

第三章 顺序程序设计

第 1 题 : `printf("%08d\n%8d",1234,1234);` 其中%08d 中 08 表示什么 ?

答 : 在 `printf` 中%08d 表示输出宽度为 8 , 不足 8 位的补 0 , %8d 表示输出宽度为 8 , 不足 8 位的补空格。本题输出 :

00001234

空空空空 1234

第三章 部分课后题解析 (P38)

3.1(1) `main(){`

`int x=-2;`

`printf("dec:%d,oct:%o,hex:%x,unsigned:%u\n" ,x,x,x,x);}`

分析 : o-8 进制 , x-16 进制 , u-无符号整型

第一步 : 将-2 看成 2 转成二进制 0000 0000 0000 0010

第二步 : 取反加 1 1111 1111 1111 1110

第三步 : 每三位一组转成 8 进制 (每四位一组转成 16 进制)

所以得到 $(-2)_{10} = (177776)_8 = (fffe)_{16}$

第四步 : 转成无符号型即第二步的值 , 即 $(-2)_{10} = (65534)_u$

注 : $2^{16}-1 = (65535)_{10} = (1111\ 1111\ 1111\ 1111)_2$

3.1(2) `main(){char c= 'B' ;`

`printf("dec:%d,oct:%o,hex:%x,ASCII:%c\n" ,c,c,c,c);`

`printf("c=%c\n" ,c+10);`

分析 : 我们知道字母 'a' = 97 (让你唯一记住的一个 ASCII 码值哦) , 所以 'A' = 97-32=65 , 'B' = 'A' + 1=66 ; 所以%d 对应的值为 66 , 又因为可以通过计算得到 $(66)_{10} = (102)_8 = (42)_{16}$; 因为 `c= 'B'` , 所以 `c+10= 'B' +10= 'L'` .

3.1(3) # 对%f (%c, %s, %d, %u) 格式无影响 ,
对%o 格式 , 输出时加前缀 0 (零) ,
对%x 格式 , 输出时候加前缀 0x。

```
float a=99.9;double b=98765.4567;  
printf( "(1)%-14.2f\n" ,a); 输出 ( 1 ) 99.90 空---空 ( 9 个空 )  
printf( "(2)%#20.3f\n" ,a); 输出 ( 2 ) 空---空 99.900 ( 14 个空 )  
printf( "(3)%6.4f\n" ,a); 输出 ( 3 ) 99.9000  
printf( "(4)%-lf\n" ,b); 输出 ( 4 ) 98765.456700  
printf( "(5)%#14.3lf\n" ,b); 输出 ( 5 ) 空---空 98765.457 ( 5 个空 )
```

3.2 scanf(" _%c_%c" ,&c1,&c2);格式特别注意 , 因为空格也是字符 , 所以含有空格的时候 , 必须老老实实的带空格输入 , 即_A_a 回车。

如果改为 scanf("%c%c" ,&c1,&c2);那么输入方法最好是 : Aa , 但是如果是下面的输入方法就是错误的 :

A 回车

a 回车 , 结束输入。

如果这时候你输入 A_a , 那么 c1=A,c2=_ , 而不是 c2=a ,

其中_ 表示空格。

3.3 int x=10; float pi=3.1416;

```
printf( "(1)%d\n" ,x); 输出为 ( 1 ) 10 ;  
printf( "(2)%6d\n" ,x); 输出为 ( 2 ) 空空空空 10 ;  
printf( "(3)%f\n" ,56.1); 输出为 ( 3 ) 56.100000 ;  
printf( "(4)%14f\n" ,pi); 输出为 ( 4 ) 空空空空空空 3.141600 ;  
printf( "(5)%e\n" ,568.1); 输出为 ( 5 ) 5.68100e+02 或  
5.681000e+002 都对 , 我们统一输出格式为前一个 ;  
printf( "(6)%14e\n" ,x); 输出为 ( 6 ) 空空空 3.14160e+00 或 空  
3.141600e+000 都对 , 我们统一输出格式为前一个 ;
```

`printf(“(7)%g\n”,x);` 输出为 (7)3.1416 ;g 为实数输出 ,比 float 精度少一位 (即最多小数点后面保留 5 位 , 多了四舍五入 , 少了原值输出)。

`printf(“(8)%12g\n”,x);` 输出为 (8)空空空空空空 3.1416 ;

输出分析 :

```
main(){ float a;  
    scanf("%3f",&a);  
    printf("%f\n",a);  
}
```

如果输入为: 12.345 回车, 那么返回值为: 12.000000 即只保留 12.作为 3 个长度, 因为是浮点类型, 所以后面保留 6 为小数。