Today's Content

Set ith bit

unset ith bit → Try your own.

check Bit

Count Bit

Negative numbers.

Ranges

Importance of Constraints

Unset or continuous Bits in 10

↓

If time allows.

Ques) Given N, i, check if ith bit in N is $\underbrace{\text{set}}_{\downarrow \; 1}$ or $\underbrace{\text{unset}}_{\downarrow \; 0}$.

$$\begin{array}{lccccc} & 4 & 3 & 2 & 1 & 0 \\ N = 21 : & 1 & 0 & 1 & 0 & 1 \end{array} \rightarrow \underline{\text{True}}.$$

i = 2

Brute force :- Convert to Binary, Store it and then check.

if 0th bit is 0 $(A \& 1) == 0$

else
<u>1</u>

$$\begin{array}{lccccccc} & 6 & 5 & 4 & 3 & 2 & 1 & 0 \\ N = \underline{82} & 1 & 0 & 1 & 0 & 0 & 1 & 0 \end{array}$$

<u>i=0</u>

$N >> \underline{4} \rightarrow 0 \; 0 \; 0 \; 0 \rightarrow 1 \rightarrow 0 \rightarrow 1$

$((N>>4) \& 1 == 0) \rightarrow$ 4th bit was unset

else it was <u>set</u>.

```
bool checkBit (int N, int i){
    if ((N>>i) &1) == 0 ){
            return false
    } else {
            return True }
```

T.C: <u>O(1)</u>.

3

N = 82

6	5	4	3	2	1	0
1	0	1	0	0	1	0

1 ← 1

i = 3 ⇒

$1<<3$

	6	5	4	3	2	1	0
	1	0	1	0	0	1	0
&	0	0	0	1	0	0	0
	0	0	0	0	0	0	0

N = 82

$1<<4$

6	5	4	3	2	1	0
1	0	1	0	0	1	0
0	0	1	0	0	0	0
0	0	1	0	0	0	0

→ $1<<4$.

when, $(N \& 1<<i) = 0$ (The bit was 0)

↳ $(1<<i)$ (That bit was 1)

Count Set Bits :-

		output
N=10,	1 0 1 0 :	2
N=27,	1 1 0 1 1 :	4

Approach1 :- Convert to Binary and count 1's.

Approach2 :- 32 bits. [0-31]

```
int c=0;
for (i=0; i<32; i++) {
    if (checkBit(N,i)) {
        c++
    }
}
return c;
```

T.C → O(1).
S.C → O(1).

Approach3 :-

N=45   c=0.

```
5 4 3 2 1 0
1 0 1 1 0 1     c++
0 1 0 1 1 0
0 0 1 0 1 1     c++
0 0 0 1 0 1     c++
0 0 0 0 1 0
0 0 0 0 0 1     c++
0 0 0 0 0 0
```

c=4 Ans.

```
int CountSetBits (int N) {
    int c=0;
    while (N>0){
        if (N&1 ==1) {
            c++
        }
        N = N>>1  → O(1);
    }
    return c;
}
```

T.C → (log N)

S.C → O(1).

* How -ve no's are stored.

8 bit numbers.

↗ Sign Bit

	7	6	5	4	3	2	1	0
+10 →	0	0	0	0	1	0	1	0
−10 →	1	0	0	0	1	0	1	0

	7	6	5	4	3	2	1	0
4 :	0	0	0	0	0	1	0	0
−4 :	1	0	0	0	0	1	0	0
+ 10 :	0	0	0	0	1	0	1	0
−14 :	1	0	0	0	1	1	1	0

1 more Problem:-

1 0 0 0 0 0 0 0 0 → −0

0 0 0 0 0 0 0 0 0 → 0

2's complement :-

$-a$ = 2's complement of a.

= 1's complement of $a + 1$.

$0 \rightleftarrows 1$

-10 :- 2's complement of 10

= 1's complement of 10 + 1

= $\sim(0\ 0001010) + 1$

→ 11110101

+ 1

11110110 → -10

$$\begin{array}{cccccccc} -2^7 & 2^6 & 2^5 & 2^4 & 2^3 & 2^2 & 2^1 & 2^0 \\ \uparrow & \nearrow & \nearrow & \nearrow & \nearrow & \nearrow & \nearrow & \nearrow \\ 7 & 6 & 5 & 4 & 3 & 2 & 1 & 0 \\ 1 & 1 & 1 & 1 & 0 & 1 & 1 & 0 \end{array}$$

→ $-2^7 + 2^6 + 2^5 + 2^4 + 2^2 + 2^1$ ⇒

$-128 + 64 + 32 + 16 + 4 + 2$

⇒ -10.

$$\begin{array}{rcccccccccc} & 10 & = & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ + & -4 & = & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\ \hline & \downarrow & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \end{array} \rightarrow 6$$

0 0 0 0 0 1 0 0

→ 1 1 1 1 1 0 1 1

→ + 1

1 1 1 1 1 1 0 0

// 4 bit no

$$\begin{array}{cccc} 3 & 2 & 1 & 0 \\ \underline{\quad} & \underline{\quad} & \underline{\quad} & \underline{\quad} \\ \downarrow & \downarrow & \downarrow & \downarrow \\ -2^3 & 2^2 & 2^1 & 2^0 \end{array}$$

8 bit no.

$$\underline{-2^7} \quad \underline{2^6} \quad \underline{2^5} \quad \underline{2^4} \quad \underline{2^3} \quad \underline{2^2} \quad \underline{2^1} \quad \underline{2^0}$$

↳ Most significant Bit

↓

MSB value is -ve.

(*) Convert Binary to decimal :-

4 bit No

-2^3	2^2	2^1	2^0		
1	0	1	1	→	-5
1	0	1	0	→	-6
0	0	1	1	→	3
1	0	0	0	→	-8
1	1	1	1	→	-1

(*) 8 bit number :-

-2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0		Decimal
0	0	0	1	0	1	0	1	→	21
1	0	0	1	0	1	0	1	→	-107
1	0	0	1	0	0	0	1	→	-111
0	0	0	1	0	0	0	1	→	17

n Bit numbers.

_ _ _ . . . . . _ _ _ → 2^0

-2^{n-1} 2^2 2^1

Max neg :- 1 0 0 0 0 . . . . → -2^{n-1}

Max +ve :- 0 1 1 1 1 1 1 . . 1 1 → $2^{n-1}-1$

2^{n-2} 2^1 2^0

g.p → $2^0 + 2^1 + 2^2 + \dots 2^{n-2}$

$a = 1, \ r = 2,$

Terms = $n-1$

$$\left(\frac{a(r^n - 1)}{r-1} \right)$$

$$\frac{1(2^{n-1}-1)}{2-1} = 2^{n-1}-1$$

Range of n bit no → $[-2^{n-1}$ to $2^{n-1}-1]$

	Bytes	bits	
byte/char	1	8	[-128 to 127]
int	4	32	$[-2^{31}$ to $2^{31}-1]$
	close approx →		$[-2*10^9$ to $2*10^9]$
long	8	64	$[-2^{13}$ to $2^{13}-1]$

close Approx [$-8*10^{18}$ to $8*10^{18}$]

$2^{10} = 1024 \simeq 10^3 \Rightarrow 2^{30} = 10^9 \rightarrow 2^{31} = 2*10^9$

↓

$2^{10} \simeq 10^3 \rightarrow (2^{30})^2 \simeq (10^9)^2 \longrightarrow 2^{60} \overset{*2^3}{\simeq} 10^{18}$

$2^{63} \simeq 8*10^{18}$

2 bit numbers :-

-2^1	2^0		
0	0	→	0
0	1	→	1
1	0	→	-2
1	1	→	-1

3 bit nos :-

min max

[-4 3]

-2^2	2^1	2^0		
0	0	0	→	0
0	0	1	→	1
0	1	0	→	2
0	1	1	→	3
1	0	0	→	-4
1	0	1	→	-3
1	1	0	→	-2
1	1	1	→	-1

Importance of Constraints :-

1) Given an array calc sum of it.

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

Constraints :-

$1 <= N <= 10^5$

$1 <= arr[i] <= 10^6$

$1 <= sum <= 10^{11}$

Constraints

↓

V.V. Important

→ TLE → TLE.

→ Datatypes → Error

Ques) Given 2 numbers a & b, return their prod.

```
prod (int a, int b) {
    int c = a * b
    long c = a * b
    long c = long (a * b)
    long c = (long) a * b;
    return c;
}
```

$1 <= a, b <= 10^6$

Take care :-

→ when we multiply int * int or long * long.

(*) Unsigned variables :- MSB weightage will be +ve.

C/ C++ / C#.

unsigned int x; → 32 bits $(0, 2^{32} - 1)$

↳ that x variable can't store a neg. no.

In java there is no such thing as unsigned.

Ques) When x, y set n continuous bits
& y unset bits.

n = 3, y = 2,   1 1 1 0 0 → 28

n = 5, y = 3,   1 1 1 1 1 0 0 0

---

3 bit no.
1 1 1 → 7
1 0 0 0 → 8 $(1<<3)-1$

4 bit no. →
1 1 1 1 → 15
1 0 0 0 0 → 16 $(1<<4)-1$

1 1 1 1 1 → $(1<<5)-1$

$$\left(\left((1<<n)-1\right)<<y\right)$$

T.C → O(1)

Q1 100 coins lying on a table.
10 of them are heads up 90 are on tails side.

split in 2 piles, such that there are same no. of heads in each pile.

90 { heads → h   Tails t

heads → 90-t
tails → 10-h) flip } 10 coins,

It is allowed, you can flip all coins of one pile at max once.

$h = 90 - t \rightarrow \underline{h + t = 90}$.

100 → 10 h
90 T

Part 1
6 heads
84 tails

Part 2
4 heads
6 tails

100 → 10h
90 T

Part 1
0 heads
90 tails

10 heads
0 tails.

19 blue

13 red ← 20 blue

13 red balls.

13th red ball ,

same color → blue ball, | diff → red ball.

blue	2 red balls	blue 1 red
O O	↓	
↓	-2 red	-1 blue.
-1 blue	+1 blue	