**Rules**

1. If we XOR a number by itself, even number of times then it will give you 0.

* 10 ^ 10 = 0
* 10 ^ 10 ^ 10 ^ 10 = 0

1. If we XOR a number with itself, odd number of times, then it will give you the number itself.

* 10 ^ 10 ^ 10 = 10
* 10 ^ 10 ^ 10 ^ 10 ^ 10 = 10

1. Also XOR of a number with 0 gives you that number again.

* 10 ^ 0 = 10

1. Find i’th bit of a number by **(num >> i) & 1**

* 10 => 1010
* shift by 0 = 1010 & 0001 = 0
* shift by 1 = 0101 & 1 = 1
* shift by 2 = 0010 & 1 = 0
* shift by 3 = 0001 & 1 = 1

1. Right shift, num >> i

* num >> 2^i = num // 2^i

1. Left shift, num << i

* num << 2^i = num \* 2^i

1. Unset right most set bit (bit with value 1) of an binary representation by

**n = n & (n - 1)**

* 5 = 101, 4 = 100 = 100
* 4 = 100, 3 = 011 = 000

n & (n - 1) == 0 allways equal to 0 when n is power of 2 as power of 2 consist of only one set bit and n-1 reverse alll the bit

