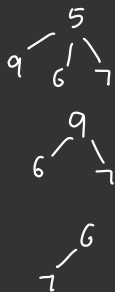


## Advanced Bitmasking Problems

→ Med - Hard problems

Q

Given an array, you find out sum of XOR of all pairs in the array.



arr[] = {5, 9, 6, 7}

sum = 0

$$\Rightarrow 5 \wedge 9 = 12$$

$$\Rightarrow 5 \wedge 6 = 3$$

$$\Rightarrow 5 \wedge 7 = 2$$

$$\Rightarrow 9 \wedge 6 = 15$$

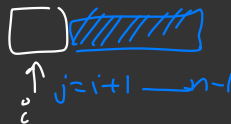
$$\Rightarrow 9 \wedge 7 = 14$$

$$\Rightarrow 6 \wedge 7 = 1$$

47

$$\begin{array}{r} 1001 \\ 0110 \\ \hline 1111 \end{array}$$

$$\begin{array}{r} 110 \\ 111 \\ \hline 001 \end{array}$$



Brute Force

```
sum = 0
for (i = 0 ——— i <= n-2) {
    for (j = i+1 ——— j <= n-1) {
        sum += (a[i] ^ a[j]);
    }
}
```

$O(N^2)$  time

$O(1)$  space

3  
}

sum = 47

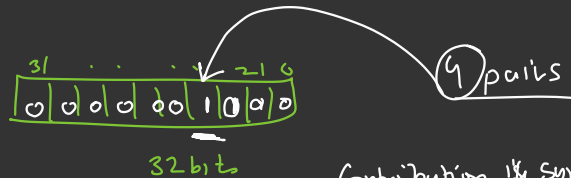
Optimised Approach

Hint-1

$$\boxed{1} \wedge \boxed{0} = \boxed{1}$$

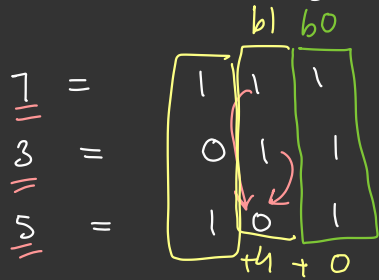
Hint-2

can you count how many pairs will contribute a 1 to ith bit position



$$\begin{aligned} \text{Contribution by sum} &= 8 \times 4 \\ &= \boxed{32} \end{aligned}$$

$$arr = \{7, 3, 5\}$$



BF logic

$$\begin{cases}
 7 \wedge 3 = 100 = 4 \\
 7 \wedge 5 = \boxed{010} = 2 \\
 3 \wedge 5 = \boxed{110} = 6
 \end{cases}$$

$\frac{4+2+6}{2} = \underline{12} \checkmark$

$$7 \wedge 5 + 3 \wedge 5, \underline{7 \wedge 3} + \underline{3 \wedge 5}$$

how many pairs will contribute

$$cnt = 2 \times 1 + 2 \times 1 + 3 \times 0 = 0 = \boxed{12}$$

$$\begin{array}{ccc}
 \underline{1} & \underline{1} & \underline{0} \\
 1 & 1 & 1
 \end{array}$$

$$2 \times 4 = \boxed{8}$$

2 pairs

$$2 \times 2$$

$$= \boxed{4}$$

$$cnt \text{ 1's } \times cnt \text{ 0's}$$

$$\begin{array}{|c|} \hline 1 \\ \hline \end{array} \wedge \begin{array}{|c|} \hline 0 \\ \hline \end{array} = 1$$

3 ways 2 ways =

$$= \underline{\underline{6 \text{ ways}}}$$

$$3 \text{ no's } \left\{ \begin{array}{c} 1 \\ 1 \\ 1 \end{array} \right\} \quad 2 \text{ no's } \left\{ \begin{array}{c} 0 \\ 0 \end{array} \right\}$$

$$\underline{\underline{a \wedge b = 1}}$$

$$\begin{array}{|c|} \hline n1 \\ \hline n3 \\ \hline n5 \\ \hline \end{array} \quad \begin{array}{|c|} \hline n2 \\ \hline n4 \\ \hline \end{array}$$

1

n1	1	---	1
n2	0	---	0
n3	---	---	1
n4	---	---	0
n5	---	---	1

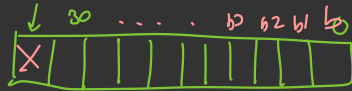
--- ---  $\frac{1}{2}$   
 ↑  
 6 pairs which will contribute one.

$$\begin{array}{rcl}
 & \textcircled{9} & 2 \quad 1 \\
 & b_3 & b_2 \quad b_1 \\
 7 & = & \begin{array}{|c|c|c|} \hline \textcircled{1} & \textcircled{1} & 1 \\ \hline \end{array} \\
 3 & = & \begin{array}{|c|c|c|} \hline \textcircled{0} & \textcircled{1} & 1 \\ \hline \end{array} \\
 5 & = & \begin{array}{|c|c|c|} \hline \textcircled{1} & 0 & 1 \\ \hline \end{array}
 \end{array}$$

$$\begin{array}{l}
 2 \times 1 \times 4 \\
 = 8 + 4 \\
 = \boxed{12}
 \end{array}
 \quad
 \begin{array}{l}
 2 \times 1 \times 2 \\
 = 4 \\
 \boxed{4}
 \end{array}
 \quad
 \begin{array}{l}
 0, 0, 0 \\
 \hline
 \textcircled{0}
 \end{array}$$

$$\begin{array}{l}
 \frac{7 \wedge 3}{0} + \frac{3 \wedge 5}{0} + \frac{7 \wedge 5}{0} \\
 = \cancel{7 \wedge 3} + \textcircled{3 \wedge 5} + \textcircled{7 \wedge 5} \\
 \quad \downarrow \quad \downarrow \quad \downarrow \downarrow \\
 \quad 1, 0 \quad 1, 0 \\
 = \underline{10} + \underline{10} \\
 = 2 \times 2 \\
 = \textcircled{4}
 \end{array}$$

$$\begin{array}{r}
 8 \quad 4 \quad 2 \quad 1 \\
 - \frac{1}{11} - \\
 2
 \end{array}$$



31 bits

Sum = 0

for (i = 0; i <= 31; i++) {

A = 0, B = 0 // cnt 1's

for (x : arr) {

if (x & (1 << i) > 0) {

A++, mask

else {  
B++;

}

Contribution = A \* B \* (1 << i);

Sum = Sum + Contribution;

CheckBit(no, i) {

return (no & (1 << i)) > 0;

}

i = 2

3  
5  
7

b1	b0
0	1
1	0
1	1

A0 B0  
3 0

2 1

2, 1

3 & 4 = 0  
5 & 4 = 0  
7 & 4 = 0

1 << 2 = 100

1 << 0 = 1

3 & 1 > 0  
5 & 1 > 0  
7 & 1 > 0

3 \* 0 \* (2<sup>0</sup>) = 0

2 \* 1 \* (2<sup>1</sup>) = 4

2 \* 1 \* 2<sup>2</sup> = 8

12

1 & 1 = 1  
1 & 0 = 0  
0 & 1 = 0  
0 & 0 = 0

ih bit  
3 -- ①  
5 -- ①  
7 -- ①  
↑ 21  
ih bit

iterate

on bits of many nos

O(N)

Time

3

$$\begin{array}{r}
 2^2 \quad 2^1 \quad 2^0 \\
 - \quad \textcircled{1} \quad - \quad - \\
 - \quad \textcircled{1} \quad - \quad - \\
 - \quad \textcircled{1} \quad - \quad -
 \end{array}$$

$$\begin{aligned}
 &= 3 \times 2^2 \\
 &= \boxed{12}
 \end{aligned}$$

$$1 \otimes *$$

$$\begin{aligned}
 5 > 2 \\
 \uparrow
 \end{aligned}$$

$$\begin{array}{r}
 101 \\
 \leftarrow 100 \\
 10) \\
 100 \\
 \hline
 \hline
 \hline
 \end{array}$$

↓  
 pick one of  
 bits  
 as 1  
 $\boxed{A}$

↓  
 as  
 0  
 $\boxed{B}$

$$\begin{aligned}
 &A_{C_1} \times B_{C_1} \\
 &= \underline{A \times B}
 \end{aligned}$$

$$\begin{aligned}
 &1=2 \quad i=1 \quad i=10 \\
 5 &= \rightarrow \boxed{1} 0 1 \\
 &\quad \downarrow \quad \downarrow \\
 &\quad 1 \quad 0 \quad 0 \\
 &\quad \boxed{1} \quad 0 \quad 0 \quad > 0
 \end{aligned}$$

mask  
 =  $k \leq 2$

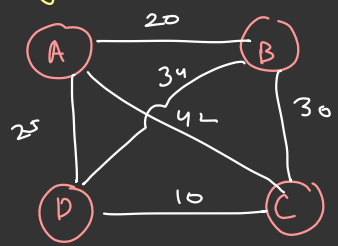
Additional challenge →

# Travelling Salesman Problem

→ N cities, given distance b/w all pair of cities

visualise

Network of nodes  
↓  
graph



Input

Adj Matrix

	A	B	C	D
A	0	20	42	25
B	20	0	30	34
C	42	30	0	10
D	25	34	10	0

Make a tour starting from A and come back to A by travelling all cities in min distance

Tour →  $A \rightarrow B \rightarrow C \rightarrow D \rightarrow A = 20 + 30 + 10 + 25 = 85$

$A \rightarrow D \rightarrow B \rightarrow C \rightarrow A = 25 + 34 + 30 + 42 = 131$

$A \rightarrow C \rightarrow D \rightarrow B \rightarrow A = 42 + 10 + 34 + 20 = 106$

✓  
85  
✓  
131  
106  
min

# no of towers

A

B, C, D  
— — —

A

$3! = 6$  ways

B, C, D

B, D, C

C, B, D

C, D, B

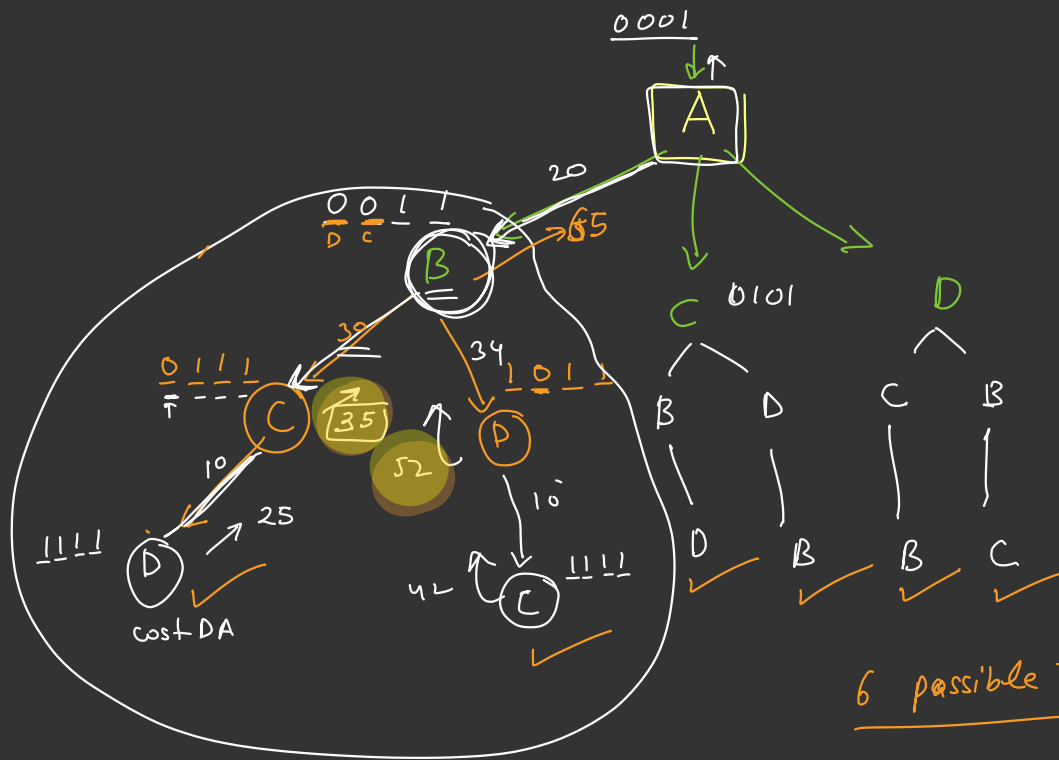
D, B, C

D, C, B

6 ways

Tower with  
Min  
Cost

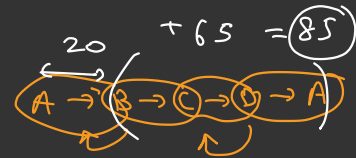




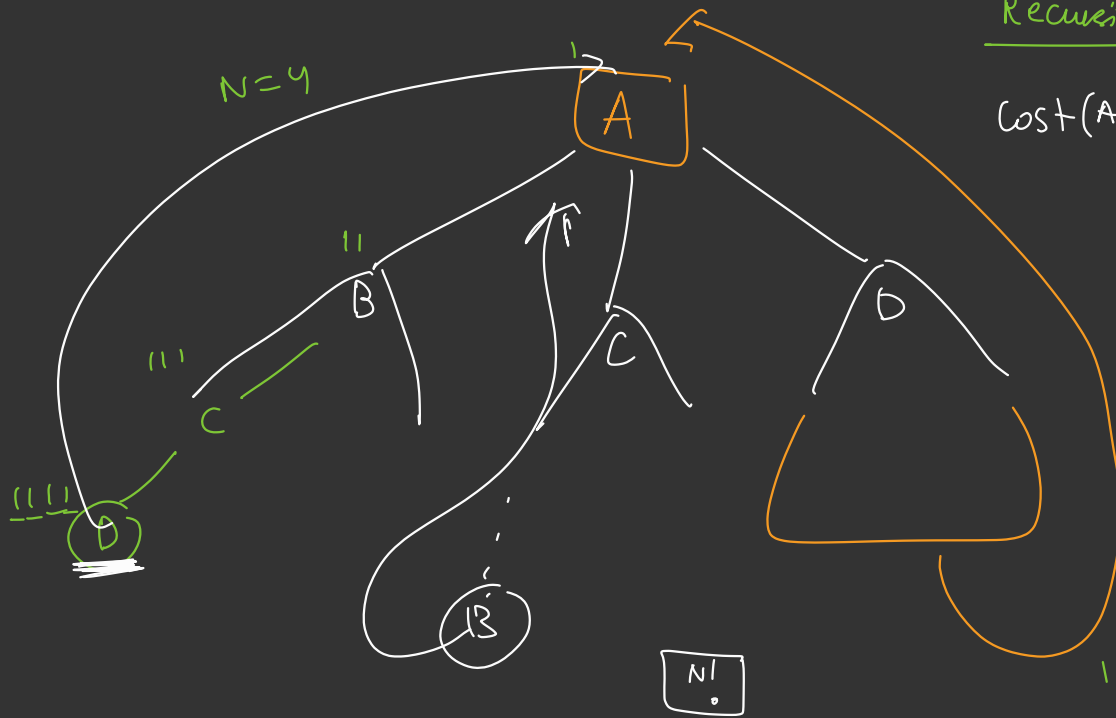
visited

0	0	0	1
D	C	B	A

	A	B	C	D
A	0	20	42	25
B	20	0	30	34
C	42	30	0	10
D	25	34	10	0



A → B → D → C → A



Exponential

## Recursive Case

$$\text{Cost}(A) = \min \left( \begin{array}{l} \text{Cost from all} \\ \text{unvisited cities} \end{array} \right)$$

+ Cost To Reach  
that city

$$= \min \left\{ \begin{array}{l} \text{Cost}(B) + A-B \\ \text{Cost}(C) + A-C \\ \text{Cost}(D) + A-D \end{array} \right.$$

Base Case → if all cities are  
visited

return ~~cost~~ minCost[current][A]



Break . 12 min (10:30)

Challenge → Max AND Pair

n elements, find a pair of element that generates a maximum AND value.

{ 4, 8, 12, 16 }

(4)	<u>0</u>	<u>0</u>	<u>1</u>	<u>0</u>	<u>0</u>
(8)	<u>0</u>	<u>1</u>	<u>0</u>	<u>0</u>	<u>0</u>
(12)	<u>0</u>	<u>1</u>	<u>1</u>	<u>0</u>	<u>0</u>
(16)	<u>1</u>	<u>0</u>	<u>0</u>	<u>0</u>	<u>0</u>

= (8)

$$8 \& 12 = (8)$$

Brute Force

Two loops

$O(n^2)$

ac78 ac7

(j > i)



optimisation

Max AND  $\rightarrow$  Move sets on the bit

left side

0 1 1 1 1

1 0 0 0  
= 8

hint

Condition



$pat \& no == pat$

0 1 0 0 0  
set bit  $\uparrow$   
Smaller  
0 0 1 0 0  
Bigger  
0 1 1 0 0

pat

1 0 1 0 1  
1 0 0 0 0  
cnt 1

Not possible

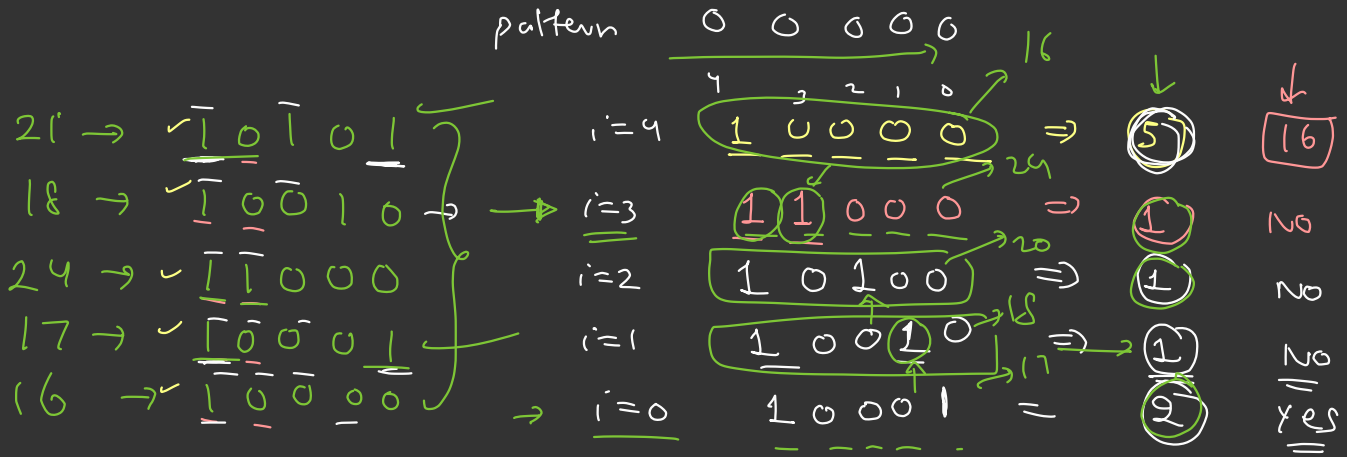
0 1 0 0 0  
8  
2

0 1 1 0 0  
1

0 1 0 1 0  
0

0 1 0 0 1  
0

9 0 0 1 0 0 x x  
8 0 1 0 0 0 ✓ x  
12 1 1 0 0 0 ✓ ✓  
16 1 0 0 0 0 x x



```
int pattern = 0;
for(int i=31; i>=0; i--){
    //count the no of numbers having given pattern
    int count = countNumbers(arr, pattern | (1<<i));
    if(count>=2){
        //actually update the pattern by setting ith bit
        pattern = (pattern | (1<<i));
        System.out.println(pattern);
    }
}
```

Set Bits  
in atleast  
2 No's at  
given i

