Given an array of no. & x

find the max. value of arr[i] ^ x ?

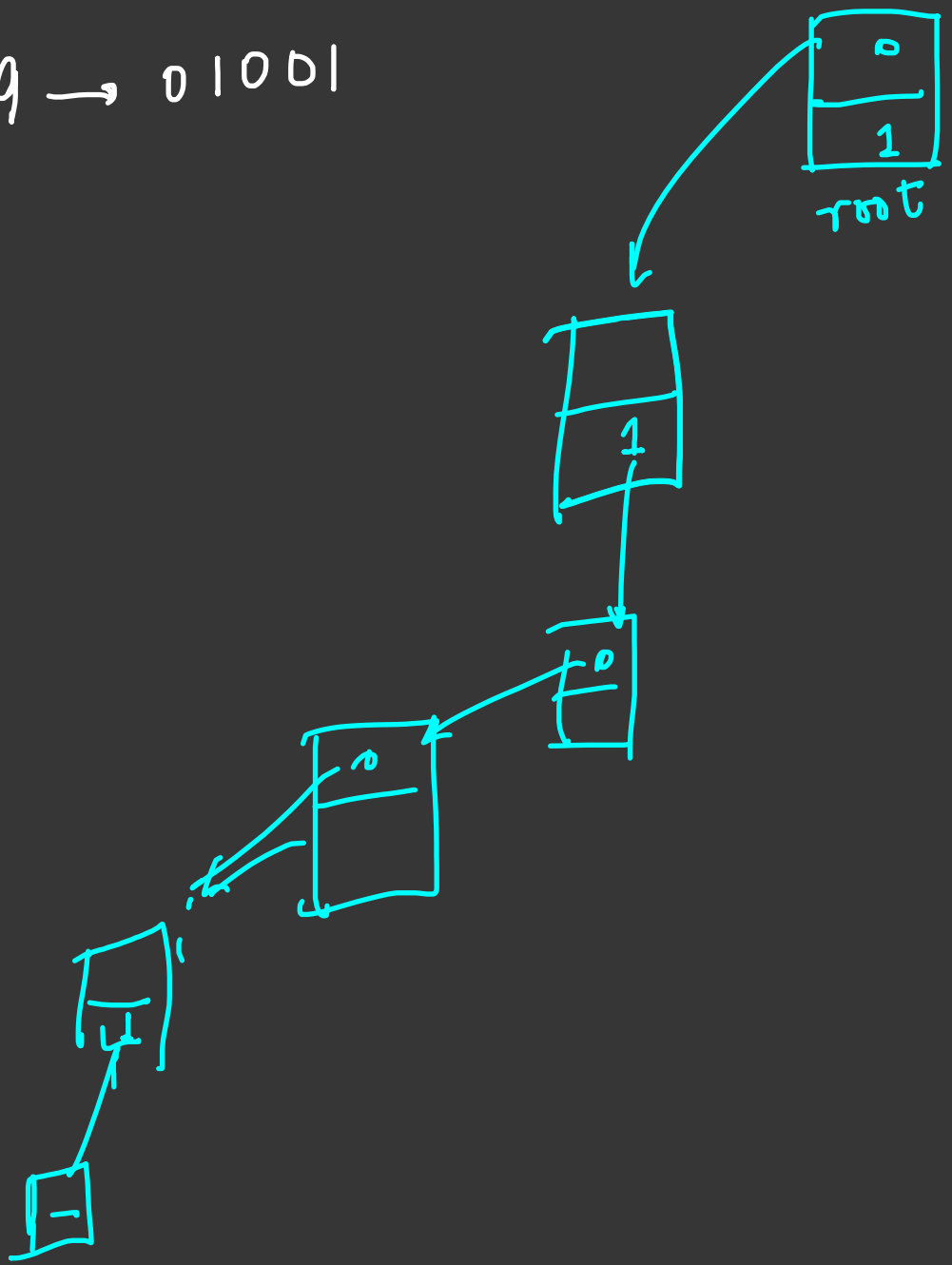
→ Insert all the no. into the trie

→ Binary bits

→ Take x & find the max no. from array ~~val~~ whose (no ^ x) ↑↑

Storing the no.

9 → 01001




```
Node {
    Link[32]
}
```

{9, 8, 7, 5, 4}

9 → 01001

Problem Statement Suggest Edit

You are given an array/list 'ARR' consisting of 'N' non-negative integers. You are also given a list 'QUERIES' consisting of 'M' queries, where the 'i-th' query is a list/array of two non-negative integers 'Xi', 'Ai', i.e 'QUERIES[i]' = ['Xi', 'Ai'].
The answer to the ith query, i.e 'QUERIES[i]' is the maximum bitwise xor value of 'Xi' with any integer less than or equal to 'Ai' in 'ARR'.
You should return an array/list consisting of 'N' integers where the 'i-th' integer is the answer of 'QUERIES[i]'.

arr1 = { 6, 8 }

arr2 = { 7, 8, 2 }

6 ^ 7

6 ^ 8

6 ^ 2

8 ^ 2

Brute force: O(N×M)

{6, 8} , {7, 8, 2}

^

O(N×M)

XOR = 1 ^ 1 = 0

1 ^ 0 = 1

2

Maximum

0 1 0

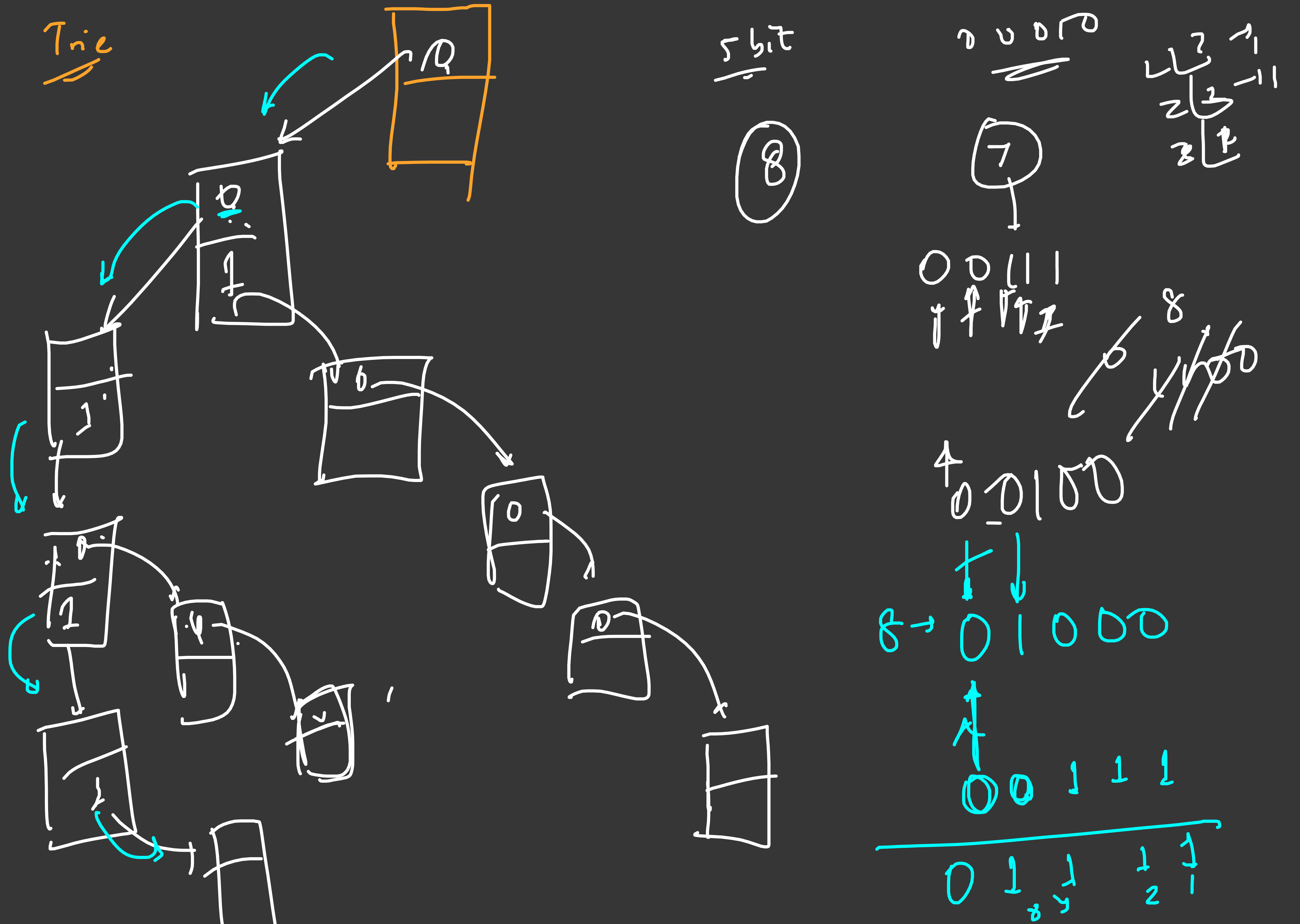
1 . .

32 bit

1 0

1 . . . .

{7, 8, 4} ^ X = 8


Trie
5 bit
8
7
00111
4
00100
8 → 01000
00111

- Insert the all the no. in array in binary form.

- Take one element and try to maximise it with the rest.

Structure of Node Trie

```
Node {
    link[2];
}
```

	And	OR
1 1	1	1
0 1	0	1
1 0	0	1
0 0	0	0

Check Bit → 0 1 0 0 0

num
8 >> 2

8 >> 3

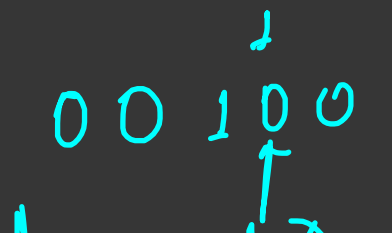
0 0 0 0 1
& 2^0
(1)

0 1 0 0 0
0 0 1 0 0
0 0 0 1 0
0 0 0 0 1
0 0 0 0 1

1 & 2^0

Set bit

Set bit



1 << i

10

num >> i & 1

1 << i & num