

Linux编程新手入门C语言中的移位操作 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/240/2021_2022_Linux_E7_BC_96_E7_A8_c103_240660.htm C语言中的移位操作，内容不多。

不过有些地方你不注意，就疏忽了。先做两个小题先。

(1) unsigned char x=3. x>1是多少？ (2) char x=3. x>1是多少？

(3) char x=-3. x>1是多少？ 3写成二进制数是00000011；-3写成二进制数是(补码)11111101。程序执行的时候，操作的是数值的编码表示，也就是数值在内存中的二进制表示。比如说，程序取-3的时候，就去取11111101。

(1)对无符号数3来说，x>1往右边移一位，由于是无符号数，所以逻辑右移，最右边一位移掉，最左边移进来的位补零，变成00000001，所以结果是1。

(2)对于有符号数3来说，x>1往右边移一位，由于是有符号数，可能发生逻辑右移，也可能发生算术右移，这一点，C标准并没有明确地指定是使用逻辑右移还是算术右移。但大多数的机器都使用算术右移，变成00000001，所以结果还是1。但是请注意，这只是说大多数的机器是这样的，你敢保证自己不会碰到特殊情况吗？

(3)对于有符号数-3来说，x

总结：左移时总是移位和补零。右移时无符号数是移位和补零，此时称为逻辑右移.而有符号数大多数情况下是移位和补最左边的位（也就是补最高有效位），移几位就补几位，此时称为算术右移。

附打印内存中字节编码的代码:

```
void print_char(char x){ unsigned char * bp=(unsigned char *)&x. int size=sizeof(x). for(int i=0. i
```

```
100Test 下载频道开通，各类考试题目直接下载。详细请访问 www.100test.com
```

```
}; }
```

```
}; }
```