

## 第三章习题

### 1. 计算增长倍数

源代码

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

int main()
{
    float r = 0.09,n = 10,p,t;
    t = 1 + r;
    p = pow(t,n);
    printf("p=%f\n",p);

    getchar();
    return 0;
}
```

结果

```
p=2.367364
-----
Process exited after 2.006 seconds with return value 0
请按任意键继续. . .
```

小狼毫 半：

### 2. 存款利息的计算

源代码

```
#include<stdio.h>
#include<math.h>
int main()
{
    float p1,p2,p3,p4,p5; //五种方式的本息和
    float r1=0.0414,r2=0.0468,r3=0.054,r5=0.0585,rh=0.0072; //五种年利率
    float n = 5;

    p1 = 1000*(1+n*r5);
    p2 = (1000*(1+2*r2))*(1+3*r3);
    p3 = (1000*(1+3*r3))*(1+2*r2);
    p4 = 1000*pow(1+r1,n);
    p5 = 1000*pow(1+(rh/4),4*n);
}
```

```

printf("p1=%f\n",p1); //第一种方案
printf("p2=%f\n",p2); //第二种方案
printf("p3=%f\n",p3); //第三种方案
printf("p4=%f\n",p4); //第四种方案
printf("p5=%f\n",p5); //第五种方案

return 0;
}

```

### 输出结果

```

p1=1292.500000
p2=1270.763062
p3=1270.763184
p4=1224.863770
p5=1036.621094

```

```

-----
Process exited after 0.01238 seconds with return value 0
请按任意键继续. . .

```

小狼毫 半：

### 3. 还款月数

#### 源代码

```

#include<stdio.h>
#include<math.h>
int main()
{
    float m,d,p,r; //m 还清月数 d 贷款 p 每月还款额 r 月利率
    d = 300000.0, p = 6000.0, r = 0.01;
    float a,b; // 中间变量
    a = log(p / (p - d*r) );
    b = log(1 + r);
    m = a / b;
    printf("m = %.1f\n",m);

    return 0;
}

```

### 输出结果

```
m = 69.7
-----
Process exited after 0.2891 seconds with return value 0
请按任意键继续. . .

小狼毫 半：
```

#### 4. 分析下面的程序

```
#include<stdio.h>
int main()
{
    char c1,c2;
    c1 = 97;
    c2 = 98;
    printf("c1 = %c,c2 = %c\n",c1,c2);
    printf("c1 = %d,c2 = %d\n",c1,c2);
    return 0;
}
```

#### 运行时会输出什么信息？为什么？

输出结果

```
c1=a,c2=b
c1=97,c2=98
-----
Process exited after 0.2391 seconds with return value 0
请按任意键继续. . .
```

第一行的输出是以 %c 的格式输出，即输出字符。因为 97, 98 分别是字母 a,b 的 ASCII 代码，故第一行输出字符 a 和 b。第二行输出的格式是 %d，故输出 ASCII 码对应的十进制整数。

如果将程序第 4, 5 行改为 `c1 = 197; c2 = 198;` 运行时会输出什么信息？为什么？

```
#include<stdio.h>
int main()
{
    char c1,c2;
    c1 = 197;
    c2 = 198;
    printf("c1 = %c,c2 = %c\n",c1,c2);
    printf("c1 = %d,c2 = %d\n",c1,c2);
    return 0;
}
```

输出结果

```

c1=?c2=?
c1=-59,c2=-58

-----
Process exited after 0.2486 seconds with return value 0
请按任意键继续. . .

```

根据标准的 ASCII 码与常用字符的对照表，char 类型的字符的取值范围为 0-127，超过这个范围使用 %c 输出的结果是不可预料的。用 %d 输出时，输出 c1 = -59, c2 = -58，这是按补码形式输出的，内存字节第一位为 1 时，作为负数，59 与 197 之和等于 256，58 和 198 之和也等于 256。所以输出的值为 c1, c2 的值减去 256 所得。（这段解释是抄的。）

**如果将程序第 3 行改为 `int c1, c2;` 运行时会输出什么信息？为什么？**

```

#include<stdio.h>
int main()
{
    int c1, c2;
    c1 = 197;
    c2 = 198;
    printf("c1 = %c, c2 = %c\n", c1, c2);
    printf("c1 = %d, c2 = %d\n", c1, c2);
    return 0;
}

```

输出结果

```

c1=?c2=?
c1=197,c2=198

-----
Process exited after 0.2404 seconds with return value 0
请按任意键继续. . .

```

如果给 c1, c2 赋的值是 97、98，那么输出结果和第一次结果一样。如果赋的值为 197、198，使用 %c 输出时，结果是不可预料的字符。使用 %d 输出时得到 c1=197, c2=198，因为他们在 int 类型的有效范围内。（这段解释也是抄的。）

## 5. 输入解释

```

#include<stdio.h>
int main()
{
    int a, b;
    float x, y;
    char c1, c2;
    scanf("a=%d b=%d", &a, &b);
    scanf("%f %e", &x, &y);
    scanf("%c%c", &c1, &c2);

    printf("a=%d, b=%d, x=%f, y=%e, c1=%c, c2=%c\n", a, b, x, y, c1, c2); //输出显示验证
    return 0;
}

```

输出验证

```

a=3 b=7
8.5 71.82Aa
a=3,b=7,x=8.500000,y=7.182000e+001,c1=A,c2=a

-----
Process exited after 16.86 seconds with return value 0
请按任意键继续. . .

```

在输入 8.5 和 71.82 两个值给 x, y后要立马输入 A 和 a, 如果点击空格或者回车, 将无法得到正确结果, 因为空格和回车也是字符。按下空格或者回车后, 即将空格或者回车代表的字符赋给了 c1, 这回出现下面的结果:

```

a=3 b=7
8.5 71.82 Aa
a=3,b=7,x=8.500000,y=7.182000e+001,c1= ,c2=A

-----
Process exited after 11.71 seconds with return value 0
请按任意键继续. . .

```

## 6. 加密

源代码

```

#include<stdio.h>
int main()
{
    char c1='c',c2='h',c3='i',c4='n',c5='a';
    c1 = c1 + 4;
    c2 = c2 + 4;
    c3 = c3 + 4;
    c4 = c4 + 4;
    c5 = c5 + 4;

    putchar(c1); 使用 putchar 输出
    putchar(c2);
    putchar(c3);
    putchar(c4);
    putchar(c5);
    putchar('\n');

    printf("%c%c%c%c%c\n",c1,c2,c3,c4,c5); //使用 printf 输出

    return 0;
}

```

输出结果

```

Glmre
Glmre

-----
Process exited after 0.01414 seconds with return value 0
请按任意键继续. . .

```

第一行为 putchar 输出, 第二行为 printf 输出。

## 7. 计算

源代码

```

#include<stdio.h>
#define Pi 3.14

```

```

int main()
{
    float r,h,c,area,s,v1,v2; // c 周长 area 圆面积 s 圆球表面积 v1 球体积 v2 圆柱体积

    printf("请输入半径和圆柱体的高\n");
    scanf("%f,%f",&r,&h);
    c = 2*Pi*r;
    area = Pi*r*r;
    s = 4*Pi*r*r;
    v1 = (4/3.0)*Pi*r*r*r;
    v2 = Pi*r*r*h;

    printf("圆周长=%.2f\n",c);
    printf("圆面积=%.2f\n",area);
    printf("圆球表面积=%.2f\n",s);
    printf("圆球体积=%.2f\n",v1);
    printf("圆柱体体积=%.2f\n",v2);

    return 0;
}

```

输出结果

```

请输入半径和圆柱体的高
1.5,3
圆周长=9.42
圆面积=7.07
圆球表面积=28.26
圆球体积=14.13
圆柱体体积=21.19

-----
Process exited after 19.95 seconds with return value 0
请按任意键继续. . .

```

## 8. 字符 char int 分析

源代码 1

```

#include<stdio.h>

int main()
{
    char c1,c2;
    printf("请输入两个字符\n");
    c1 = getchar();
    c2 = getchar();

    printf("使用 putchar 输出");
    putchar(c1);
    putchar(c2);
    putchar('\n');

    printf("使用 printf 输出\n");
    printf("输出字符 %c,%c\n",c1,c2);
    printf("输出 ASCII 码 %d,%d\n",c1,c2);

    return 0;
}

```

输出结果

```
请输入两个字符
qw
使用 putchar 输出qw
使用 printf 输出
输出字符 q,w
输出 ASCII 码 113,119

-----
Process exited after 2.33 seconds with return value 0
请按任意键继续. . .
```

源代码 2

```
#include<stdio.h>

int main()
{
    int c1,c2;
    printf("请输入两个字符\n");
    c1 = getchar();
    c2 = getchar();

    printf("使用 putchar 输出");
    putchar(c1);
    putchar(c2);
    putchar('\n');

    printf("使用 printf 输出\n");
    printf("输出字符 %c,%c\n",c1,c2);
    printf("输出 ASCII 码 %d,%d\n",c1,c2);

    return 0;
}
```

输出结果

```
请输入两个字符
et
使用 putchar 输出et
使用 printf 输出
输出字符 e,t
输出 ASCII 码 101,116

-----
Process exited after 3.156 seconds with return value 0
请按任意键继续. . .
```

源代码 3

```
#include<stdio.h>

int main()
{
    char c1,c2;
    int c3,c4;
```

```

printf("请输入两个整数\n");
scanf("%d %d",&c3,&c4);
c1=c3;
c2=c4;
printf("输出字符结果 %c,%c\n",c1,c2);
printf("输出 ASCII 码 %d,%d\n",c1,c2);

return 0;
}

```

输出结果

```

请输入两个整数
87 98
输出字符结果 w,b
输出 ASCII 码 87,98

-----
Process exited after 15.95 seconds with return value 0
请按任意键继续. . .

```

```

请输入两个整数
130 133
输出字符结果 ??
输出 ASCII 码 -126,-123

-----
Process exited after 9.913 seconds with return value 0
请按任意键继续. . .

```

结果分析可知

1. 变量 c1,c2 可定义为字符型或整型，二者皆可。
2. 如果要输出字符的 ASCII 码，只能使用 printf 函数。
3. 二者并不是无条件的等价，int 的范围比 char 的范围要大，在 char 的范围内，int 和 char 是可以相互替换，超过范围后会出现不可预料的错误。