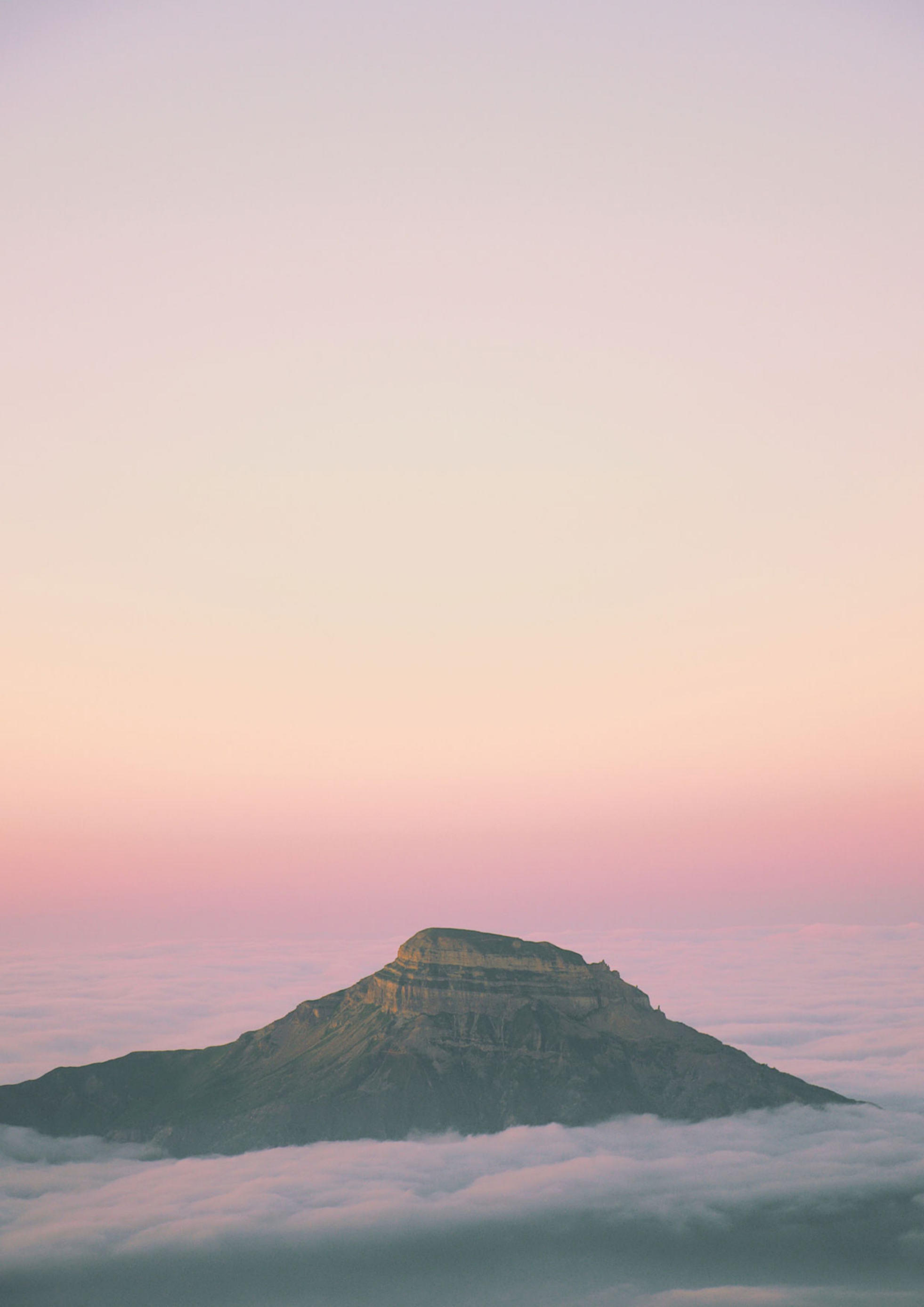

Bitwise Operators (Tricks and tips)

→ even or odd

→ In binary the least significant bit of even number is 0 and for odd it is 1.

$$x \& 1 == 0 \text{ (even)}$$

$$x \& 1 == 1 \text{ (odd)}$$

→ if it is a power of 2

→ A single bit is high, all other bit's are low

→ for x-1 the bits after the single high bit will become 1

$x = 001000$

$x-1 = 000111$

$$x \,\&\, x-1 == 0 \text{ (power of 2)}$$

$$x \,\&\, x-1 \;!= 0 \text{ (not a power of 2)}$$

→ edge case, dosent work for '0' it will come out as 0 for x & x-1 but 0 is not a power of 2

→ also include condition for negative Numbers

→ playing with k^{th} bit (from the right)

→ To check if k^{th} bit is set or not, we perform and operation of given number with the number 2^k if it results in 0 then the k^{th} bit is not set, otherwise it is set

x & 1<<k == 0 (not set)

x & 1<<k != 0 (set)

→ 1<<k means shift the number 1 by k positions

→ a<<b shift number a to left by b positions

→ To Toggle the k^{th} bit (remaining bits change)

x ^ (1<<k)

→ set the k^{th} bit (remaining bits change)

x | (1<<k)

→ unset the k^{th} bit (remaining bits change)

x & !(1<<k)

→ multiply or divide a number by 2^k

$x/2 \rightarrow x>>1$	$x*2 \rightarrow x<<1$
$x/4 \rightarrow x>>2$	$x*4 \rightarrow x<<2$
$x/8 \rightarrow x>>3$	$x*8 \rightarrow x<<3$
$x/2^k \rightarrow x>>k$	$x*2^k \rightarrow x<<k$

→ find out $x \% 2^k$

$$x \,\&\, ((1<<k)-1)$$

multiple of 2^k

$$\underbrace{00011001}\,10111$$

for $(1<<k)$	100000
for $(1<<k)-1$	011111

→ So, the remaining we get is remainder so,

$$x \,\&\, (1<<k)-1$$

→ Swap two numbers without using Temp variable

x	y	operation
x	y	—
$x \hat{}\, y$	y	$x = x \hat{}\, y$
$x \hat{}\, y$	x	$y = (x \hat{}\, y) \hat{}\, y = x$
y	x	$x = (x \hat{}\, y) \hat{}\, x = y$

→ same as, but bitoperations take less time

→ $x = x + y \quad y = y$

→ $x = x \quad y = x - y$

→ $x = x - y \quad y = y$

→ no. of sets bits in $A = x$

no. of set bits in $B = y$

no. of set bits in $(A \wedge B) = z$

→ z is even if $x + y$ is even

→ z is odd if $x + y$ is odd

→ $A \wedge B = x + y - 2k$, here we subtract $2k$ bits because if two corresponding bits in representation are '1' then both get cancelled and result becomes 0, i.e, we removed two bits, likewise if there k no. of 1 bits in the corresponding some positions in the number we remove $2k$ bits

→ Now $x + y - 2k$, is even if $x + y$ is even or it is odd

```
if(x==A){
   x=B
}else if(x==B){
   x=A
}
```

$x = A \wedge B \wedge x$

→ if x is A then A get cancelled and will result in B and vice versa

→ $A+B = (A \wedge B) + 2(A \& B)$

$A+B = (A | B) + (A \& B)$

→ finding no. of set bits in a number X

(only for c/c++)