

大小写相互转换

```
// 转小写
(char) ('a' | ' ') = 'a'
(char) ('A' | ' ') = 'a'

// 转大写
(char) ('b' & '_') = 'B'
(char) ('B' & '_') = 'B'

// 大小写互转
(char) ('d' ^ ' ') = 'D'
(char) ('D' ^ ' ') = 'd'
```

判断两个数是否异号

```
int x = -1, y = 2;
bool f = ((x ^ y) < 0); // true

int x = 3, y = 2;
bool f = ((x ^ y) < 0); // false
```

不用临时变量交换两个数

```
int a = 1, b = 2;
a ^= b;
b ^= a;
a ^= b;
// 现在 a = 2, b = 1
```

可以消除 n 的二进制中最后一位 1

```
n & (n - 1)
```

任意位置 置 1 or 0

```
// 从右往左数
// 注意：「<<」优先级大于「|」 「^=」
// 将第 i 位置1
used |= 1 << i;

// 将第 i 位 的1标记为 0
used ^= 1 << i;
```

判断某一位置是否位 1

```
((used >> i) & 1) == 1
```

最低位的 0 变成 1

```
n |= n + 1;
```

最低位的 1 变成 0

```
n &= n - 1;
```

xor 运算

```
0 ^ N = N
N ^ N = 0
A^B^C^D=D^B^C^A (满足交换率和结合率)
```