32 bit signed int

Range of int : $[-2 \times 10^9, 2 \times 10^9]$

Range of long : $[-8 \times 10^{18}, 8 \times 10^{18}]$

64 bit signed int

Quiz1:

```
int a = 10^5
int b = 10^6
int c = a * b
```

$10^5 \times 10^6 = 10^{11}$

OVERFLOW

Quiz2:

```
int a = 10^5
int b = 10^6
long c = a * b
```

Overflow

Output register already store wrong answer

Quiz3:

```
int a = 10^5
int b = 10^6
long c = (long)(a*b)
```

Output register already store wrong answer

Quiz 4:

int a = 10^5

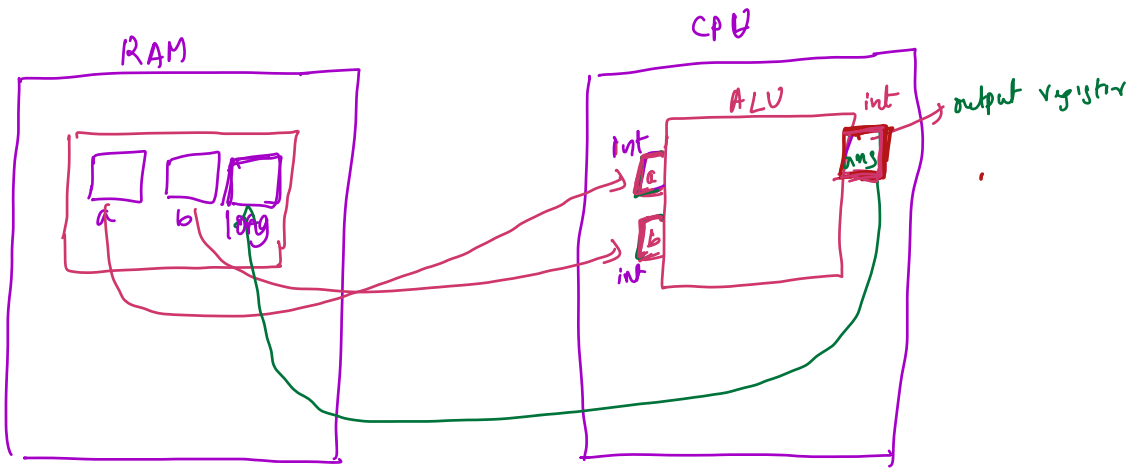
int b = 10^6

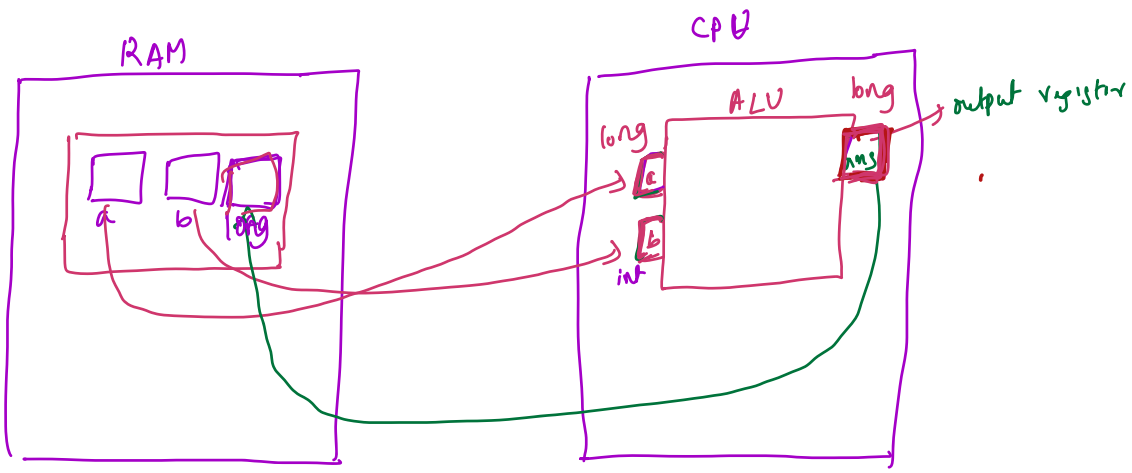
long c = ((long) a) * b

"NO OVERFLOW"

int c = a * b

a * b
a + b





```
long c = ax (b)
         |   |
        int  long
```

Question: Given an array of size N, return sum of all elements

```
long
int sum = 0;
for (i=0; i<N; i++){
    sum += A[i];
}
return sum;
```

// sum = sum + A[i]
long int

$N = 10^5$

$A[i] \leq 10^9$

$10^9 | 10^9 | 10^9 | 10^9 | 10^9 \ldots$ (10^5 terms)

$sum = 10^5 \times 10^9 = 10^{14}$

$N = 10^6 \Rightarrow$

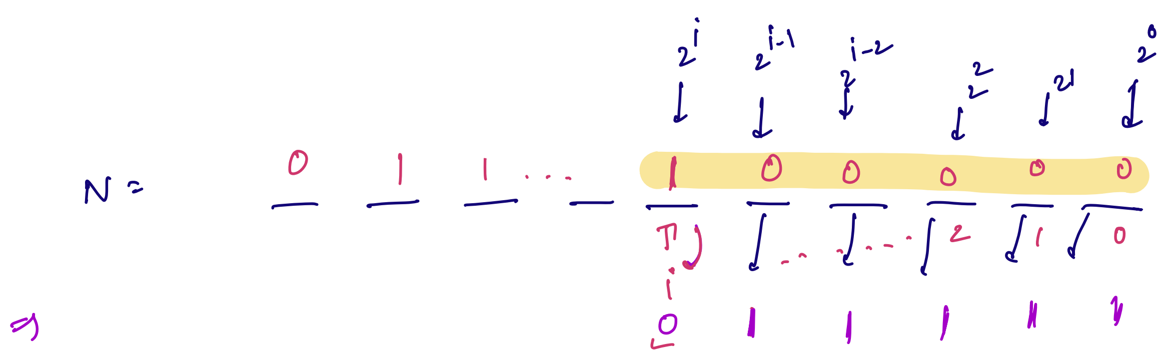
Question: Given an integer N, toggle all the bits starting from the rightmost set bit

N = 20 → 0000000 1 0 **1 0 0** (bit 31 … bit 4, 3, 2, 1, 0) ⇒ 1 0 0 1 1 ⇒ 19

N = 24 → 1 **1 0 0 0** ⇒ 1 0 1 1 1 ⇒ 23

Solution:

3 :	0 1 1	0 1 0	⇒ 2
5 :	1 0 1	1 0 0	⇒ 4
15 :	1 1 1 1	1 1 1 0	⇒ 14
6 :	1 1 0	1 0 1	⇒ 5
8 :	1 0 0 0	0 1 1 1	⇒ 7
20 :	1 0 1 0 0	1 0 0 1 1	⇒ 19
24 :	1 1 0 0 0	1 0 1 1 1	⇒ 23



Let's assume i to be the right most set bit of N

$$N' = N - 2^i + 2^{i-1} + 2^{i-2} + 2^{i-3} + \dots + 2^2 + 2^1 + 2^0$$

$2^0 + 2^1 + 2^2 + \dots + 2^{i-1}$

$[0, \dots, i-1]$

$i - 1 - 0 + 1 = i$

$a = 1$

$r = 2$

$K = i$

$$S_K = \frac{a(r^K - 1)}{r - 1} = \frac{1(2^i - 1)}{2 - 1} = 2^i - 1$$

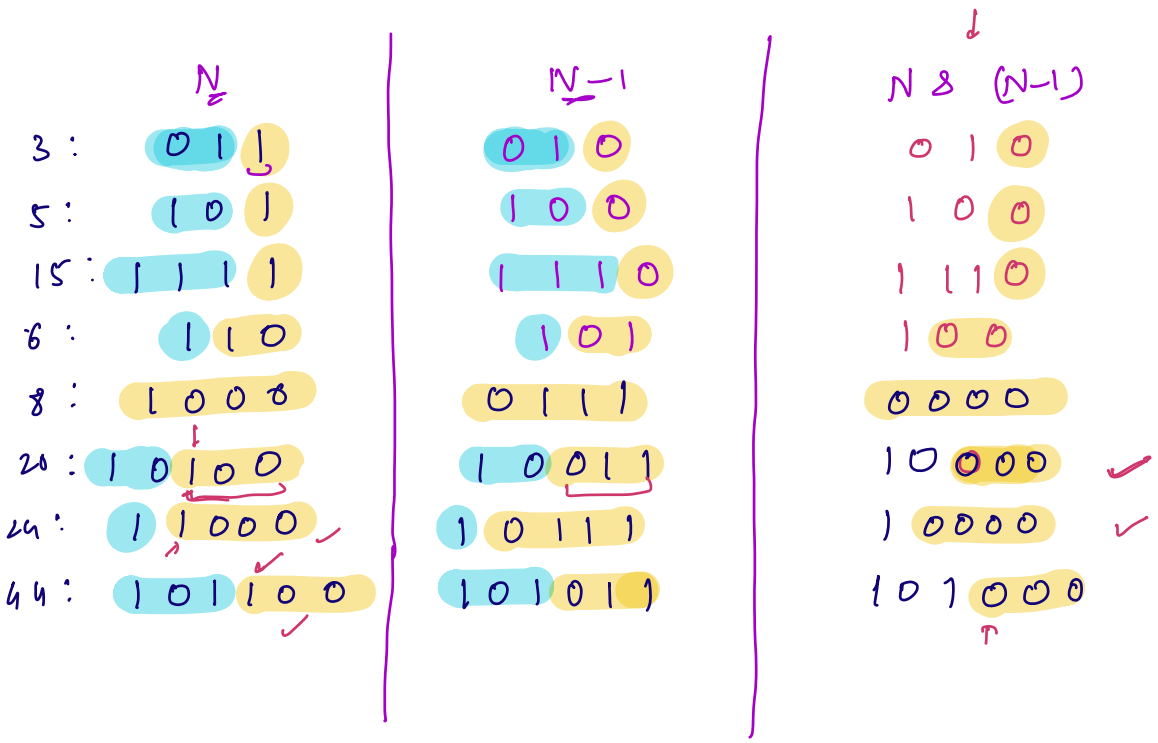
$$N' = N - 2^i + 2^i - 1$$

$$N' = N - 1$$

⇒ In $N-1$, all the bits from the rightmost set bit would be toggle

Question: Count no. of set bits

N = 10	1010	=> 2
N = 16	10000	=> 1
N = 15	01111	=> 4



$0 \& 1 = 0$

$1 \& 0 = 0$

$a \& (\sim a) = 0$

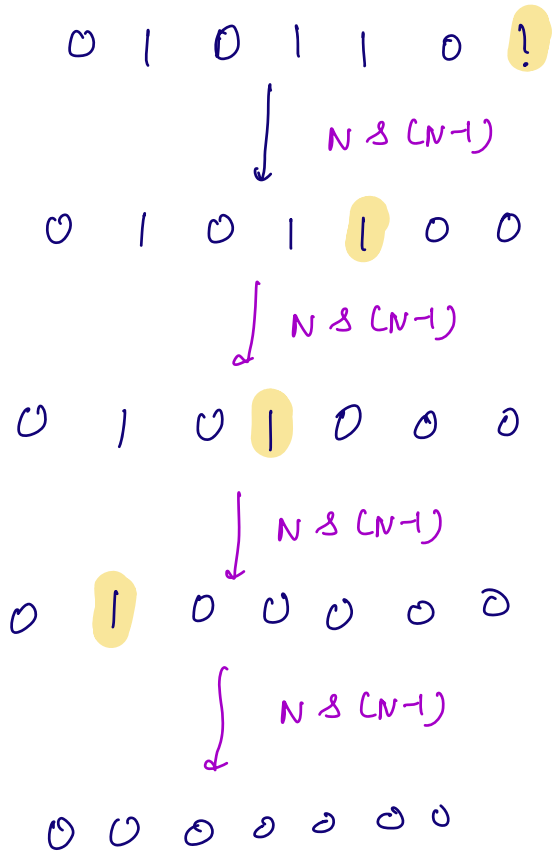
$a = 1011$

$\sim a = 0100$

0000

$a \& a = a$

N & (N-1) would unset the right most set bit



count = 1 2 3 4

15 = 1 1 1 1 => 4

16 = 1 0 0 0 0 => 1

17 = 1 0 0 0 1 => 2

#1tens = No. of set bits

```
int numSetBits(int N){
    int count=0;
    while(N>0){
        N= N & (N-1);
        count++;
    }
}
return count;
```

T.C : O(#set bits)

Worst Case, #set bits = $\log_2 N$

O(log N)

N = 15 → 1 1 1 1

15 → 1 1 1 1 ⇒ (4 bits) $\log_2 N$ ✓

No. of set bits

Question: Given 2 integers x, y find the number which has x continuous set bits and y unset bits

Ex:
- x = 2, y = 3 → 00000 1 1 000 ⇒ 24
- x = 3, y = 3 → 00000 1 1 1 000 ⇒ 56
- x = 2, y = 1 → 000000 1 1 0 ⇒ 6

x set, y unset

x y

x = 2, y = 4 → 1 1 0 0 0 0 ⇒ 48

5 4 3 2 1 0

Solution:

→ TASK 1: Get x continuous set bits

x = 5 → 00....0 1 1 1 1

~~00010 0000~~

~~1 1 1 0 1 1 1 1~~

1 0 0 0 0 0 ⇒ 0 1 1 1 1 1

(1<<x) → (1<<x) − 1

N-1

1<<1 = 10

1<<2 = 100

1<<3 = 1000

1<<4 = 10000

1<<5 = 100000

$x = 5$ $(1 << x) - 1$ $\quad 0\,1\,1\,1\,1\,1$

$y = 2$

$0\,1\,1\,1\,1\,1 \;<<\; 2 \;=\; 0\,1\,1\,1\,1\,1\,0\,0$

```
int func (int x, int y){
    return ((1 << x) - 1) << y
}
```

T.C: O(1)

Question: Given N, check if N is a power of 2

N = 32 → YES
N = 64 → YES
N = 48 → NO

$N = 00\ldots0\,1\,0000\ldots0$

if $N \,\&\, (N-1) == 0$: "POWER OF 2"
else "NO"

T.C: O(1)