</td><td>展示、运行例 2-10 程序。</td></tr>
<tr><td>59</td><td>

例2-11 生成阶乘表

输入一个正整数n(n<=16)，生成一张阶乘表，输出0！到n！的值。要求定义和调用函数fact(n)计算n!，函数类型是double。

```
for (i = 0; i <= n ; i++){
   product = fact(i);     /* 调用自定义函数fact(i)计算i! */
   输出product的值;
}
```

```
for (i = 0; i <= n ; i++){
   power = pow(2, i);  /*调用幂函数pow(2,i)计算2的i次方*/
   输出power的值;
}
```

</td><td>用对比的方式改写程序，生成一张阶乘表。</td></tr>
<tr><td>60</td><td>

源程序：生成阶乘表

```
#include <stdio.h>
double fact (int n);  /* 自定义函数的声明 */
int main (void)
{  int i, n;
   double result;

   printf ("Enter n:");
   scanf ("%d", &n);
   for (i = 0; i <= n ; i++){
      result = fact (i);      /* 调用自定义函数fact(i)计算i! */
      printf ("%d!=%.0f\n", i, result);
   }
  return 0;
}
```

```
double fact (int n)      /* 函数首部 */
{  int i;  double product;
   product = 1;
   for (i = 1; i <= n; i++){
      product = product * i;
   }
   return  product;  /* 将结果回送主函数 */
}
```

```
Enter n: 3
0!=1
1!=1
2!=2
3!=6
```

</td><td>展示、运行例 2-11 程序。
说明两点：
（1）求 2^i 调用库函数 pow(2, i)，求阶乘调用自定义函数 fact(i)。两者的调用方式一样；
（2）库函数 pow(2, i)由系统定义，而 fact(i)不是库函数，需要自己定义。两者的定义方式不同。
函数的定义和调用不要展开，学生能接受并能模仿编程即可。</td></tr>
<tr><td>61</td><td>

函数的概念

- C语言中有两种类型函数
  - 标准库函数
  - 自定义函数
- 函数可以做到一次定义、多次调用
- 使用自定义函数的程序框架

```
double fact (int n);        /* 声明自定义函数，以分号结束 */
int main (void)
{       ……
    result = fact (i);      /* 调用自定义函数fact(i)计算i! */
        ……
}
/* 定义求 n! 的函数 */
```

</td><td>简单总结本节内容。
使学生了解自定义函数编程结构，能够模仿编程即可。</td></tr>
</table>

2.3 练习与习题参考答案

2.3.1 练习参考答案

2-1 输出短句（Programming in C is fun!）：在屏幕上显示一个短句“Programming in C is fun!”。试编写相应程序。

解答：

```
#include <stdio.h>
int main (void)
{
    printf ("Programming in C is fun!\n");

    return 0;
}
```

2-2 下列语句的运行结果是什么？与例 2-2 的运行结果有何不同？为什么？

printf("Programming is fun. And Programming in C is even more fun!\n");

解答：

运行结果：Programming is fun. And Programming in C is even more fun!

例 2-2 的运行结果分 2 行显示。原因在于，例 2-2 中在每条输出语句的结尾都采用了‘\n’换行。

2-3 输出倒三角图案：在屏幕上显示如下倒三角图案。试编写相应程序。

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

解答：

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

    return 0;
}
```

2-4 温度转换：求华氏温度 150°F 对应的摄氏温度（计算公式同例 2-3）。试编写相应程序。

解答：

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

```
{
    int celsius, fahr;

    fahr = 150;
    celsius = 5 * (fahr - 32) / 9;
    printf ("fahr = %d, celsius = %d\n", fahr, celsius);

    return 0;
}
```

2-5 算术表达式 5*(fahr-32)/9 能改写成 5(fahr-32)/9 吗？为什么？如果将其改写为 5/9*(fahr-32)，会影响运算结果吗？

解答：

5(fahr-32)/9 不是合法的 C 表达式，因为不能省略运算符*；

5/9*(fahr-32)的值为 0。

2-6 计算物体自由下落的距离：一个物体从 100 米的高空自由落下，求它在前 3 秒内下落的垂直距离。设重力加速度为 10m/s2。试编写相应程序。

解答：

```
#include <stdio.h>
int main (void)
{
    double height;

    height = 0.5 * 10 * 3 * 3;
    printf ("height = %.2f\n", height);

    return 0;
}
```

2-7 输入提示和输入语句的顺序应该如何安排？例 2-5 中，scanf("%lf%lf%d", &initial, &factor, &day) 能改写为 scanf("%lf%d%lf", &initial, &factor, &day) 吗？为什么？能改写为 scanf("%lf%d%lf", &initial, &day, &factor) 吗？如果可以，其对应的输入数据是什么？

解答：

输入提示在前，输入语句在后。

不能改写为 scanf("%lf%d%lf", &initial, &factor, &day)，因为%d 与浮点型变量 factor 不匹配，%lf 与整型变量 day 不匹配。

与 scanf("%lf%d%lf", &initial, &day, &factor)对应的输入数据为：1.0　365　0.01

2-8 计算摄氏温度：输入华氏温度，输出对应的摄氏温度，计算公式同例 2-3。试编写相应程序。

解答：

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

```
    int celsius, fahr;

    scanf ("%d", &fahr);
    celsius = 5 * (fahr - 32) / 9;
    printf ("Celsius = %d\n", celsius);

    return 0;
}
```

2-9 整数四则运算：输入 2 个正整数，计算并输出它们的和、差、积、商。试编写相应程序。

解答：

```
#include <stdio.h>
int main (void)
{
    int num1, num2;

    scanf ("%d%d", &num1, &num2);
    printf ("%d + %d = %d\n", num1, num2, num1+num2);
    printf ("%d - %d = %d\n", num1, num2, num1-num2);
    printf ("%d * %d = %d\n", num1, num2, num1*num2);
    printf ("%d / %d = %d\n", num1, num2, num1/num2);

    return 0;
}
```

2-10 计算分段函数 I（判断 x 是否不为 0）：编写程序，输入 x，计算并输出下列分段函数 f(x) 的值（保留 1 位小数）。试编写相应程序。

$$y = f(x) = \begin{cases} \dfrac{1}{x} & x \neq 0 \\ 0 & x = 0 \end{cases}$$

解答：

```
#include <stdio.h>
int main (void)
{
    double x, y;

    scanf ("%lf", &x);
    if (x != 0){
        y = 1/x;
    }
    else{
        y = 0;
    }
    printf ("f(%.1f) = %.1f\n", x, y);
```

```
    return 0;
}
```

2-11 计算分段函数 II（判断 x 是否小于 0）：输入 x，计算并输出下列分段函数 f(x)的值（保留 2 位小数）。可在头文件中包含 math.h，并调用 sqrt 函数求平方根，调用 pow 函数求幂。试编写相应程序。

$$y = f(x) = \begin{cases} (x+1)^2 + 2x + \dfrac{1}{x} & x < 0 \\ \sqrt{x} & x \geq 0 \end{cases}$$

解答：

```
#include <stdio.h>
#include <math.h>
int main (void)
{
    double x, y;

    scanf ("%lf", &x);
    if (x >= 0){
        y = sqrt(x);
    }
    else{
        y = pow (x+1, 2) + 2*x + 1/x;
    }
    printf ("f(%.2f) = %.2f\n", x, y);

    return 0;
}
```

2-12 输出华氏-摄氏温度转换表:输入两个整数 lower 和 upper，输出一张华氏－摄氏温度转换表，华氏温度的取值范围是[lower, upper]，每次增加 2°F，计算公式同例 2-6。试编写相应程序。

解答：

```
#include <stdio.h>
int main (void)
{
    int fahr, lower, upper;
    double celsius;

    scanf ("%d%d", &lower, &upper);
    if (lower > upper){
        printf ("Invalid.\n");
    }
    else{
```

```
        printf("fahr celsius:\n");
        for (fahr = lower; fahr <= upper; fahr = fahr + 2){
            celsius = 5.0 * (fahr - 32.0) / 9 ;
            printf ("%d%6.1f\n", fahr, celsius);
        }
    }

    return 0;
}
```

2-13 求给定序列前 N 项和（1 + 1/2 + 1/3 + ...）：输入一个正整数 n，计算序列 1 + 1/2 + 1/3 + ... 的前 n 项之和。

解答：

```
#include <stdio.h>
int main (void)
{
    int i, n;
    double sum;

    scanf ("%d", &n);
    sum = 0;
    for (i = 1; i <= n; i++){
        sum = sum + 1.0 / i;
    }
    printf ("sum = %.6f\n", sum);

    return 0;
}
```

2-14 求给定序列前 N 项和（1 + 1/3 + 1/5 + ...）：输入一个正整数 n，计算序列$1+\frac{1}{3}+\frac{1}{5}+.....$的前 n 项之和。

解答：

```
#include <stdio.h>
int main (void)
{
    int i, n;
    double sum;

    scanf ("%d", &n);
    sum = 0;
    for (i = 1; i <= n; i++){
        sum = sum + 1.0/(2*i-1);
    }
```

```
    printf("sum = %.6f\n", sum);

    return 0;
}
```

2-15 求给定序列前 N 项和（1 - 1/4 + 1/7 - 1/10 + ... ）：输入一个正整数 n，计算序列 $1-\frac{1}{4}+\frac{1}{7}-\frac{1}{10}+\frac{1}{13}-\frac{1}{16}+.....$ 的前 n 项之和。试编写相应程序。

解答：

```
#include <stdio.h>
int main (void)
{
    int flag, i, n, denominator;
    double item, sum;

    scanf ("%d", &n);
    sum = 0;
    flag = 1;
    denominator = 1;
    for (i = 1; i <= n; i++){
        item = flag * 1.0 / denominator;
        sum = sum + item;
        flag = -flag;
        denominator = denominator + 3;
    }
    printf ("sum = %.3f\n", sum);

    return 0;
}
```

2-16 执行下列程序段后，sum 的值是____。

```
for (i = 1; i <= 10; i++){
    sum = 0;
    sum = sum + i;
}
```

解答：sum 的值是 10。

2-17 生成 3 的乘方表：输入一个正整数 n，生成一张 3 的乘方表，输出 3^0 到 3^n 的值，可以调用幂函数计算 3 的乘方。试编写相应程序。

解答：

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

```
    int i, n;
    double mypow;

    scanf ("%d", &n);
    for (i = 0; i <= n; i++){
        mypow = pow(3, i);
        printf ("pow(3,%d) = %.0f\n", i, mypow);
    }

    return 0;
}
```

2-18 求组合数：根据下列公式可以算出从 n 个不同元素中取出 m 个元素（m≤n）的组合数。编写程序，输入 2 个正整数 m 和 n（m≤n），计算并输出组合数。要求定义和调用函数 fact(n) 计算 n!，函数类型是 double。

$$C_n^m = \frac{n!}{m!(n-m)!}$$

解答：

```
#include <stdio.h>
double fact (int n);
int main (void)
{
    int m, n;
    double s;

    scanf ("%d%d", &m, &n);
    s = fact (n)/(fact (m) * fact (n-m));
    printf ("result = %.0f\n", s);

    return 0;
}
double fact (int n)
{
    int i;
    double product;

    product = 1;
    for (i = 1; i <= n; i++){
        product = product * i;
    }

    return product;
}
```

2.3.2 习题参考答案

一．选择题

1	2	3	4	5	6	7	8
C	D	B	A	B	C	D	C

二．填空题

1	整型、浮点型	2	a = 2#b = 3
3	temp = a b = temp	4	n%2 == 0（或 n%2!=1）
5	sqrt(s*(s-a)*(s-b)*(s-c))	6	#include <math.h> #include <stdio.h>
7	算术、关系、赋值	8	k = 0#flag = 1
9	顺序、分支、循环 （注：与排列顺序无关）	10	i = 1#s = 1.000 i = 2#s = 1.500 i = 3#s = 1.833 i = 4#s = 1.833

三．程序设计题

1．求整数均值：输入 4 个整数，计算并输出这些整数的和与平均值，其中平均值精确到小数点后 1 位。试编写相应程序。

解答：

```
#include <stdio.h>
int main (void)
{
    int num1, num2, num3, num4;
    double average, sum;

    scanf ("%d%d%d%d", & num1, & num2, & num3, & num4);
    sum = num1+ num2+ num3 + num4;
    average = sum / 4;
    printf ("Sum = %.0f; Average = %.1f\n", sum, average);

    return 0;
}
```

2．阶梯电价：为了提倡居民节约用电，某省电力公司执行“阶梯电价”，安装一户一表的居民用户电价分为两个“阶梯”：月用电量 50 千瓦时（含 50 千瓦时）以内的，电价为 0.53 元/千瓦时；超过 50 千瓦时的，超出部分的用电量，电价上调 0.05 元/千瓦时。输入用户的月用电量（千瓦时），计算并输出该用户应支付的电费（元）。若用电量小于 0，则输出"Invalid Value!"。试编写相应程序。

解答：

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

```
    double cost, e;

    scanf ("%lf", &e);
    if(e < 0){
        printf ("Invalid Value!\n");
    }
    else{
        if (e <= 50){
            cost = 0.53 * e;
        }
        else{
            cost = 0.53 * 50 + (e - 50) * 0.58;
        }
        printf ("cost = %.2f\n", cost);
    }

    return 0;
}
```

3．序列求和：输入两个正整数 m 和 n(0<m<=n)，求 $\sum_{i=m}^{n}(i^2+\frac{1}{i})$，结果保留 6 位小数。试编写相应程序。

解答：

```
#include <stdio.h>
int main (void)
{
    int i, m, n;
    double sum;

    scanf ("%d%d", &m, &n);
    if (m > 0 && m <= n){
        sum = 0;
        for (i = m; i <= n; i++){
            sum = sum + i * i + 1.0 / i;
        }
        printf ("sum = %.6f\n", sum);
    }

    return 0;
}
```

4. 求交错序列前 N 项和：输入一个正整数 n，计算交错序列 $1-\frac{2}{3}+\frac{3}{5}-\frac{4}{7}+\frac{5}{9}-\frac{6}{11}+.....$ 的前 n 项之和。试编写相应程序。

解答：

```
#include <stdio.h>
int main (void)
{
	int flag, i, n;
	double item, sum;

	scanf ("%d", &n);
	sum = 0;
	flag = 1;
	for (i = 1; i <= n; i++){
		item = flag * 1.0 * i / (2 * i - 1);
		sum = sum + item;
		flag = -flag;
	}
	printf ("%.3f\n", sum);

	return 0;
}
```

5．平方根求和：输入一个正整数 n，计算$1+\sqrt{2}+\sqrt{3}¼¼+\sqrt{n}$的值（保留 2 位小数）。可包含头文件 math.h，并调用 sqrt 函数求平方根。试编写相应程序。

解答：

```
#include <stdio.h>
#include <math.h>
int main (void)
{
	int i,n;
	double sum;

	scanf ("%d",&n);
	sum = 0;
	for (i = 1; i <= n; i++){
		sum = sum + sqrt(i);
	}
	printf ("sum = %.2f\n", sum);

	return 0;
}
```

6．求给定序列前 n 项和（1! + 2! + ...）：输入一个正整数 n（n<=12），求$e=1!+2!+3!+.....+n!$的值。要求定义和调用函数 fact(n)计算 n!，函数类型是 double。试编写相应程序。

解答：

```
#include <stdio.h>
```

```
int main (void)
{
	int i, n;
	double sum;
	double fact (int n);

	scanf ("%d", &n);
	if(n <= 12){
		sum = 0;
		for (i = 1; i <= n; i++){
			sum = sum + fact (i);
		}
		printf ("%.0f\n", sum);
	}

	return 0;
}

double fact (int n)
{
	int i;
	double product;

	product=1;
	for (i = 1; i <= n; i++)
		product = product * i;

	return product;
}
```

2.4 实验指导教材参考答案

实验 2.1 在屏幕上显示信息

一、调试示例

略

二、基础编程题

（1）输出短句：在屏幕上显示一个短句“Programming in C is fun!”。

解答：

```
# include <stdio.h>
int main(void )
{
	printf("Programming in C is fun!\n");
```

```
        return 0;
}
```

思考：

① 如何在屏幕上显示自己的学号、姓名和班级？

解答：将上述程序中的 printf("Programming in C is fun!\n");替换为 printf("学号 姓名 班级\n");

如：printf("31401234　张三　计算机 1401 班\n");

② 如何在屏幕上显示数字、英文字母和汉字等信息？例如：“你在计算中心 A1 机房吗？”

解答：将上述程序中的 printf("Programming in C is fun!\n");替换为 printf("你在计算中心 A1 机房吗？\n");

（2）输出三角形：在屏幕上显示下列三角形图案。（每行的*之间无空格）

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

解答：

```
# include <stdio.h>
int main(void )
{
        printf("****\n***\n**\n*\n");
        return 0;
}
```

（3）输出菱形图案：在屏幕上显示下列菱形图案。 （第 2 行两个 A 之间有 3 个空格）

```
  A
A   A
  A
```

解答：

```
#include <stdio.h>
int main(void)
{
        printf("  A\nA   A\n  A\n");
        return 0;
}
```

思考：如何在屏幕上显示一个由各种字符组成的图案？例如：

```
HHHHHH
HdddddH
HcccccH
HHHHHH
```

解答：

```
#include<stdio.h>
```

```
int main(void)
{
    printf("HHHHHH\nHdddddH\nHcccccH\nHHHHHH\n");
    return 0;
}
```

三、改错题

输出带框文字：在屏幕上输出以下 3 行信息。(源程序 error01_2.cpp)

```
*************
  Welcome
*************
```

解答：

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

    return 0;
}
```

四、拓展编程题

（1）打印菜单：在屏幕上打印如下 5 行菜单。

```
[1] Select crisps
[2] Select popcorn
[3] Select chocolate
[4] Select cola
[0] Exit
```

解答：

```
#include<stdio.h>
int main(void)
{
    printf("[1] Select crisps\n");
    printf("[2] Select popcorn\n");
    printf("[3] Select chocolate\n");
    printf("[4] Select cola\n");
    printf("[0] Exit\n");
    return 0;
}
```

（2）输出倒三角图案：在屏幕上显示下列倒三角图案。

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

```
  * *
    *
```

解答：

```
#include <stdio.h>
int main()
{
    printf("* * * *\n");
    printf(" * * *\n");
    printf("   * *\n");
    printf("     *\n");
    return 0;
}
```

实验 2.2 基本数据处理

一、调试示例

求华氏温度 100° F 对应的摄氏温度。计算公式如下：

$$c = \frac{5 \times (f - 32)}{9}$$

其中：c 表示摄氏温度；f 表示华氏温度。

解答：参见例 2-3。

二、基础编程题

（1）计算华氏温度：求摄氏温度 26°C 对应的华氏温度。

解答：

```
#include <stdio.h>
int main (void)
{
    int celsius, fahr;
    celsius = 26;
    fahr = 9* celsius /5+32;
    printf ("fahr = %d\n", fahr);

    return 0;
}
```

（2）计算物体自由下落的距离：计算物体自由下落的距离：一个物体从 100 米的高空自由落下，求它在前 3 秒内下落的垂直距离。设重力加速度为 10m/ s^2。

解答：参见练习 2-6。

（3）计算平均分：已知某位学生的数学、英语和计算机课程的成绩分别是 87 分、72 分和 93 分，求该生 3 门课程的平均分。

解答：

```
#include <stdio.h>
int main (void)
{
	int math, eng, comp, average;

	math = 87;
	eng = 72;
	comp = 93;
	average = (math + eng + comp) / 3;
	printf ("math = %d, eng = %d, comp = %d, average = %d\n", math, eng, comp, average);

	return 0;
}
```

三、改错题

计算某个数 x 的平方赋值给 y：分别以“y = x * x”和“x * x = y”的形式输出 x 和 y 的值。请不要删除源程序中的任何注释。

解答：

```
#include <stdio.h>
int main (void)
{
	int x, y;

	x = 3;
	y = x * x;
	printf ("%d = %d * %d\n", y, x, x);
	printf ("%d * %d = %d\n", x, x, y);

	return 0;
}
```

四、拓展编程题

（1）分糖果：幼儿园里，有 3 个小朋友编号分别是 1、2、3，他们按自己的编号顺序围坐在一张圆桌旁，每个小朋友的面前分别有 8、9、10 颗糖果。现在做一个分糖果游戏，从 1 号小朋友开始，将自己的糖果平均分成三份（如果有多余的糖果，则自己立刻吃掉），自己留一份，其余两份分给相邻座位的两个小朋友。接着，2 号、3 号小朋友也同样这么做。请问一轮后，每个小朋友面前分别有多少糖果？

解答：

```
#include <stdio.h>
int main (void)
{
	int child1=8, child2=9, child3=10;
	child1 = child1/3;	child2 = child2 + child1; child3 = child3 + child1;
	child2 = child2/3;	child1 = child1 + child2; child3 = child3 + child2;
```

```
    child3 = child3/3;    child1 = child1 + child3; child2 = child2 + child3;
    printf ("1:%d, 2:%d, 3:%d\n", child1, child2, child3);

    return 0;
}
```

（2）求一个三位数的各位数字：当 n 为 152 时，分别求出 n 的个位数字（digit1）、十位数字（digit2）和百位数字（digit3）的值。

解答：

```
#include <stdio.h>
int main (void)
{
    int n, digit1, digit2, digit3;

    n = 152;
    digit1 = n % 10;
    digit2 = (n / 10) % 10;
    digit3 = n / 100;
    printf ("%d = %d + %d*10 + %d*100\n", n, digit1, digit2, digit3);

    return 0;
}
```

实验 2.3 计算分段函数

一、调试示例

计算分段函数[1]：输入 x，计算并输出下列分段函数 f(x) 的值（保留 1 位小数）。

$$y = f(x) = \begin{cases} \dfrac{1}{x} & x \neq 0 \\ 0 & x = 0 \end{cases}$$

解答：参见练习 2-10。

二、基础编程题

（1）计算摄氏温度：输入华氏温度，输出对应的摄氏温度，计算公式同例 2-3。

解答：参见练习 2-8。

（2）计算存款利息：输入存款金额 money、存期 year 和年利率 rate，根据下列公式计算存款到期时的利息 interest（税前），输出时保留 2 位小数。

$$interest = money(1 + rate)^{year} - money$$

解答：

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

```
{
    double interest , money, year, rate;

    scanf ("%lf %lf %lf", &money, &year, &rate);
    interest = money * pow (1+rate, year) - money;
    printf ("interest = %.2f\n", interest);

    return 0;
}
```

（3）计算分段函数[2]：输入 x，计算并输出下列分段函数 f(x)的值（保留 2 位小数）。可在头文件中包含 math.h，并调用 sqrt 函数求平方根，调用 pow 函数求幂。试编写相应程序。

$$y = f(x) = \begin{cases} (x+1)^2 + 2x + \dfrac{1}{x} & x < 0 \\ \sqrt{x} & x \geqslant 0 \end{cases}$$

计算分段函数

解答：参见练习 2-11。

（4）整数算术运算：输入两个整数 num1 和 num2，计算并输出它们的和、差、积、商与余数。

解答：参见练习 2-9。

三、改错题

计算分段函数[3]：输入实数 x，计算并输出下列分段函数 f(x) 的值，输出时保留 1 位小数。

$$y = f(x) = \begin{cases} \dfrac{1}{x} & x = 10 \\ x & x \neq 10 \end{cases}$$

解答：

```
#include <stdio.h>
int main (void)
{
    double x, y;

    scanf ("%lf", &x);
    if (x != 10)
        y = x;
    else
        y = 1/x;
    printf ("f(%.1f) = %.1f\n", x, y);

    return 0;
}
```

四、拓展编程题

（1）阶梯电价：为了提倡居民节约用电，某省电力公司执行“阶梯电价”，安装一户一表的居民用户电价分为两个“阶梯”：月用电量 50 千瓦时（含 50 千瓦时）以内的，电价为 0.53 元/千瓦时；超过 50 千瓦时的，超出部分的用电量，电价上调 0.05 元/千瓦时。输入用户的月用电量（千瓦时），计算并输出该用户应支付的电费（元）。

解答：参见习题第 2 题。

（2）计算火车运行时间：输入两个整数 time1 和 time2，表示火车的出发时间和到达时间，计算并输出旅途时间。有效的时间范围是 0000 ~ 2359（前两位表示小时，后两位表示分钟），不需要考虑出发时间晚于到达时间的情况。

解答：

```
#include <stdio.h>
int main (void)
{
    int hh, mm, time1, time2;

    scanf ("%d %d", &time1, &time2);
    hh = time2 / 100 - time1 / 100;
    mm = time2 % 100- time1 % 100;
    if (mm < 0) {
        hh --;
        mm = mm + 60;
    }
    printf ("%02d:%02d\n", hh, mm);

    return 0;
}
```

（3）判断一个三位数是否为水仙花数：输入一个三位数 number（100≤number≤999），判断其是否为水仙花数，即其个位、十位、百位数字的立方和等于该数本身。若 number 不是三位数，则输出"Invalid Value!"。

```
#include <stdio.h>
int main (void)
{
    int number,a,b,c;
    printf("Enter number:");
    scanf ("%d", &number);
    if (number < 100 || number>999)
       printf("Invalid Value!");
    else{
        a=number%10;   b=number/10%10;   c=number/100;
        if(a*a*a+b*b*b+c*c*c==number)
```

```
            printf("Yes\n");
        else
            printf("No\n");
    }

    return 0;
}
```

实验 2.4 指定次数循环

一、调试示例

求 1 到 n 的和：输入一个正整数 n，计算序列 1 + 2 + 3 + ... 的前 n 项之和。
解答：参见例 2-7。

二、基础编程题

（1）求 n 分之一序列前 n 项和：输入一个正整数 n，计算序列 1 + 1/2 + 1/3 + ... 的前 n 项之和。
解答：参见练习 2-13。

（2）求奇数分之一序列前 n 项和：输入一个正整数 n，计算序列 1 + 1/3 + 1/5 + ... 的前 n 项之和。
解答：参见练习 2-14。

（3）求简单交错序列前 n 项和：输入一个正整数 n，计算序列 1 - 1/4 + 1/7 - 1/10 + ... 的前 n 项之和。
解答：参见练习 2-15。

三、改错题

输出华氏-摄氏温度转换表：输入两个整数 lower 和 upper，输出一张华氏－摄氏温度转换表，华氏温度的取值范围是[lower, upper]，每次增加 2°F。若输入的范围不合法，则输出"Invalid."。
解答：参见练习 2-12。

四、拓展编程题

（1）求交错序列前 n 项和：输入一个正整数 n，计算交错序列 1 - 2/3 + 3/5 - 4/7 + 5/9 - 6/11 + ... 的前 n 项之和，输出时保留 3 位小数。。
解答：参见习题第 4 题。

（2）求平方与倒数序列的部分和：输入两个正整数 m 和 n(0＜m≤n)，计算序列 m*m＋1/m＋(m+1)*(m+1)＋1/(m+1)＋(m+2)*(m+2)＋1/(m+2).....＋n*n＋1/n，结果保留 6 位小数。
解答：参见习题第 3 题。

（3）输出三位水仙花数：输入两个正整数 m 和 n(100≤m≤n≤999)，输出 m 和 n 区间内的所有水仙花数。若输入的 m 或者 n 不符合题目的要求，则输出"Invalid Value."。三位水仙花数即其个位、十位、百位数字的立方和等于该数本身。

解答：

```
#include <stdio.h>
int main (void)
{
    int m, n, number, a, b, c;
    printf("Enter m, n:");
    scanf ("%d%d", &m, &n);
    if (m>=100 && m<=n && n<=999)
    {   for(number=m; number<=n; number++) {
            a=number%10;   b=number/10%10;   c=number/100;
            if(a*a*a+b*b*b+c*c*c==number)
                printf("%d ", number);
        }
    }
    else    printf("Invalid Value.");

    return 0;
}
```

实验 2.5 使用函数

一、调试示例

求排列数：根据下列公式可以算出从 n 个不同元素中取出 m 个元素（m≤n）的排列数。输入两个正整数 m 和 n（m≤n），计算并输出排列数。要求定义和调用函数 fact(n)计算 n!，函数类型是 double。

解答：略

二、基础编程题

（1）生成 3 的乘方表：输入一个正整数 n，生成一张 3 的乘方表，输出 3^0 到 3^n 的值，可以调用幂函数计算 3 的乘方。

解答：参见练习 2-17。

（2）求平方根序列前 n 项和：输入一个正整数 n，计算$1+\sqrt{2}+\sqrt{3}¼¼+\sqrt{n}$的值（保留 2 位小数）。可包含头文件 math.h，并调用 sqrt()函数求平方根。

解答：参见习题第 5 题。

（3）求组合数：根据下列公式可以算出从 n 个不同元素中取出 m 个元素（m≤n）的组合数。输入两个正整数 m 和 n（m≤n），计算并输出组合数。要求定义和调用函数 fact(n)计算 n!，函数类型是 double。

$$C_n^m = \frac{n!}{m!(n-m)!}$$

解答：参见练习 2-18。

三、改错题

简单实现 x 的 n 次方：输入实数 x 和正整数 n，要求定义和调用 mypow（x,n）函数计算 x^n。

解答：

```
#include <stdio.h>
double mypow (double x, int n);
int main (void)
{
	int n;
	double result, x;

	printf("Enter x,n: ");
	scanf("%lf %d", &x, &n);
	result = mypow(x, n);
	printf("result=%f\n", result);

	return 0;
}
double mypow (double x, int n)
{
	int i;
	double result;

	result = 1.0;
	for (i = 1; i <= n; i++){
		result = result * x;
	}

	return result;
}
```

四、拓展编程题

（1）求幂之和：输入一个正整数 n，求下列式子的和，可以调用函数 pow()求幂。

$sum = 2^1 + 2^2 + 2^3 + ... + 2^n$

解答：

```
#include <stdio.h>
#include <math.h>
int main (void)
{
	int i, n;
	double sum;

	sum = 0;
	scanf("%d", &n);
```

```
    for (i = 1; i <= n; i++)
        sum = sum + pow (2, i);
    printf ("result = %.0f\n", sum);

    return 0;
}
```

（3）求阶乘序列前 n 项和：输入一个正整数 n，求 s=1!+2!+……+n!的值。要求定义和调用函数 fact(n)计算 n!，函数类型是 double。

解答：参见习题第 6 题