3rd Session on Graph Interview Problems

Medium

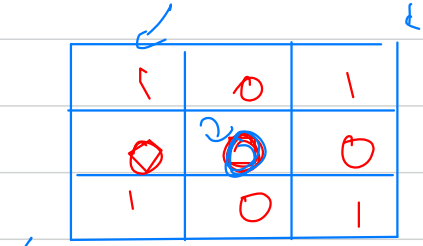
Q.1 You have a $n \times n$ grid of binary value (0, 1) find the maximum distance of a '0' value (manhattan distance) nearest '1' value.

1	0	0
0	0	0
0	0	0

Ans

$n \leq 10^2$

0 → water
1 → land



ans → 2

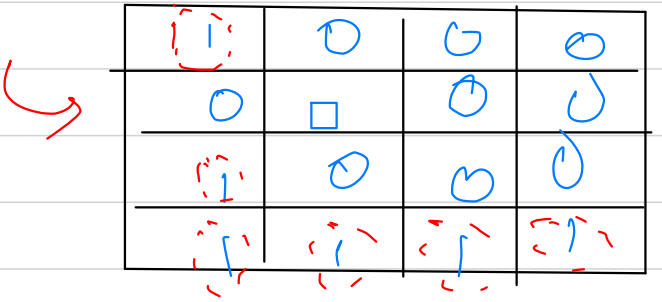
(1,1) → 2

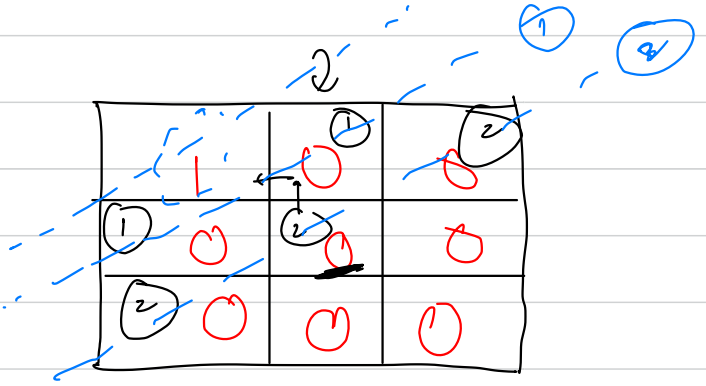
What is Manhattan distance ??

(x_1, y_1) (x_2, y_2)

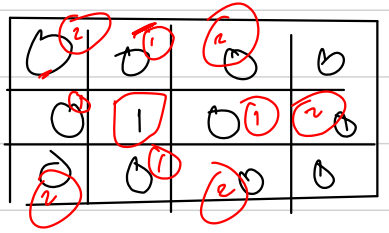
$|x_1 - x_2| + |y_1 - y_2|$

→ Manhattan distance



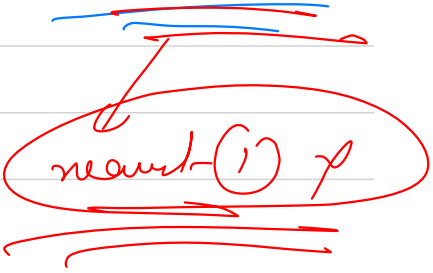


level based increment

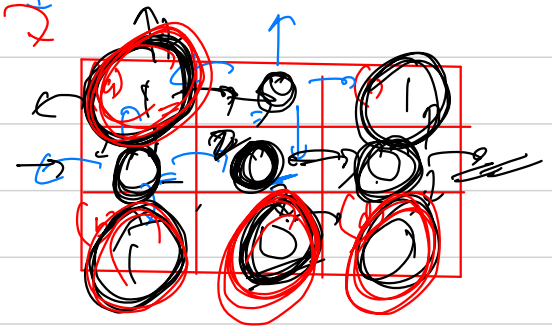


if we one by one consider all the ones & for each '1' value go level by level for zeroes, then we zero with min level will be a candidate for ans.

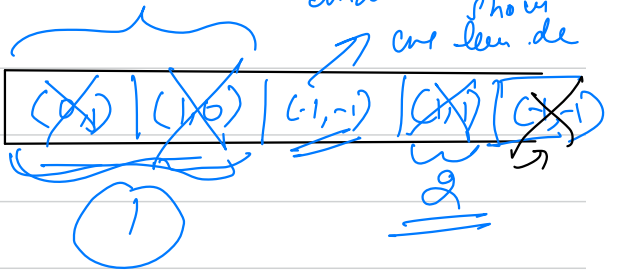
(BFS)

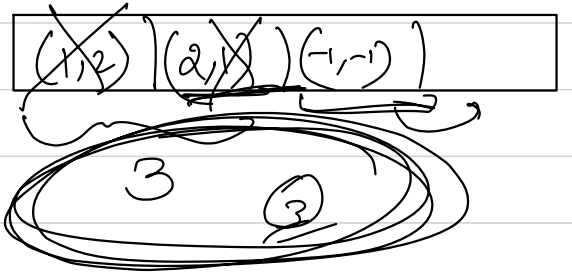


BFS

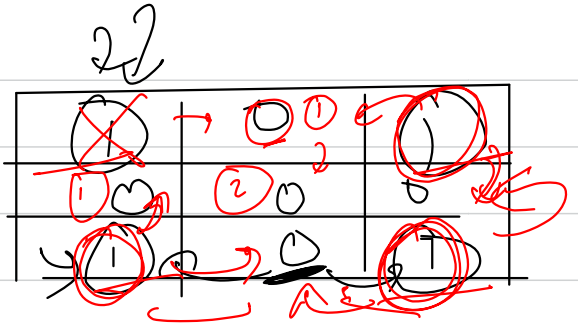


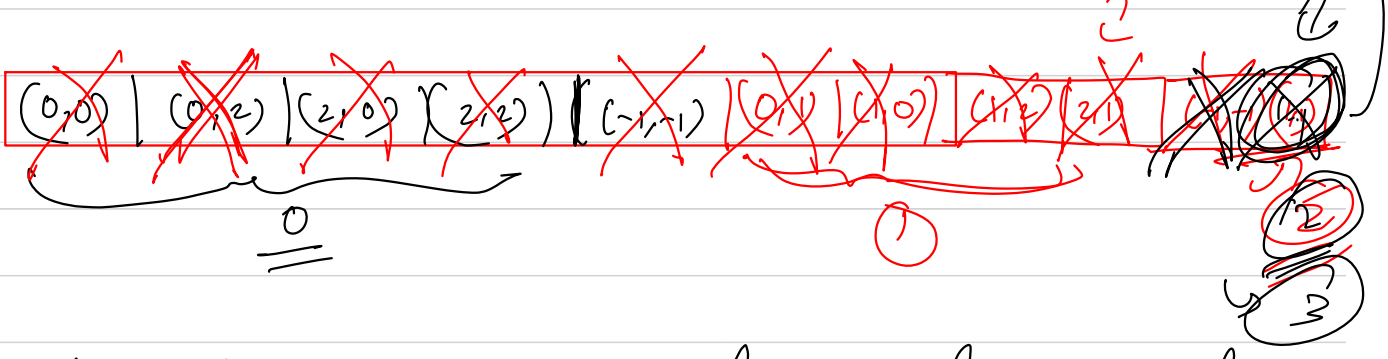
indicates that shown cur level de





Simultaneously





from all the ones, we will similarly do BFS

Hard → medium-hard

$N \leq 10^2$

N couples (2N people) - All the persons are arranged in some random permutation in a linear row.

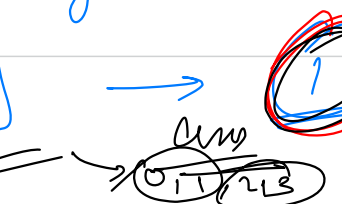
Now the couple want to hold the hands & they have sit adjacent for doing this.

$(2i-2, 2i-1)$

$(0,1)$ $(2,3)$ $(4,5)$ —— - - - $(2N-2, 2N-1)$

find min swaps to adjust every couple correspondly to their partner

eg $[0, 2, 1, 3]$ → 1



2, 4, 5 → 1 component → n' → n'-1

6 → 1 component

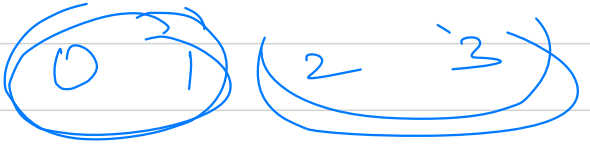
5, 3 → 1 component

(0, 1)

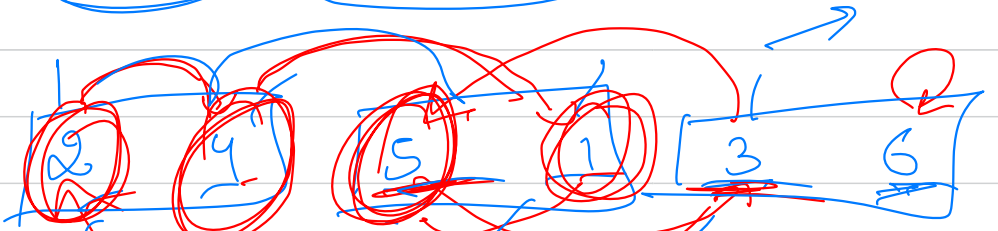
(1, 0)

1

graph



→ Min swaps to make array sorted



2, 4, 1

5, 3

6

2, 4

1

2

3

2

1

6

n^1 → ($n^1 - 1$ swaps) → DSU

Sort

Total N nodes →

K Components → C_1, C_2, C_3, C_4 ------ (C_k)

n_1 n_2 n_3 n_4 n_k

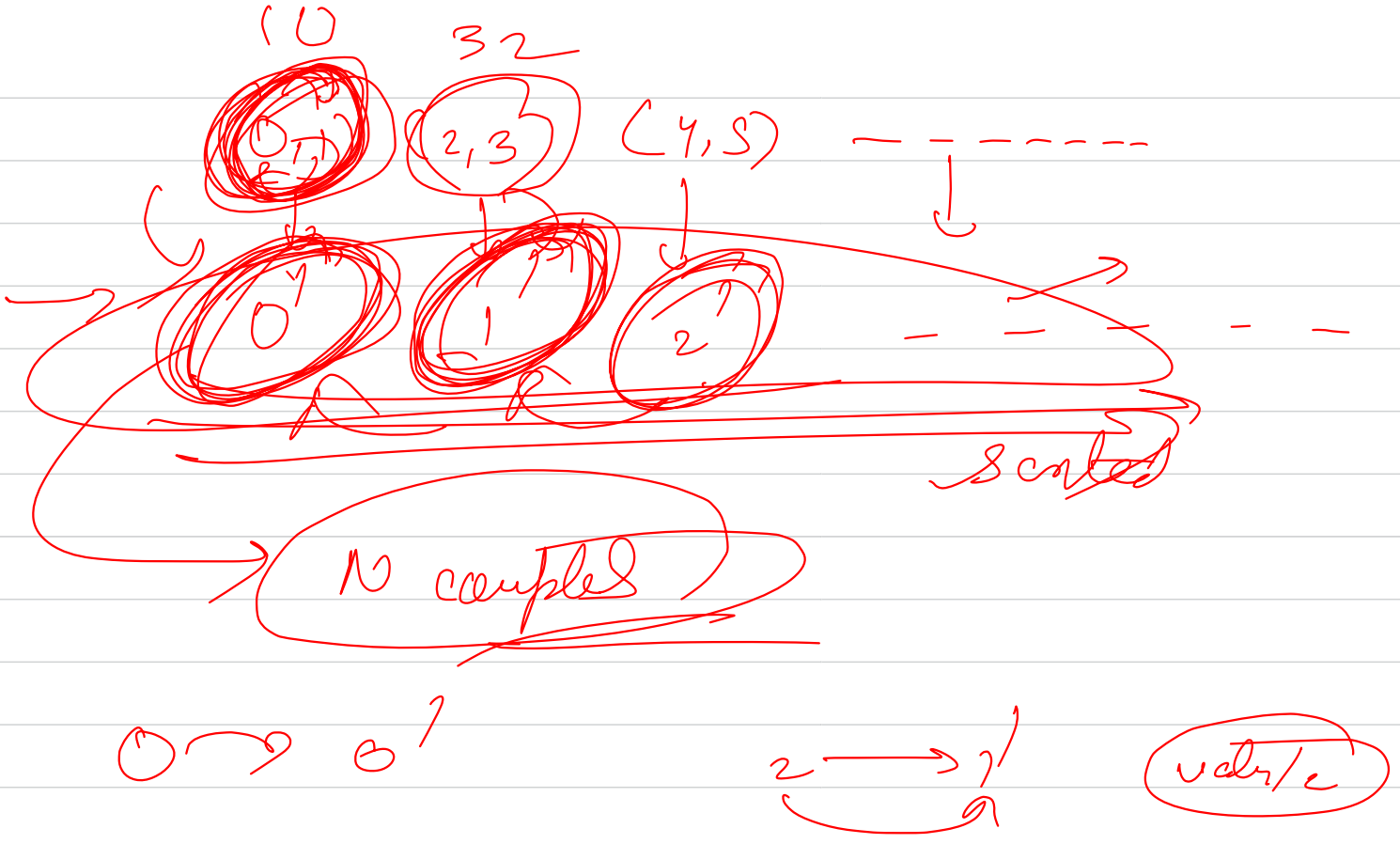
N $n_1 + n_2 + n_3 \cdots n_k$ =

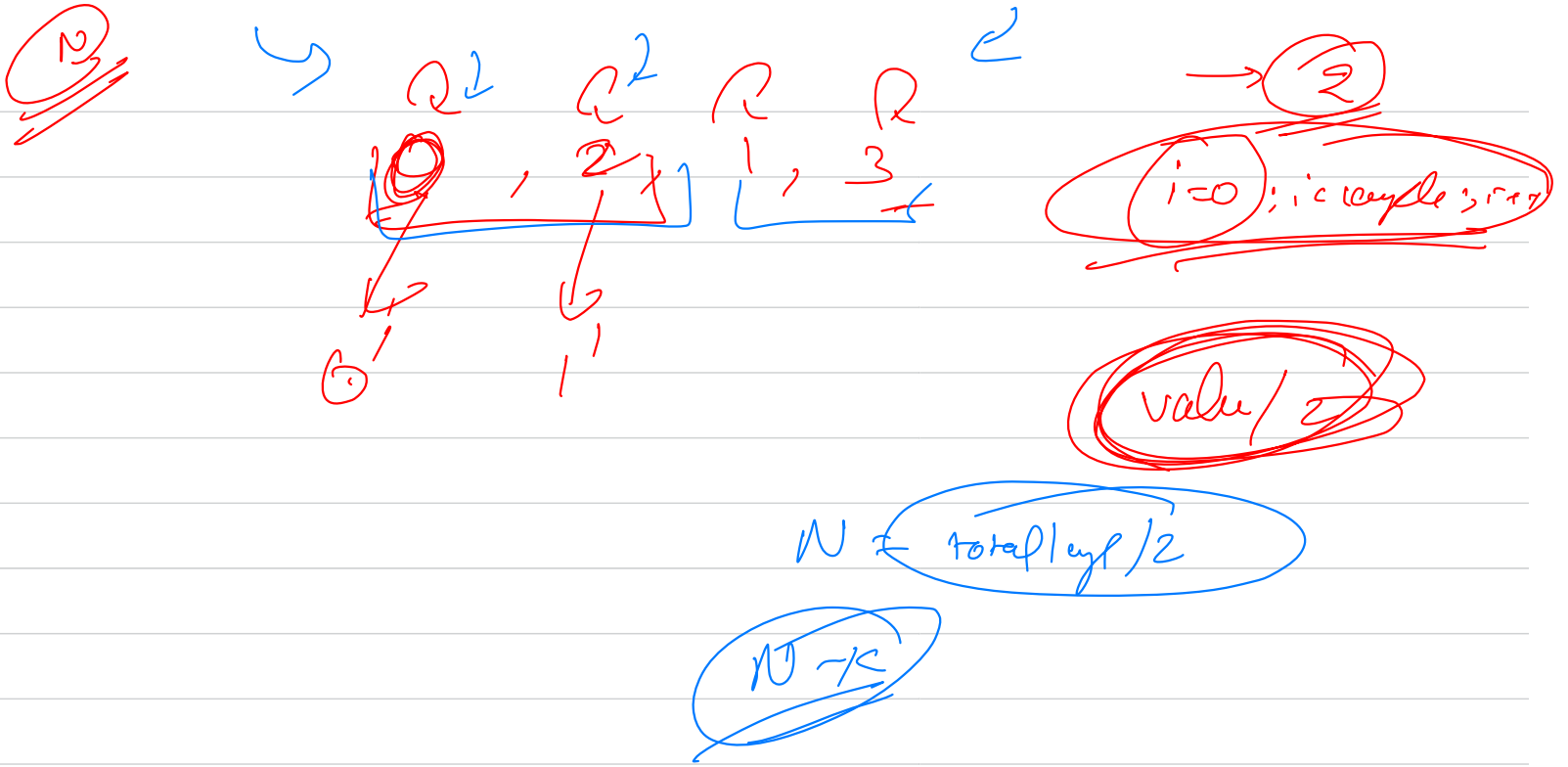
So to resolve all k components

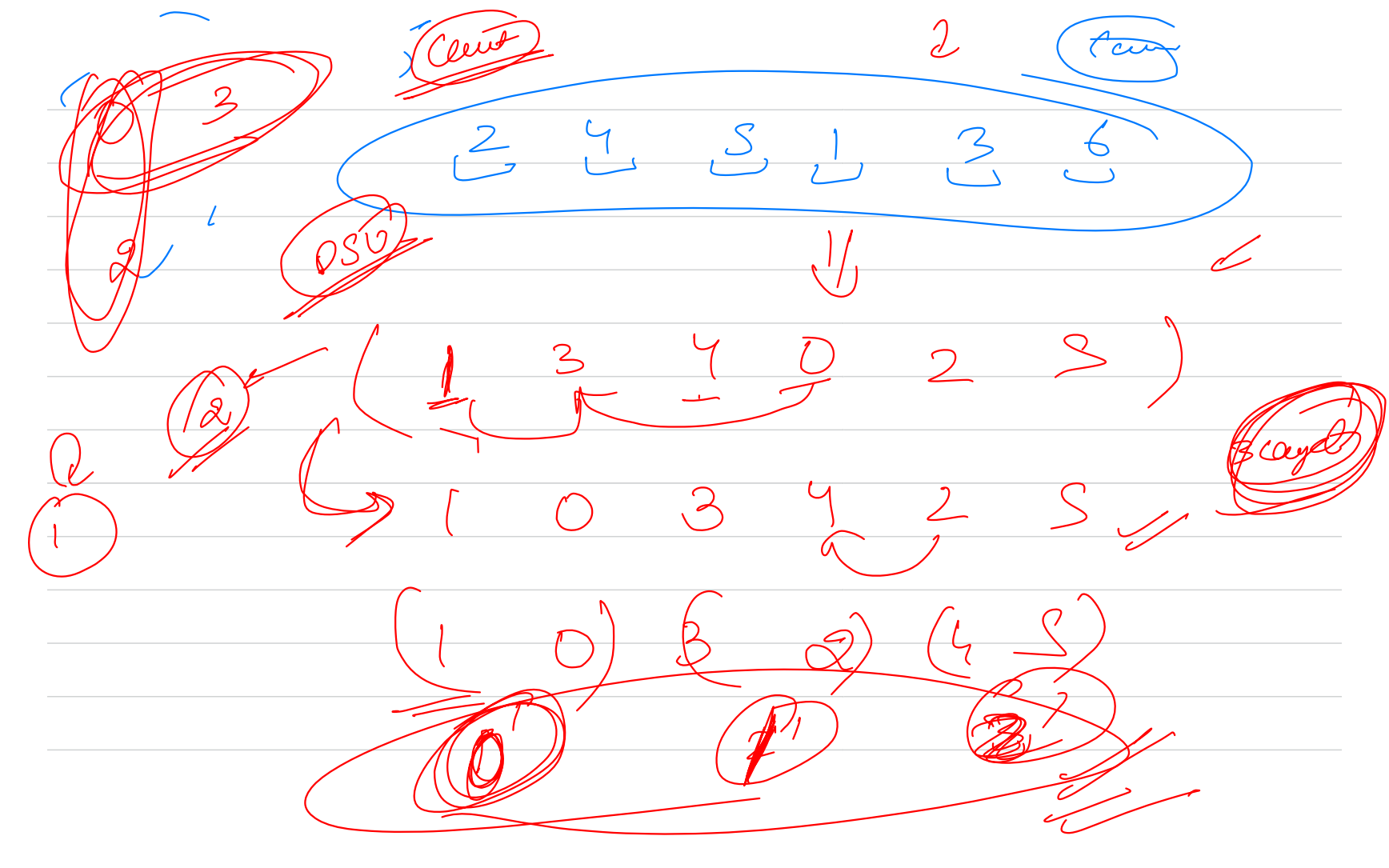
Total swaps = $n_1 - 1 + n_2 - 1 + n_3 - 1 + n_4 - 1 \cdots n_k - 1$

$= (n_1 + n_2 + n_3 \cdots n_k) - (1 + 1 + 1 + 1 + 1 \cdots$

N — — P


10
32
(0,1)
2,3
(4,5)
0'
1'
2'
scales
N couples
0 → 0'
2 → 1'
valid


2
i=0 ; i < couple ; i++
value / 2
N = totalEyp / 2
N - K



Medium

Q ⇒ 3

Directed graph

[0, n-1]

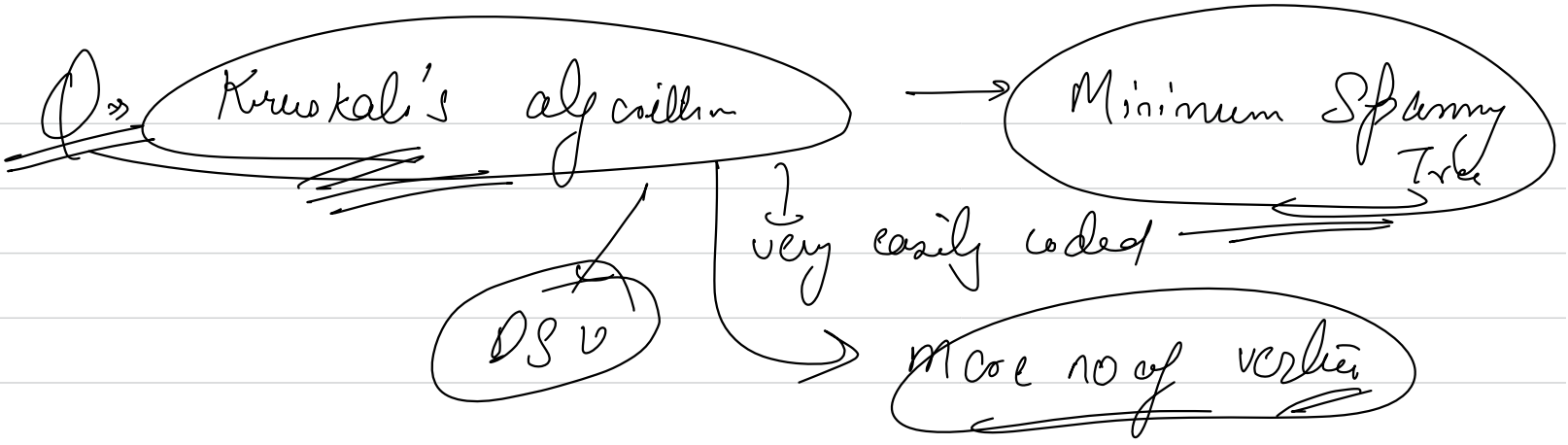
find all paths from 0 → n-1

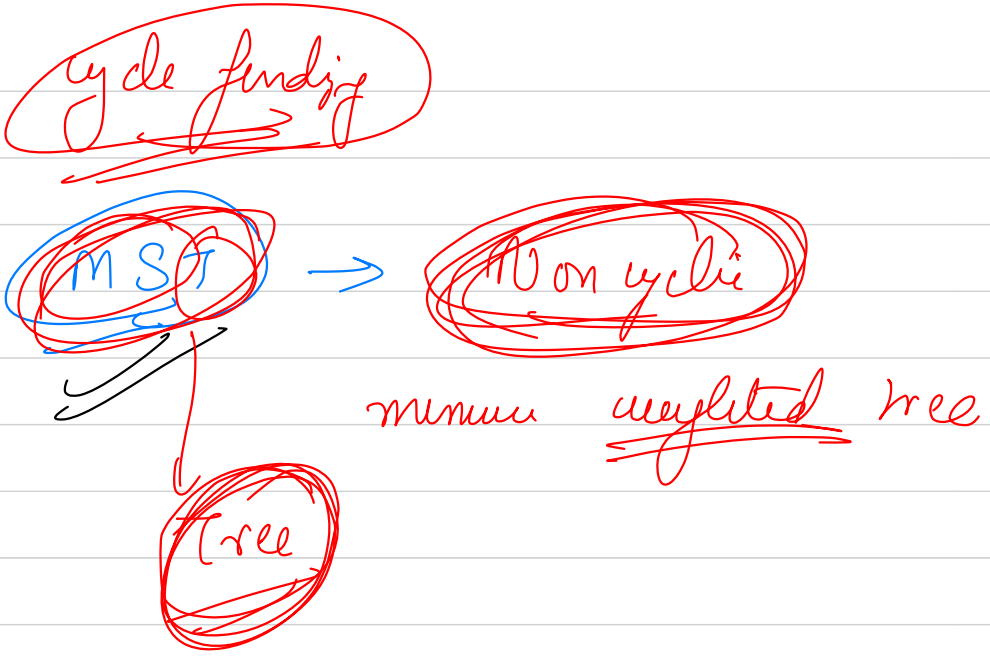
↳ Brute force

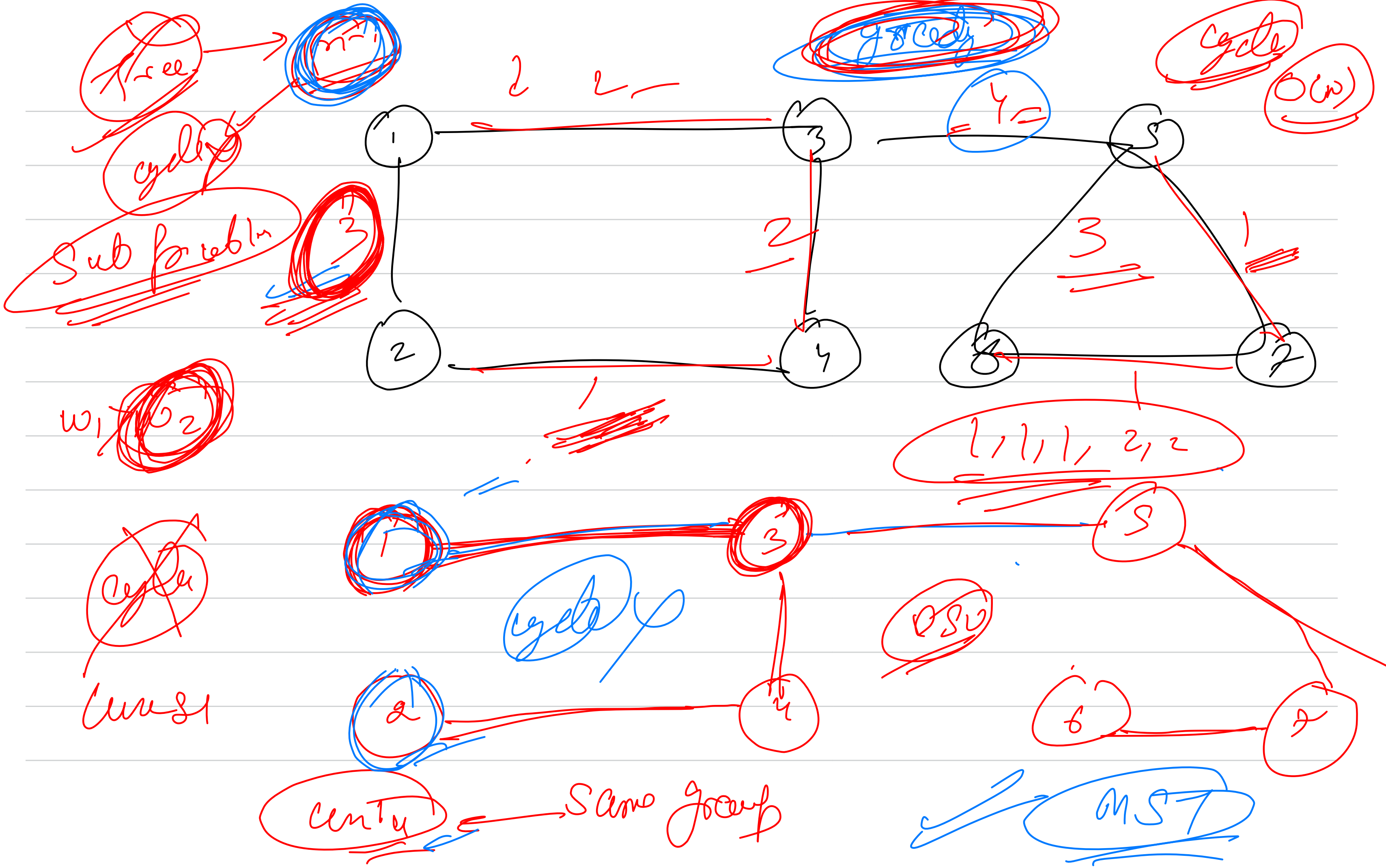
↳ DFS + Backtracking

n → 18

find n paths


Kruskal's algorithm
Minimum Spanning Tree
very easily coded
DSU
More no of vertices


cycle finding
MST
Non cyclic
minimum weighted tree
Tree


Tree
n-1
cycle
Sub problem
greedy
cycle
O(n)
w1
w2
1, 1, 1, 1, 2, 2
DSU
Same group
MST