**Bit Manipulation**

**Additional**

* XOR
  + for flipping selective bits, \_\_\_\_ is chosen.
  + for flipping a bit, XOR it with 1, it will get reversed.
  + N ^ 1 =
  + N ^ 0 =
* NOT
  + The bitwise complement operator, \_\_\_\_, flips every bit in a number.

**Usages**

* = N\*2
* = N\*pow(2,2)
* = N\*(pow(2,k))
* = floor(N/2)
* = floor(N/2^2)
* = floor(N/2^k)
* = last bit in N
* = last 2 bits in N
* = last 3 bits in N
* = last k bits in N
* = -N
* = least significant byte of integer or the last 8 bits of integer.
  + An Integer normally has 4 bytes(32 bits)
  + F in hex is 1111 in binary, so FF(or 0xFF) is 11111111 in binary
  + Doing \_\_\_\_\_\_\_\_\_ removes the first 3 bytes and only keeps the last byte(8 bits) of integer
  + Eg 1783 in binary is 11011110111
  + 1783 & 0xFF only keeps the last 8 bits of 11011110111, and is, 11110111, which is 247

**Example Interview Questions**

**Multiply a no by 2**

**Divide a no by 2**

**set the kth bit of N(counting from right) to 1.**



**clear the kth bit of N(counting from right).**



**toggle/flip the kth bit of N(counting from right).**



**turn off the first set bit(1 bit) of a number N.**



**get the count of 1s in a no.**



**How to calculate the no of bits to convert from no A to no B.**



**Check if N is a power of 2 or not.**



**Check if N is a power of 4 or not.**



**How to get the last 3 bits of an integer.**



**Get the 5 highest bits of an integer(8 bit integer).**



**check whether the kth bit in N is 1.**



**swap two nos using bitwise operations.**



**swap even and odd bits in a no(4 byte integer)**



**Misc**

* IP address
  + normally represented as A:B:C:D
  + has 4 bytes, each of A, B, C, D representing a byte(8 bits).
  + each of A, B, C, D is 1 byte or 8 bits, and can have values from 0 to 255
* Right most bit(assuming 16 bit integer)???
* Left most bit(assuming 16 bit integer)???
* Sign bit(assuming 16 bit integer)???