

Welcome 😊

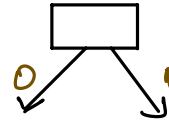
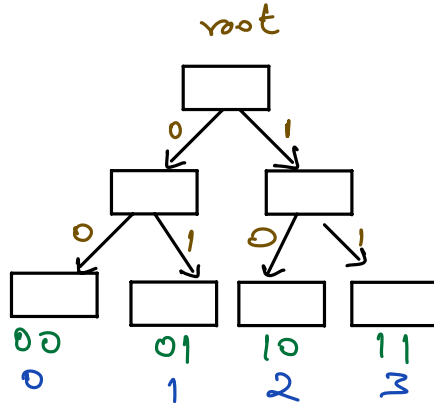
Agenda: Tries on bits

2 ques's

2 general probs on trees

Tries of Bits

MSB
↓
LSB



Numbers \Rightarrow 0 to 15
height = 4 $2^4 - 1$

Numbers \Rightarrow 0 to 31
height = 5

Numbers \Rightarrow 0 to $2^H - 1$

Q Given an integer array A, find the max. value of $A[i] \wedge A[j]$ $\forall (i, j)$ pairs.

eg: $A = [\overset{0}{3} \quad \overset{1}{5} \quad \overset{2}{2}]$

1) $3 \wedge 5 \Rightarrow$

0	1	1
1	0	1
<hr/>		
1	1	0

= 6

2) $3 \wedge 2$

0	1	1
0	1	0
<hr/>		
0	0	1
<hr/>		
11		
= 1		

3) $5 \wedge 2 \Rightarrow$

1	0	1
0	1	0
<hr/>		
1	1	1

= 7 \Rightarrow Ans

Bruteforce

$\nabla_{1,j}$ check

$$AC[i] \wedge AC[j] \text{ \& calc. mon.}$$

T.C $\Rightarrow O(N^2)$

$$S.L \Rightarrow O(1)$$

$$15^7$$

msb

1 1 1 1

0 1 1 1

$$\overline{1000} \rightarrow 8$$

$$>$$

15^n 11

msb

1 1 1 1

1 0 1

0 1 0 0 → 4

MSB

10000

msb

0 1 1 1 1

\Rightarrow To maximize the number, MSB should be set to 1

⇒ Travelling bits from MSB to SB → Tries of Bits

eg: $A [20, 30, 15, 25, 10]$ $K = 5$

ans = ~~1027~~
30

20 \Rightarrow 10100

$$30 \Rightarrow 11110_{20} \Rightarrow 01010 = 10$$

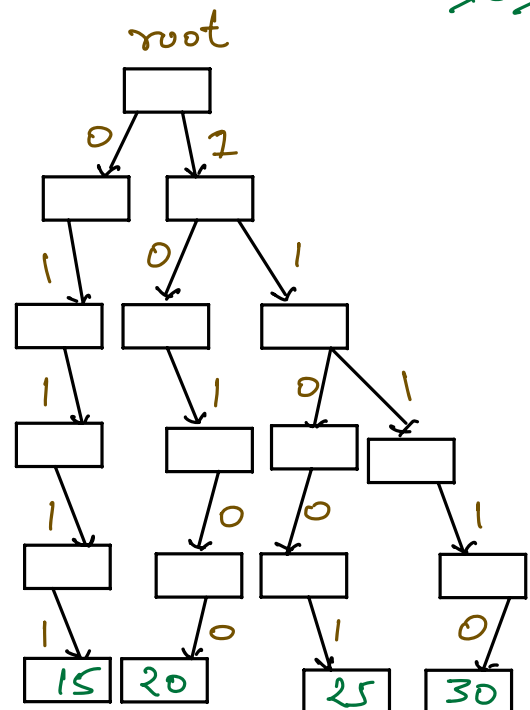
$$15 \Rightarrow 01111^{\wedge} 20 \Rightarrow 11011 = 27$$

$$25 \Rightarrow 11001^{\wedge}15 \Rightarrow 10110 = 22$$

$$10 \Rightarrow 0 \mid 0 \mid 0^20 \Rightarrow 1 \mid 1 \mid 1 \mid 0 = 30$$

T.C $\rightarrow O(N)$

S.C $\rightarrow O(N \times 30) \approx O(N)$



$$A[i] \leq 10^9 \leq 2^{30}$$

Steps

- 1) Find max XOR for $A[i]$ \rightarrow except 1st number.
- 2) Insert $A[i]$

Code

```
1) int find (root, x)
{
    temp = root
    for (i  $\rightarrow$  29 to 0)
    {
        b = (x >> i) & 1 // ith bit of X
        t = 1 - b // t = b^1
        if (temp.child[t]) temp = temp.child[t]
        else temp = temp.child[b]
    }
    return x ^ temp.data
}
```

Q Given an integer array A, find subarray with max XOR value.

eg: A : [4, 6, 1]

eg \Rightarrow 4 \rightarrow 4

4 ^ 6 \rightarrow 2

4 ^ 6 ^ 1 \rightarrow 3

6 \rightarrow 6

6 ^ 1 \rightarrow 7 \leftarrow Ans.

1 \rightarrow 1

Brute force

For every subarray, calc. XOR and store max.

T.C $\Rightarrow O(N^3) \rightarrow O(N^2)$

S.C $\Rightarrow O(1)$

$$a \wedge a \Rightarrow 0$$

$$a \wedge b \wedge a \Rightarrow b$$

Prefix Sum
Prefix XOR

$$\text{Subarray XOR } (i \rightarrow j) \Rightarrow A[i] \wedge A[i+1] \dots \wedge A[j]$$

$$\begin{aligned} \text{Subarray XOR } (i \rightarrow j) &\Rightarrow P[j] \wedge P[i-1], i > 0 \\ &\rightarrow P[j], i = 0 \end{aligned}$$

Ans \rightarrow max (max $\forall i$ $P[i]$, max XOR pair in $P[]$)

$$\begin{array}{ccc} P \text{ XOR } & 4 & 6 & 1 \\ & [4, 4 \wedge 6, 4 \wedge 6 \wedge 1] \end{array}$$

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

$$S.C \rightarrow O(N)$$

Q Given a BST where exactly 2 nodes are swapped, find the two nodes. (distinct values)

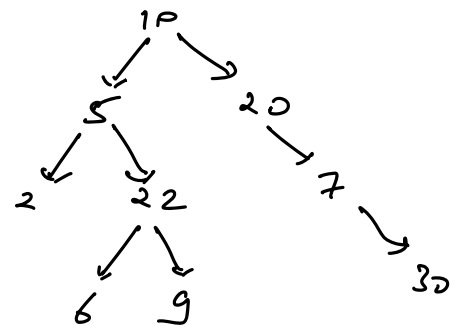
LNR

Inorder traversal is sorted.

2 5 6 22 9 10 20 7 30

↓ first no. of first pair

↓ 2nd no. of 2nd pair

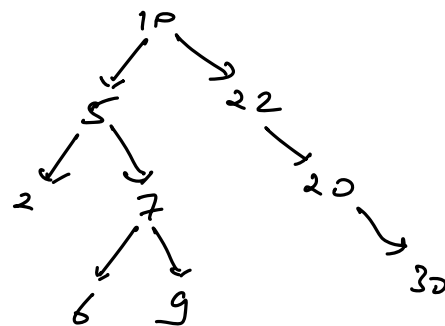


2 5 6 7 9 10 22 20 30
↓ ans

If only 1 mismatch

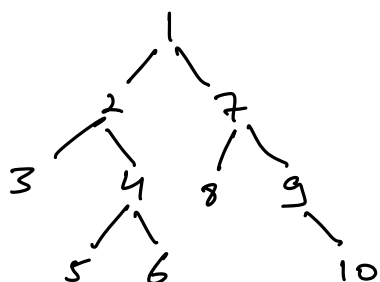
T.C $\Rightarrow O(N)$

S.C $\Rightarrow O(H) \xrightarrow{\text{ Morris }} O(1)$



Q Flatten the given binary tree to linked list in preorder manner s.t right child will become next & left child will be NULL

NLR



{ Head, Tail }

pair flatten (root)

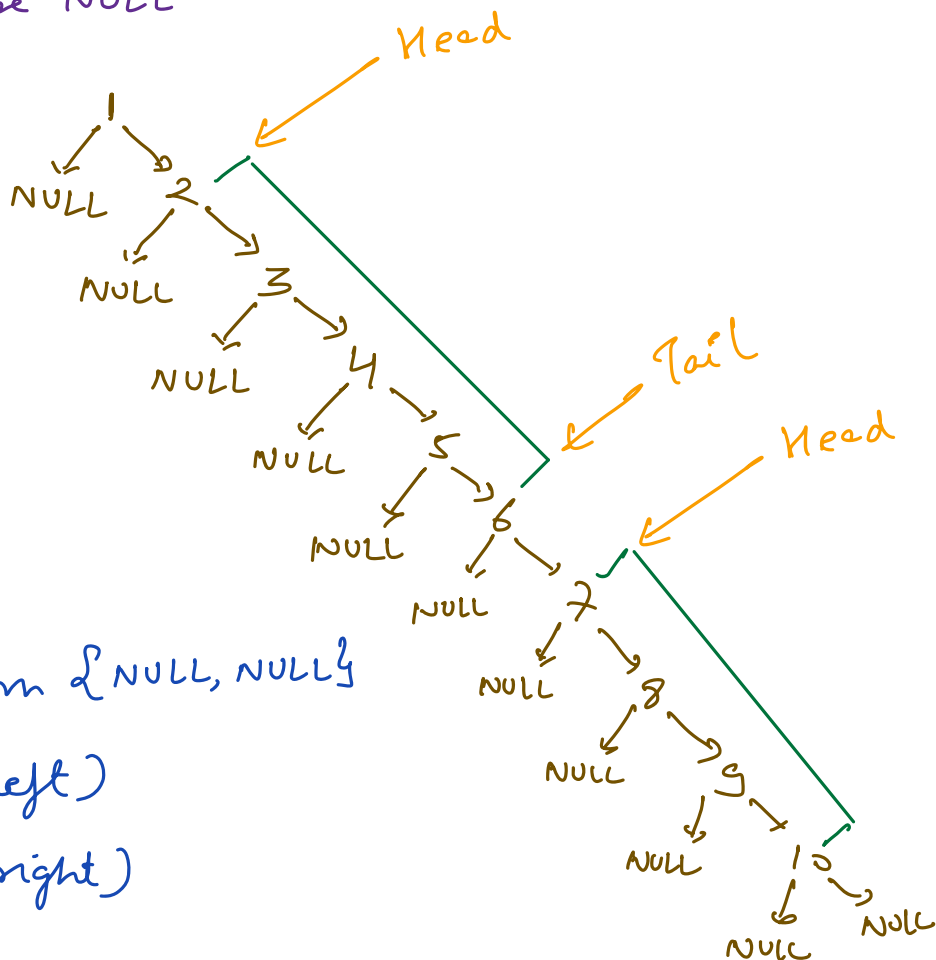
{ if (root == NULL) return { NULL, NULL }

L = flatten (root.left)

R = flatten (root.right)

root.left = NULL

if (L.Head == NULL && R.Head == NULL)
 return { root, root }



```

    else if ( L.Head == NULL ) {
        root.right = R.Head
        return { root, R.Tail }
    }
    else if ( R.Head == NULL ) {
        root.right = L.Head
        return { root, L.Tail }
    }
    else {
        root.right = L.Head,
        L.Tail.right = R.Head
        return { root, R.Tail }
    }
}
}

```

T.C $\rightarrow O(N)$

S.C $\rightarrow O(1)$