

C语言位运算

GCC 编译命令

```
1 gcc -o bit_opts.exe bit_opts.c
2 gcc -g -o bit_opts.exe bit_opts.c # GDB Debugging
```

任务一：将输入的一个无符号整数的第 i 位置 0 或置 1

函数设计

```
1 void Task1()
2 {
3     printf("==== Task 1 ====\n");
4     unsigned int m = 0, i, num;
5
6     printf("Please input an unsigned interger: ");
7     scanf("%u", &m);
8     printf("Please input the ith and bit value: ");
9     scanf("%u%u", &i, &num);
10
11     printf("Before set %dth bit to %d:\n", i, num);
12     PrintBits(m);
13
14     printf("After set %dth bit to %d:\n", i, num);
15
16     if(num == 0) m &= ~(1 << i);
17     else m |= (1 << i);
18
19     PrintBits(m);
20     printf("\n");
21
22     return ;
23 }
```

测试数据

- 无符号整数: 127, 1024
- 比特位: 5, 8
- 置/复位: 0, 1

运行结果

```
○ ==== Task 1 ====  
Please input an unsigned interger: 127  
Please input the ith and bit value: 5 0  
Before set 5th bit to 0:  
1111111  
After set 5th bit to 0:  
1011111
```

```
○ ==== Task 1 ====  
Please input an unsigned interger: 1024  
Please input the ith and bit value: 8 1  
Before set 8th bit to 1:  
1000000000  
After set 8th bit to 1:  
1010000000
```

任务二：检测一个无符号整数的第 i 位置是 0 还是 1

函数设计

```
1 void Task2()  
2 {  
3     printf("==== Task 2 ====\n");  
4     unsigned int m = 0, pos = 0;  
5     printf("Please input an unsigned interger: ");  
6     scanf("%u", &m);  
7     printf("Please input the ith bit position: ");  
8     scanf("%u", &pos);  
9  
10    if(m & (1 << pos)) printf("The ith bit is 1.\n");
```

```

11     else printf("The ith bit is 0.\n");
12     PrintBits(m);
13     printf("\n");
14
15     return ;
16 }

```

测试数据

- 无符号整数: 12, 934
- 比特位: 2, 6

运行结果

```

==== Task 2 ====
Please input an unsigned interger: 12
Please input the ith bit position: 2
The ith bit is 1.
1100

```

```

==== Task 2 ====
Please input an unsigned interger: 934
Please input the ith bit position: 6
The ith bit is 0.
1110100110

```

任务三：在屏幕上输出一个整数的二进制表示

函数设计

```

1 void PrintBits(unsigned int m)
2 {
3     bool first = false;
4     for(int j=31;j≥0;j--){
5         if(m & (1 << j)){
6             printf("1");
7             first = true;
8         }else if(first) printf("0");
9     }

```

```

10     printf("\n");
11     return ;
12 }
13 void Task3()
14 {
15     printf("==== Task 3 ==== \n");
16     unsigned int m = 0;
17
18     printf("Please input an unsigned interger: ");
19     scanf("%u", &m);
20     PrintBits(m);
21     printf("\n");
22     return ;
23 }

```

测试数据

- 无符号整数：934

测试结果

```

==== Task 3 ====
Please input an unsigned interger: 934
1110100110

```

任务四：将一个16bit 整数转化为 BCD 码表示

函数设计

```

1 void Task4()
2 {
3
4     printf("==== Task 4 ==== \n");
5     unsigned int m = 0, n = 0;
6
7     printf("Please input an unsigned interger: ");
8     scanf("%u", &m);
9
10    for(int i=0; m > 0; i++){
11        n |= ((m % 10) << (4 * i));
12        m /= 10;

```

```

13     }
14
15     for(int j=15;j>=0;j--){
16         if(n & (1 << j)) printf("1");
17         else printf("0");
18         if(j % 4 == 0 ) printf(" ");
19     }
20
21     printf("\n\n");
22     return ;
23 }

```

测试数据

- 16bit 整数: 1234

运行结果

```

==== Task 4 ====
Please input an unsigned interger: 1234
0001 0010 0011 0100

```

任务五：检测当前计算机是大端机还是小端机

函数设计

```

1 void Task5()
2 {
3     printf("==== Task 5 ====\n");
4     union {
5         char c;
6         int i;
7     }un;
8
9     un.i = 1;
10    if(un.c == 1) printf("The machine is small end storage mode.\n");
11    else printf("The machine is large end storage mode.\n");
12    printf("\n");
13    return ;
14
15 }

```

运行结果

```
==== Task 5 ====  
The machine is small end storage mode.
```

任务六：编写一个 C 语言的宏函数，将 4 个 unsigned char 型变量合成一个 unsigned int 型变量

函数设计

```
1  #define COMBINE_BYTES(a, b, c, d) ((unsigned int)((a << 24) | (b << 16) |  
   (c << 8) | d))  
2  void Task6()  
3  {  
4      printf("==== Task 6 ====\n");  
5      unsigned char a, b, c, d;  
6      printf("Please input four char characters: ");  
7      scanf(" %c %c %c %c", &a, &b, &c, &d);  
8      printf("Output the four char characters bits:\n");  
9  
10     void print(unsigned char n){  
11         for(int j=7;j≥0;j--){  
12             if(n & (1 << j)) printf("1");  
13             else printf("0");  
14         }  
15         printf("\n");  
16         return ;  
17     }  
18     print(a);  
19     print(b);  
20     print(c);  
21     print(d);  
22  
23     unsigned int m;  
24     m = COMBINE_BYTES(a, b, c, d);  
25     printf("Output the combined unsigned int bits:\n");  
26     for(int j=31;j≥0;j--){  
27         if(m & (1 << j)) printf("1");  
28         else printf("0");  
29         if(j % 8 == 0 ) printf(" ");  
30     }  
31     printf("\n");
```

```
32     return ;
33
34 }
```

测试数据

- 数字字符: '1', '2', '3', '4'

运行结果

```
==== Task 6 ====
Please input four char characters: 1 2 3 4
Output the four char characters bits:
00110001
00110010
00110011
00110100
Ouput the combined unsigned int bits:
00110001 00110010 00110011 00110100
```

完整源代码

```
1  #include<stdio.h>
2  #include<stdbool.h>
3  #define COMBINE_BYTES(a, b, c, d) ((unsigned int)((a << 24) | (b << 16) |
4  (c << 8) | d))
5
6  void PrintBits(unsigned int m)
7  {
8      bool first = false;
9      for(int j=31;j>=0;j--){
10         if(m & (1 << j)){
11             printf("1");
12             first = true;
13         }else if(first) printf("0");
14     }
15     printf("\n");
16     return ;
17 }
18 void Task1()
```

```

19  {
20      printf("==== Task 1 ==== \n");
21      unsigned int m = 0, i, num;
22
23      printf("Please input an unsigned interger: ");
24      scanf("%u", &m);
25      printf("Please input the ith and bit value: ");
26      scanf("%u%u", &i, &num);
27
28      printf("Before set %dth bit to %d:\n", i, num);
29      PrintBits(m);
30
31      printf("After set %dth bit to %d:\n", i, num);
32
33      if(num == 0) m &= ~(1 << i);
34      else m |= (1 << i);
35
36      PrintBits(m);
37      printf("\n");
38
39      return ;
40  }
41  void Task2()
42  {
43      printf("==== Task 2 ==== \n");
44      unsigned int m = 0, pos = 0;
45      printf("Please input an unsigned interger: ");
46      scanf("%u", &m);
47      printf("Please input the ith bit position: ");
48      scanf("%u", &pos);
49
50      if(m & (1 << pos)) printf("The ith bit is 1.\n");
51      else printf("The ith bit is 0.\n");
52      PrintBits(m);
53      printf("\n");
54
55      return ;
56  }
57
58  void Task3()
59  {
60      printf("==== Task 3 ==== \n");
61      unsigned int m = 0;
62
63      printf("Please input an unsigned interger: ");
64      scanf("%u", &m);

```



```

65     PrintBits(m);
66     printf("\n");
67     return ;
68 }
69
70 void Task4()
71 {
72
73     printf("==== Task 4 ==== \n");
74     unsigned int m = 0, n = 0;
75
76     printf("Please input an unsigned interger: ");
77     scanf("%u", &m);
78
79     for(int i=0; m > 0; i++){
80         n |= ((m % 10) << (4 * i));
81         m /= 10;
82     }
83
84     for(int j=15; j ≥ 0; j--){
85         if(n & (1 << j)) printf("1");
86         else printf("0");
87         if(j % 4 == 0 ) printf(" ");
88     }
89
90     printf("\n\n");
91     return ;
92 }
93 void Task5()
94 {
95     printf("==== Task 5 ==== \n");
96     union {
97         char c;
98         int i;
99     }un;
100
101     un.i = 1;
102     if(un.c == 1) printf("The machine is small end storage mode.\n");
103     else printf("The machine is large end storage mode.\n");
104     printf("\n");
105     return ;
106
107 }
108 void Task6()
109 {
110     printf("==== Task 6 ==== \n");

```

```

111     unsigned char a, b, c, d;
112     printf("Please input four char characters: ");
113     scanf(" %c %c %c %c", &a, &b, &c, &d);
114     printf("Output the four char characters bits:\n");
115
116     void print(unsigned char n){
117         for(int j=7;j≥0;j--){
118             if(n & (1 << j)) printf("1");
119             else printf("0");
120         }
121         printf("\n");
122         return ;
123     }
124     print(a);
125     print(b);
126     print(c);
127     print(d);
128
129     unsigned int m;
130     m = COMBINE_BYTES(a, b, c, d);
131     printf("Output the combined unsigned int bits:\n");
132     for(int j=31;j≥0;j--){
133         if(m & (1 << j)) printf("1");
134         else printf("0");
135         if(j % 8 == 0 ) printf(" ");
136     }
137     printf("\n");
138     return ;
139
140 }
141 int main()
142 {
143     // running tasks in turn
144     Task1();
145     Task2();
146     Task3();
147     Task4();
148     Task5();
149     Task6();
150     return 0;
151 }
152

```