

Q.2 Solve $T(n) = 2T(\sqrt{n}) + \log n$

assume, $n = 2^m \rightarrow$ take log both side

$$\log n = m$$

$$\begin{aligned} T(n) &= T(2^m) = 2T(\sqrt{2^m}) + m \\ &= T(2^m) = 2T(2^{m/2}) + m \end{aligned}$$

assume $\rightarrow \underline{S(m)} = \underline{T(2^m)}$

$$\checkmark S(m) = 2S\left(\frac{m}{2}\right) + m \rightarrow$$

direct
master
theorem apply

$$O(m \log m)$$

$$m = \log n$$

$$\underline{\underline{O(\log n \log(\log n))}}$$

\mathcal{Q}_n

$$T(n) = \underline{\underline{T(\sqrt{n}) + 1}}$$

$$T(n) = T(m) = T(2^{m/2}) + 1$$

$$S(m) = S\left(\frac{m}{2}\right) + 1$$

$$\underline{\underline{O(\log m)}}$$

$$\underline{\underline{O(\log \log n)}}$$

Bit manipulation

observed



q-12



Just 2 mins

↳

operators

Bitwise

logical

Bitwise

operand boolean logic

value - bit

$a > b$

and

$(c < d)$

false and true

$\&$ \rightarrow and \rightarrow
 $|$ \rightarrow or
 \wedge \rightarrow xor

\sim negation

68 \rightarrow 1000100
 34 \rightarrow 100010
 17 \rightarrow 10001
 8 \rightarrow 1000

$$\left(\frac{x}{2} \right) / 2 / 2$$

2^3

\gg \rightarrow right shift

$$x \gg 3$$

steps

division by 8

$$(01001011) \gg 3$$

$$00010010$$

quotient
remainder

$\ll \rightarrow$ left shift

\rightarrow [multiply by 2]

Ex $(01001011) \ll 2$

remove \swarrow \searrow remain

00101100
remain \searrow append

$1 \times (2 \times 2 \times 2 \times 2)$

Qⁿ

Given a value n , calculate

$1 \leq n \rightarrow 2^n$

2^n

$\log n$

Qⁿ You've 2 players A and B, A plays with array 'a', B plays with array 'b'. Both wants to delete their respective arrays completely.

To delete they can do the following multiple times,

choose some ^{subset} numbers from the array, remove all of them. The cost of one operation is bitwise OR of

the chosen no's. You've to minimize the cost for them and the one with greater value of $(\sum a[i] - \text{Total})$ wins. Predict who wins if they play optimally.

Q You've 2 players A and B, A plays with array 'a', B plays with array 'b'. Both wants to delete their respective arrays completely.

To delete they can do the following multiple times,

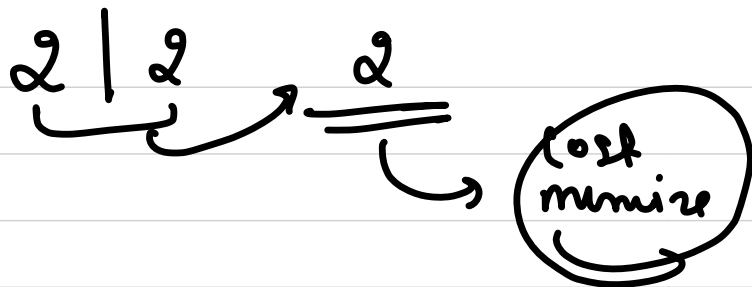
choose some ^{subset} numbers from the array, remove all of them. The cost of one operation is bitwise OR of the chosen no.s. You've to minimize the cost for them and the one with greater value of $(\sum a[i] - \text{Total cost})$ wins. Predict who wins if they play optimally.

$$\begin{aligned} N &\leq 10^5 \\ a_i &\leq 10^5 \\ b_i &\leq 10^5 \end{aligned}$$

$$\boxed{B \quad 3}$$

Ex

$$\begin{aligned} a &\rightarrow \boxed{1, 2, 1} \rightarrow 1 + 2 \rightarrow \underline{\underline{3}} & 4 - 3 &\Rightarrow 1 \\ b &\rightarrow \boxed{1, 2, 3} \rightarrow \underline{\underline{3}} & 6 - 3 &\Rightarrow 3 \end{aligned}$$



$[a_1, a_2, \dots, a_n]$

unique →

$x_1 \rightarrow y_1$

$x_2 \rightarrow y_2$

$x_3 \rightarrow y_3$

\vdots
 $x_i \rightarrow y_i$

$x_i \rightarrow \text{unique}$

$x_1 \leftarrow [x_1, x_1, x_1, \dots, x_1]$

$x_2 \leftarrow [x_2, \dots, x_2]$

\vdots
 \vdots
 \vdots

[total array]

$x_1, x_1, \dots, x_1, x_2, x_2, \dots, x_2$

$x_3, x_3, \dots, x_3,$

→ $(x_1 + x_2 + x_3 + \dots + x_n)$

$$\boxed{1, 2, \dots, 1}$$

$1, 1, 1, \dots, 1, 2, 2, \dots, 2, \dots$

unordered set \rightarrow $\boxed{1}$
 $x | x \rightarrow \boxed{x}$

$$\text{cost} \rightarrow \boxed{1 | 2} + \boxed{1}$$

$$\rightarrow \boxed{1 | 1} + (2)$$

$\boxed{1} \leftarrow \{1, 1, \dots, 1\}$ **subset**

$$\rightarrow \text{cost} \rightarrow \boxed{1 | 1 | 1 \dots | 1} + \boxed{2 | 2 | \dots | 2} + \{1, 2, \dots, 2\} \rightarrow \boxed{2}$$

$$\boxed{3 | 3 \dots 3} \dots \dots \{3, 2, \dots, 3\} \rightarrow \boxed{3}$$

$$\boxed{1 + 2 + 3 \dots}$$

Q.7 Check if the k^{th} bit is set or not?

$n = 01001011$

$k = 4$

$((n \gg k) \& 1) \rightarrow 0 \rightarrow \text{not set}$
 $\begin{array}{r} 101 \\ 001 \end{array} \gg 1$
 $\rightarrow 1 \rightarrow \text{set}$

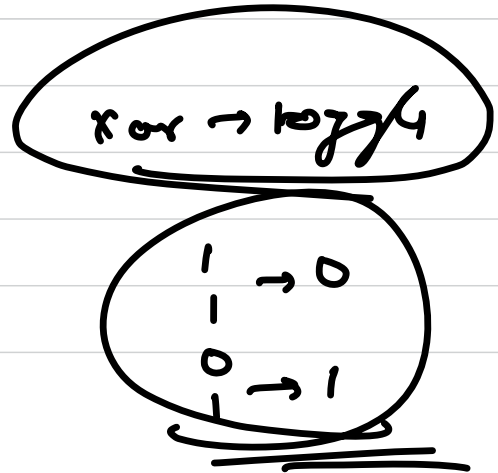
$n \& (1 \ll k) \rightarrow 0 \rightarrow \text{not set}$
 $\text{else} \rightarrow \text{set}$

Q₁ → set the ith bit //

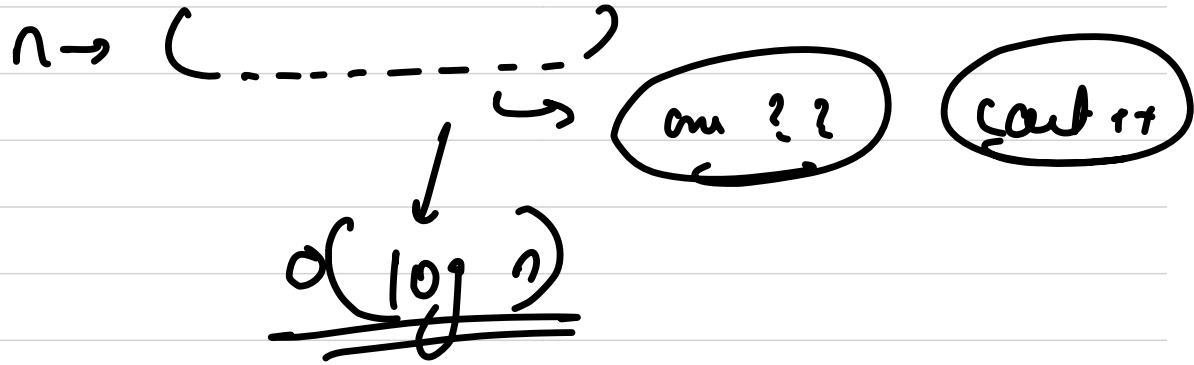
$$n = n | (1 \ll i)$$

Q₂ → toggle the kth bit

$$n = n ^ (1 \ll k)$$



Qⁿ Given a no. n , count its set bits.



Brian
Kernighan
algo

The relation of n and $n-1$

in $(n-1)$ we have ^{almost} all the bits of n same
except the rightmost 1 in n , and all the
bits to the right of rightmost 1.

5 set bit

$\frac{n-1}{2}$ → rightmost bit and every bit to the right is flipped.

→ $(n \& n-1)$

5 run

$O(\# \text{ of set bit})$

1 set bit neutralizes → unset

134 → 10000100
& 133 → 10000101

10000100 → cancel out