

## 20. 位运算

- 为什么需要位运算？→ 机器里的数字表示方式和存储格式就是二进制
- 十进制与二进制的转换
  - $4(d): 0100$
  - $8(d): 01000$
  - $5(d): 0101$
  - $6(d): 0110$

### 位运算符

- 左移 ( $\ll$ ):  $0011 \rightarrow 0110$
- 右移 ( $\gg$ ):  $0110 \rightarrow 0011$
- 取反 ( $\sim$ ):  $\sim 0011 \rightarrow 1100$
- 或 OR ( $|$ ):  $0011 | 1011 \rightarrow 1011$
- 与 AND ( $\&$ ):  $0011 \& 1011 \rightarrow 0011$
- 异或 XOR ( $\wedge$ ):  $0011 \wedge 1011 \rightarrow 1000$  (相同为 0, 不同为 1)

### XOR - 异或 $\wedge$

- 异或: 相同为 0, 不同为 1, 即, “不进位加法”.
- 操作特点:
  - $x \wedge 0 = x$
  - $x \wedge 1s = \sim x$   $1s = \sim 0$ , 即, 全 1
  - $x \wedge (\sim x) = 0$
  - $x \wedge x = 0$
  - $c = a \wedge b \rightarrow a \wedge c = b, b \wedge c = a$  交换两个数
  - $a \wedge b \wedge c = a \wedge (b \wedge c) = (a \wedge b) \wedge c$  associative

### 指定位置的位运算

1. 将 x 最右边的 n 位清零:  $x \& (\sim 0 \ll n)$ ;
2. 获取 x 的第 n 位值 (0 或者 1):  $(x \gg n) \& 1$ ;
3. 获取 x 的第 n 位的 幂值:  $x \& (1 \ll n)$ ;
4. 仅将第 n 位置为 1:  $x | (1 \ll n)$ ;
5. 仅将第 n 位置为 0:  $x \& (\sim (1 \ll n))$ ;
6. 将 x 最高位至第 n 位 (含) 清零:  $x \& ((1 \ll n) - 1)$ .

## 实战位运算要点 ( 重要 )

- 判断奇偶：
  - $x \% 2 == 1 \rightarrow (x \& 1) == 1$
  - $x \% 2 == 0 \rightarrow (x \& 1) == 0$
- 除以 2 :
  - $x >> 1 \rightarrow x / 2$
  - $\text{mid} = (\text{left} + \text{right}) >> 1; \rightarrow \text{mid} = (\text{left} + \text{right}) / 2$
- 清零最低位的 1 :
  - $x = x \& (x - 1)$
- 得到最低位的 1 :
  - $x \& -x$
- 零 0 :
  - $x \& \sim x \rightarrow 0$

#Algorithm/Part II : Theory/Algorithm#