

Maximum XOR

You are given two arrays arr1 & arr2. Your task is to find max value of $A \oplus B$, where A & B are any elements from arr1 & arr2.

Solving an easier problem

Ques- Given an array of numbers, & a number x. find the max value of $(arr[i] \oplus x)$.

① → Insert all the numbers into the tree. (binary bits)

② → Take x & find the max no. from array where $(no \oplus x) \uparrow$.

Node's $0 \rightarrow 0$

link C2D; $1 \rightarrow 1$

3

array → [9, 8, 7, 5, 4]

01001 → 00111 → 00101
01000 → 00100

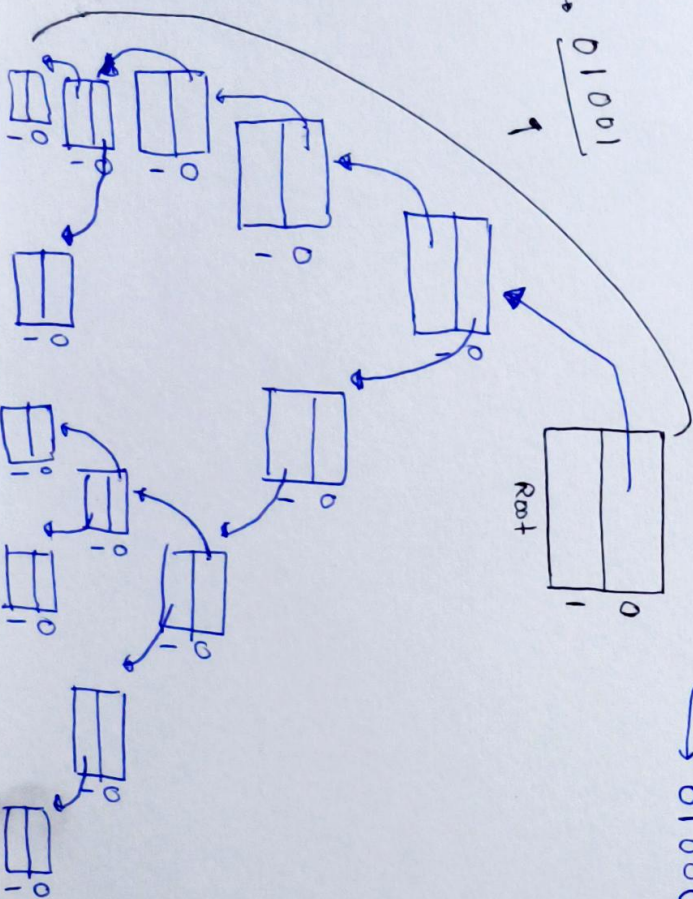
→ Insert

9 → 01001

1

Root

Start inserting from extreme left (MSB)



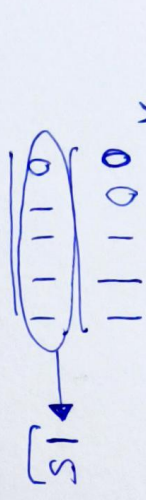
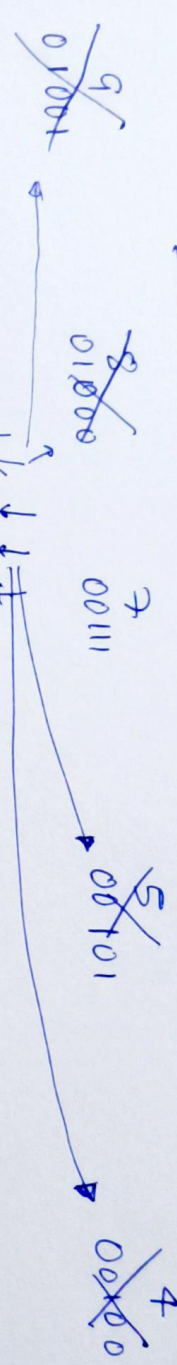
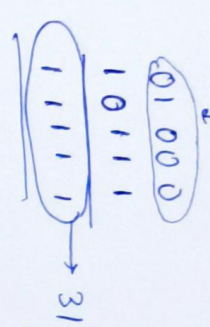
Maximizing XOR

$x = 8$

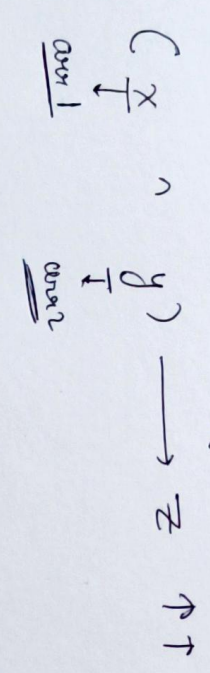
Intuition \rightarrow If both no. have different bits, especially the resultant XOR is supposed to be maximum.

at MSB's then

wanted



Now let's talk about big problem that we were solving.



ans1 \rightarrow present in tree.

$x \rightarrow$ will be all elements from ans2.

$x \in \text{ans2}$

root
present();
get-max(x); \rightarrow calculate max xor with all elements of tree.

\rightarrow ans1 insertion in tree.

$T.C \rightarrow O(N \times 32)$

$+ O(M \times 32)$

going over all elements in ans2 & finding max-xor