

num = 5

GOAL: Find No. of set bits for each No. in the range 0 to num and push them individually in an array.

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

[0, 1, 1, 2, 1, 2]

① Simple Soln.

Num=5

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

TIME COMPLEXITY: $O(N \log \underline{MAX})$

① Simple Soln.

Num=5

[0, 1, -----]

0	0	0	0 ←
0	0	1	1 —
0	1	0	2 .
0	1	1	3
1	0	0	4
1	0	1	5 —

TIME COMPLEXITY: $O(N \log \underline{\text{MAX}})$

① Simple Soln.

Num=5

[0, 1, -----]

0	0	0	0 ←
0	0	1	1 —
0	1	0	2
0	1	1	3
1	0	0	4
1	0	1	5 —

N
32
32 N

TIME COMPLEXITY: $O(N \log \underline{\text{MAX}})$

OBSERVATION

$$\hookrightarrow \text{if } x/2 = y$$

then

$$\boxed{\text{No. of set bits in } x - \text{No. of set bits in } y \leq 1}$$

e.g.

$$\begin{aligned} & x/2 = 3 \\ & x 7 \rightarrow 111 \\ & y 3 \rightarrow 011 \end{aligned} \quad \left. \begin{aligned} & 12/2 = 6 \\ & x 12 \rightarrow 1100 \\ & y 6 \rightarrow 0110 \end{aligned} \right\}$$

\hookrightarrow Dividing by 2 means right shift by 1 bit.
 \therefore LSB is lost.

OBSERVATION

$$\hookrightarrow \text{if } x/2 = y$$

then

$$\boxed{\text{No. of set bits in } x - \text{No. of set bits in } y \leq 1}$$

e.g.

$$\begin{aligned} & x/2 = 3 \\ & x 7 \rightarrow 111 \\ & y 3 \rightarrow \overline{011} \end{aligned} \quad \left. \begin{aligned} & 12/2 = 6 \\ & x 12 \rightarrow 1100 \\ & y 6 \rightarrow 0110 \end{aligned} \right\}$$

\hookrightarrow Dividing by 2 means right shift by 1 bit.
 \therefore LSB is lost.

OBSERVATION

$$\hookrightarrow \text{if } x/2 = y \quad \xrightarrow{\text{LSB} \rightarrow 0} \perp$$

then, $\boxed{\text{No. of set bits in } x - \text{No. of set bits in } y \leq 1}$

e.g.

$x \xrightarrow{\text{LSB}} 7/2 = 3$	$y \xrightarrow{\text{LSB}} 12/2 = 6$
$x \xrightarrow{\text{LSB}} 111 \rightarrow$	$y \xrightarrow{\text{LSB}} 011 \rightarrow$
$x \xrightarrow{\text{LSB}} 3 \rightarrow 011$	$y \xrightarrow{\text{LSB}} 6 \rightarrow 0110$

$\therefore \text{LSB is lost.}$

\Rightarrow Dividing by 2 means right shift by 1 bit.

\hookrightarrow when we have odd No, $\{x/2 = y\}$

then, No. of set bits in $x = 1 + \text{No. of set bits in } y$

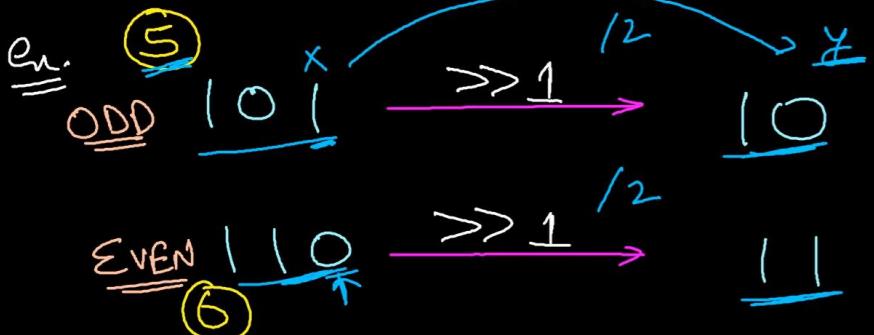
& for even No., No. of set bits in $x = \text{No. of set bits in } y$.

e.g. ⑤
ODD 101 $\xrightarrow{\gg 1}$ 10

EVEN 110 $\xrightarrow{\gg 1}$ 11

↪ when we have odd No, $\{ \underline{x}/2 = \underline{y} \}$
then, No of set bits in $x = 1 +$ No. of set bits in y

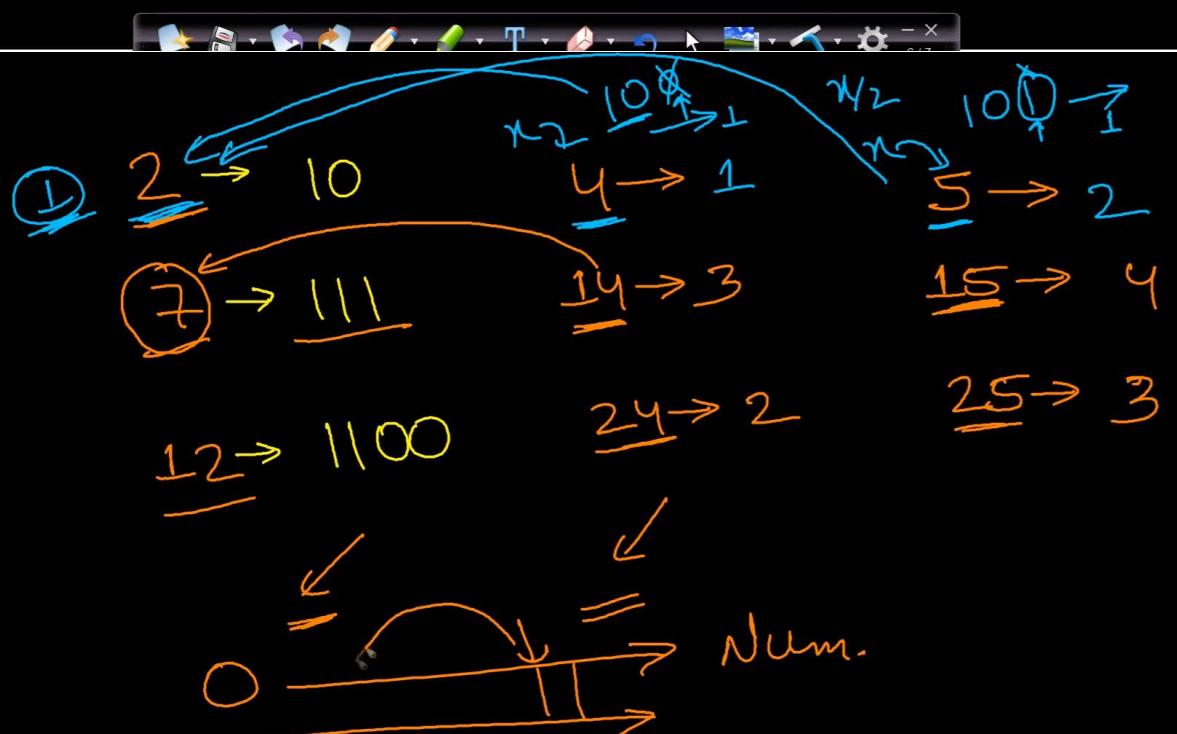
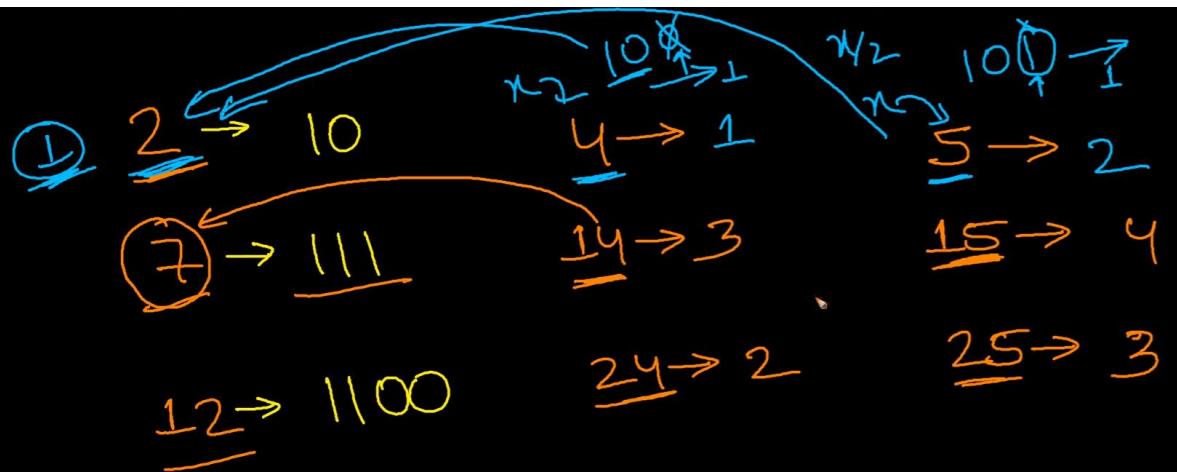
& for even No., No of set bits in $x =$ No. of set bits in y .



1 $\xrightarrow{2 \rightarrow 10}$ $\xrightarrow{x_2 \frac{10}{\cancel{0}} \gg 1}$ $\xrightarrow{4 \rightarrow}$ $\xrightarrow{5 \rightarrow}$
7 $\rightarrow 111$ $\xrightarrow{14 \rightarrow}$ $\xrightarrow{15 \rightarrow}$
12 $\rightarrow 1100$ $\xrightarrow{24 \rightarrow}$ $\xrightarrow{25 \rightarrow}$

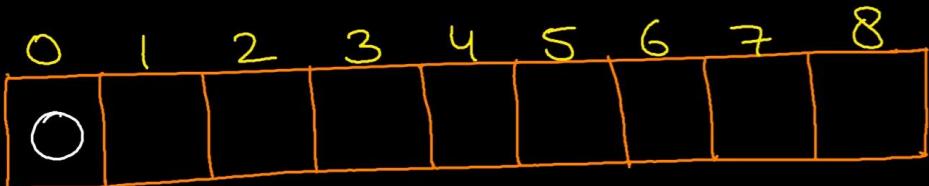
$\textcircled{1}$ $2 \rightarrow 10$ $n_2 \xrightarrow{10} 1$ $n_2 \xrightarrow{10} 1$
 $7 \rightarrow 111$ $4 \rightarrow 1$ $n \rightarrow 5 \rightarrow$
 $12 \rightarrow 1100$ $14 \rightarrow$ $15 \rightarrow$
 $24 \rightarrow$ $25 \rightarrow$


 $\textcircled{1}$ $2 \rightarrow 10$ $n_2 \xrightarrow{10} 1$ $n_2 \xrightarrow{10} 1$
 $7 \rightarrow 111$ $4 \rightarrow 1$ $n \rightarrow 5 \rightarrow 2$
 $12 \rightarrow 1100$ $14 \rightarrow$ $15 \rightarrow$
 $24 \rightarrow$ $25 \rightarrow$

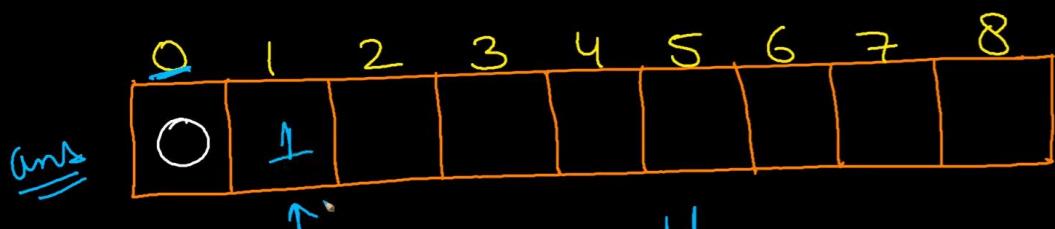
↳ We can compute current set bit count using previous count at $\frac{N}{2}$ in $O(1)$ time.

$$\underline{\text{Num}} = 8$$



↳ We can compute current set bit count using previous count at $\frac{N}{2}$ in $O(1)$ time.

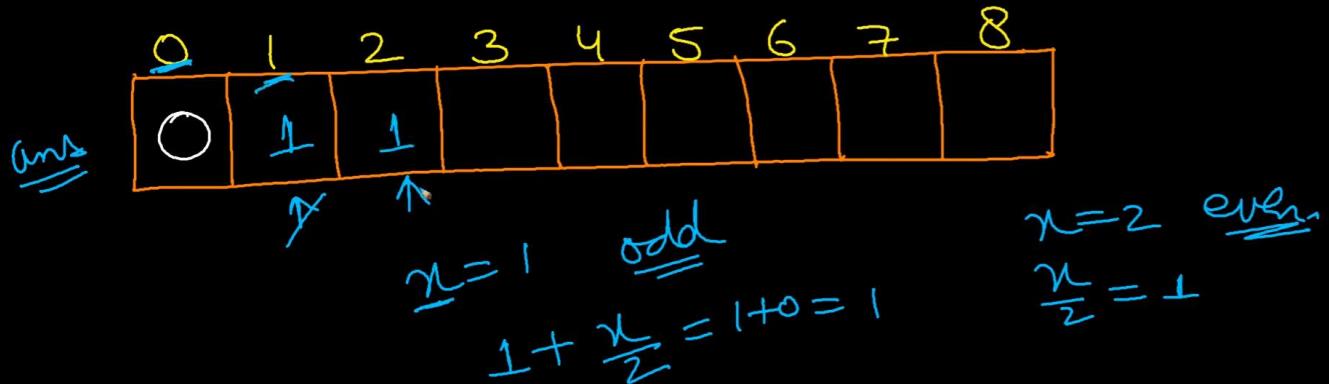
$$\underline{\text{Num}} = 8$$



$$\begin{aligned} \underline{x} &= 1 \quad \underline{\text{odd}} \\ 1 + \frac{\underline{x}}{2} &= 1+0=1 \end{aligned}$$

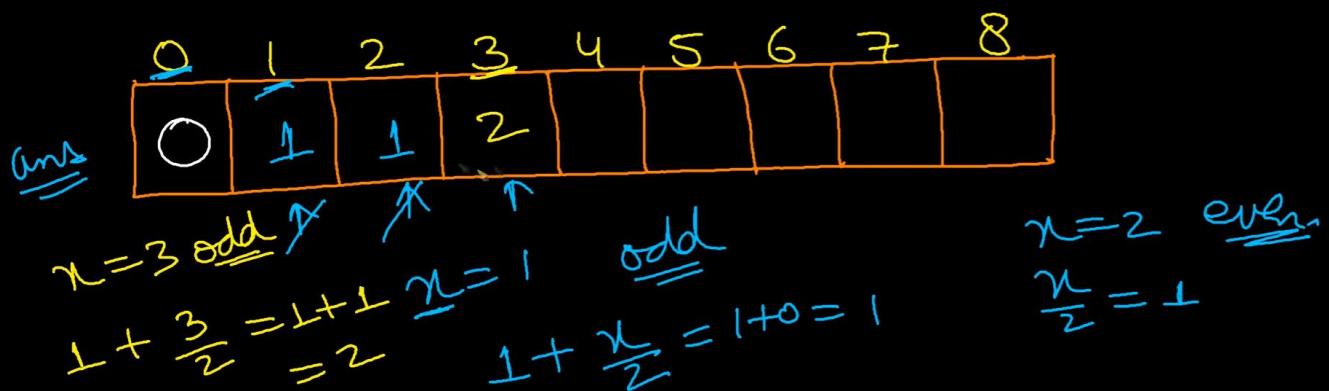
↳ We can compute current set bit count using previous count at $\frac{n}{2}$ in $O(1)$ time.

$$\underline{\text{Num}} = 8$$



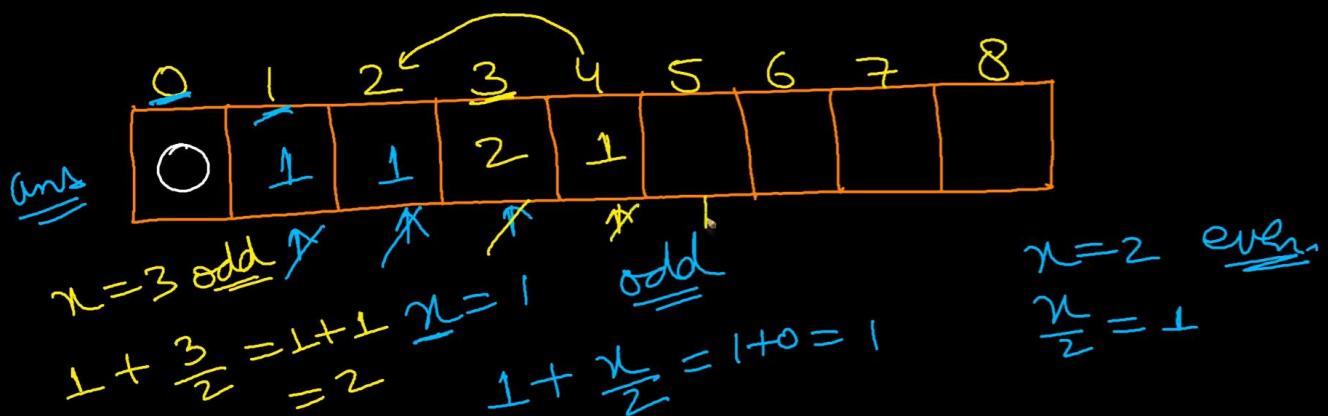
↳ We can compute current set bit count using previous count at $\frac{n}{2}$ in $O(1)$ time.

$$\underline{\text{Num}} = 8$$



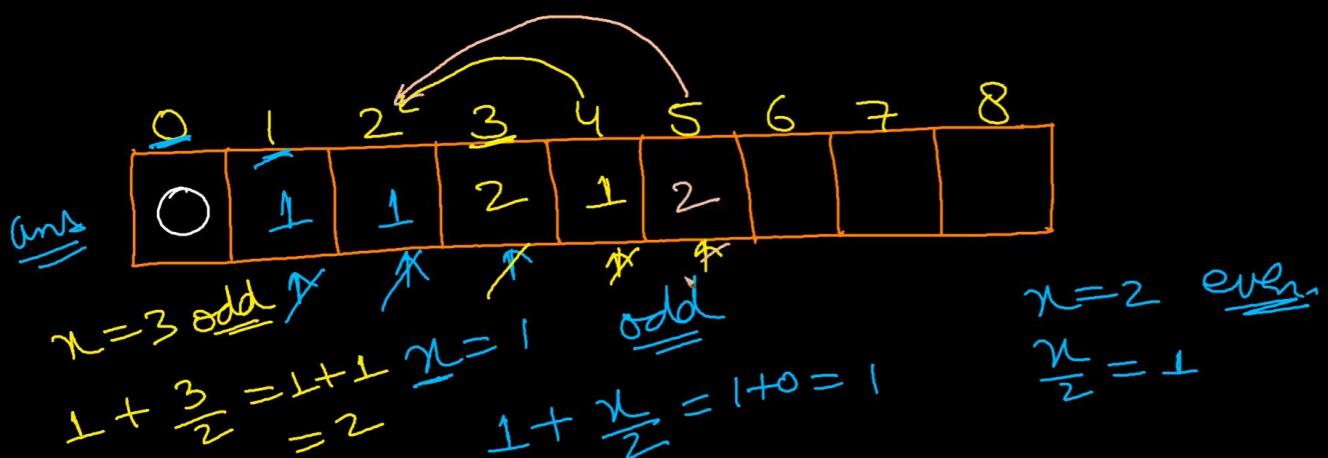
↳ We can compute current set bit count using previous count at $\frac{n}{2}$ in $O(1)$ time.

$$\underline{\text{Num}} = 8$$



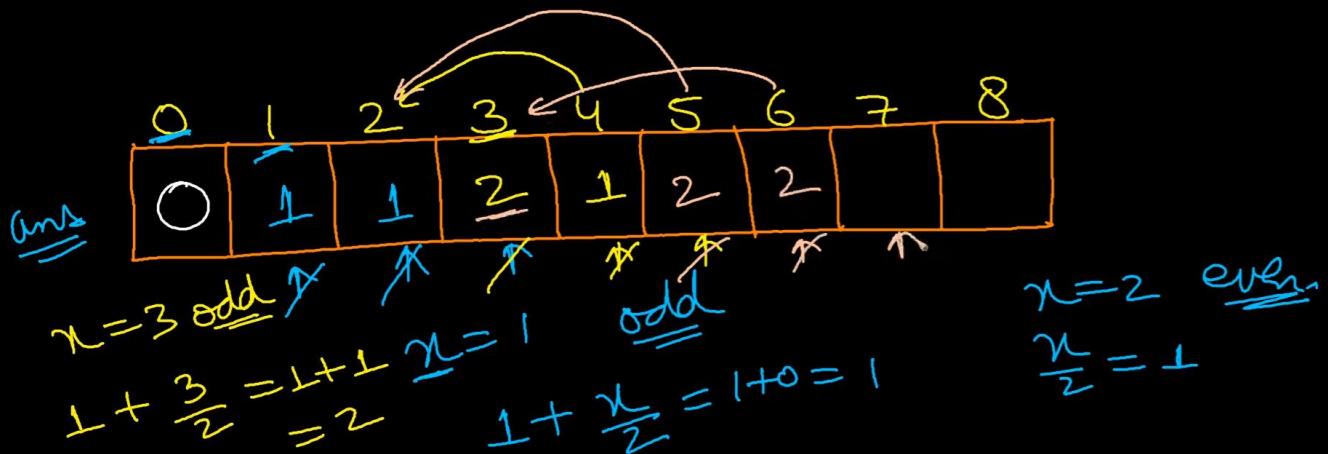
↳ We can compute current set bit count using previous count at $\frac{n}{2}$ in $O(1)$ time.

$$\underline{\text{Num}} = 8$$



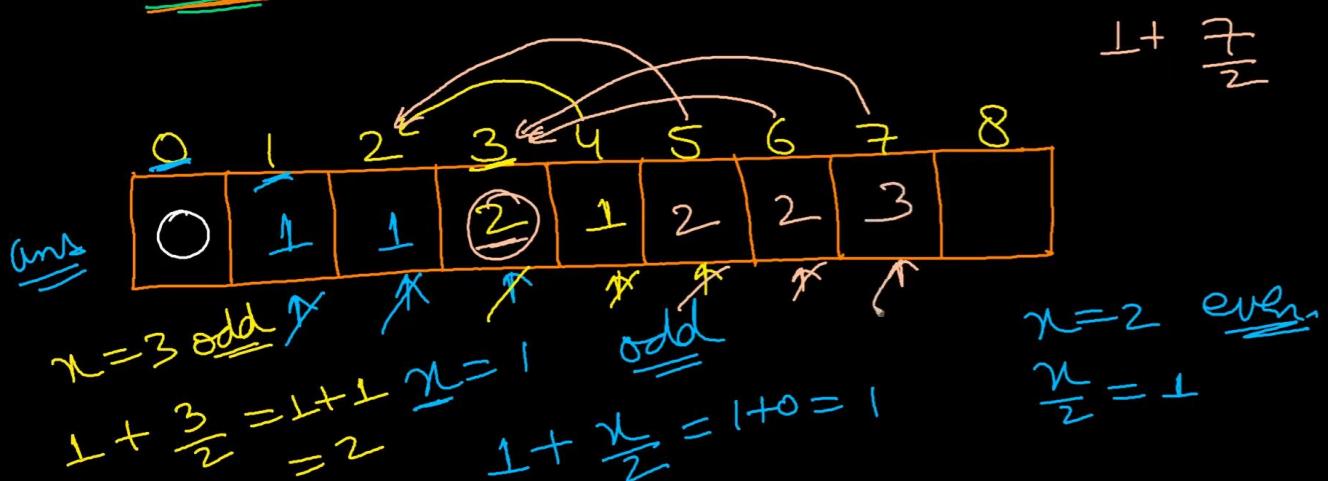
↳ We can compute current set bit count using previous count at $\frac{n}{2}$ in $O(1)$ time.

$$\underline{\text{Num}} = 8$$



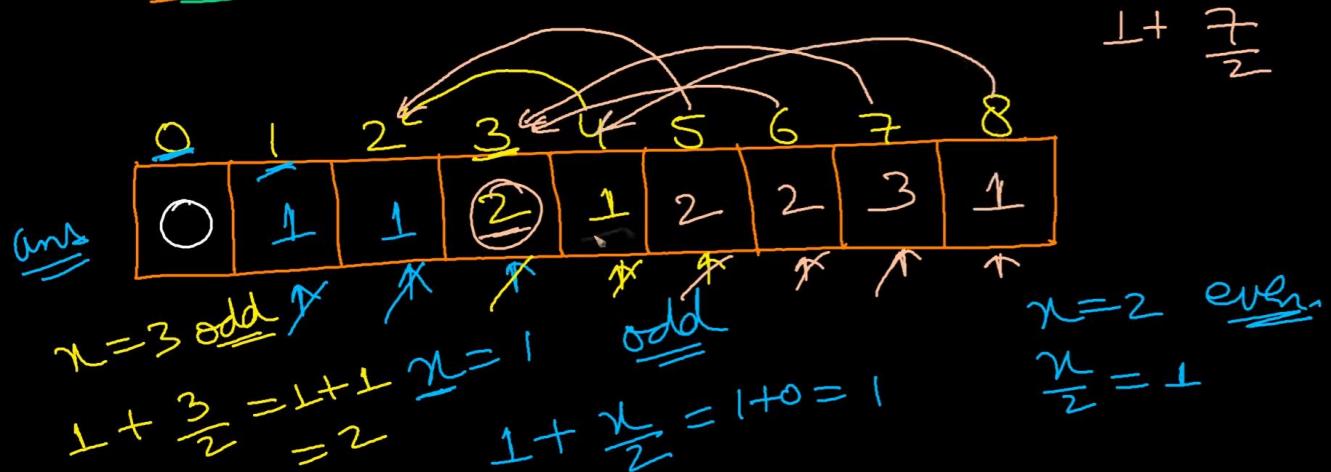
↳ We can compute current set bit count using previous count at $\frac{n}{2}$ in $O(1)$ time.

$$\underline{\text{Num}} = 8$$



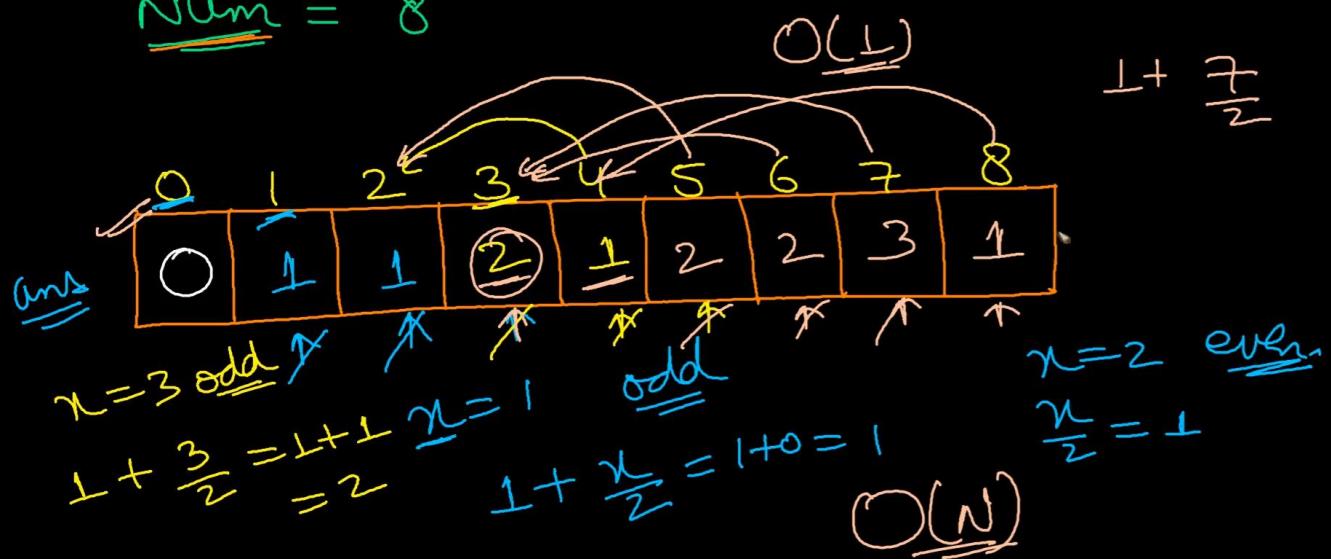
↳ We can compute current set bit count using previous count at $\frac{N}{2}$ in $O(1)$ time.

$$\underline{\text{Num}} = 8$$



↳ We can compute current set bit count using previous count at $\frac{N}{2}$ in $O(1)$ time.

$$\underline{\text{Num}} = 8$$



Explore - LeetCode

https://leetcode.com/explore/challenge/card/may-leetcode-challenge/537/week-4-may-22nd-may-28th/3343/ 150%

LeetCode Explore Problems Mock Contest Articles Store Premium

Go to Discuss

```
1 class Solution {
2     public:
3         vector<int> countBits(int num) {
4             //mem[i] = No of 1s from 0 to number i
5             vector<int> mem(num+1);
6             mem[0] = 0;
7
8             for(int i=1;i<=num;++i)
9                 mem[i] = mem[i/2] + i%2;
10
11         return mem;
12     }
13 };
```

Explore - LeetCode

https://leetcode.com/explore/challenge/card/may-leetcode-challenge/537/week-4-may-22nd-may-28th/3343/ 150%

LeetCode Explore Problems Mock Contest Articles Store Premium

Go to Discuss

```
1 class Solution {
2     public:
3         vector<int> countBits(int num) {
4             //mem[i] = No of 1s from 0 to number i
5             vector<int> mem(num+1);
6             mem[0] = 0;
7
8             for(int i=1;i<=num;++i)
9                 mem[i] = mem[i/2] + i%2;
10
11         return mem;
12     }
13 };
```

O to Num

O

$\textcircled{O} \rightarrow O$

$n = n/2 + 1$