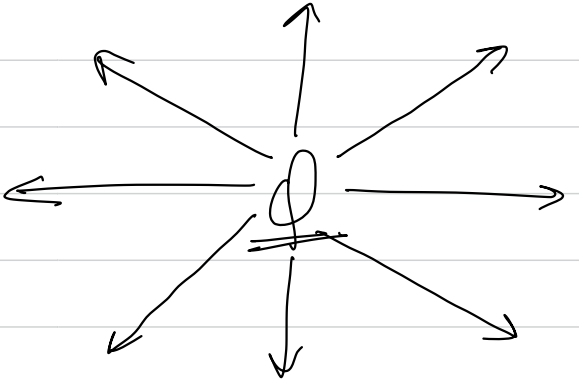
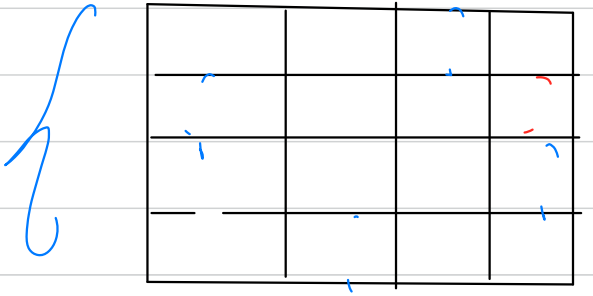
Q. What is pruning ??

↳ pruning is in the context with abandoning those sub problems which will never lead to a successful answer.

✱ N Queen problem

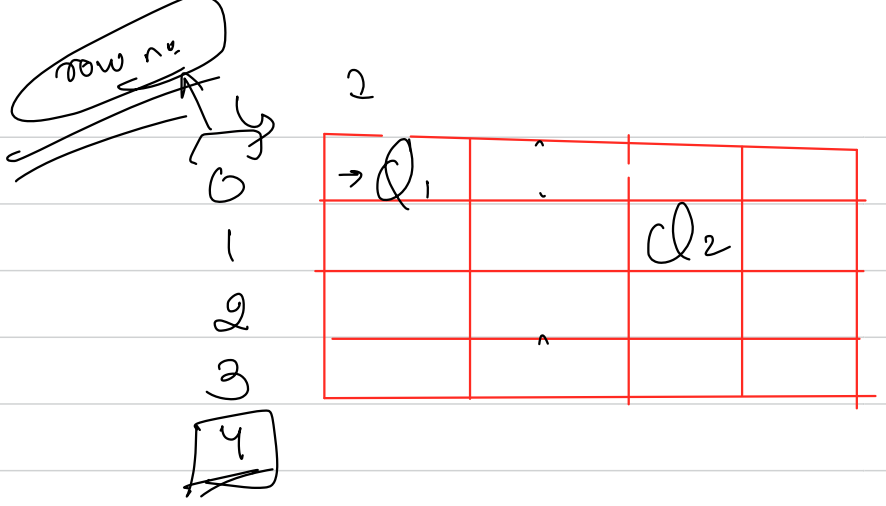




N x N

(N=4) → ans → 2

Given a N×N board, find the no. of ways to place N queens, so that no queen attacks the other.



Q_1, Q_2, Q_3, Q_4

$N \times N$

$N = 4$

→ · Before I place a queen, i will check that is it a safe spot to place.

Backtracking

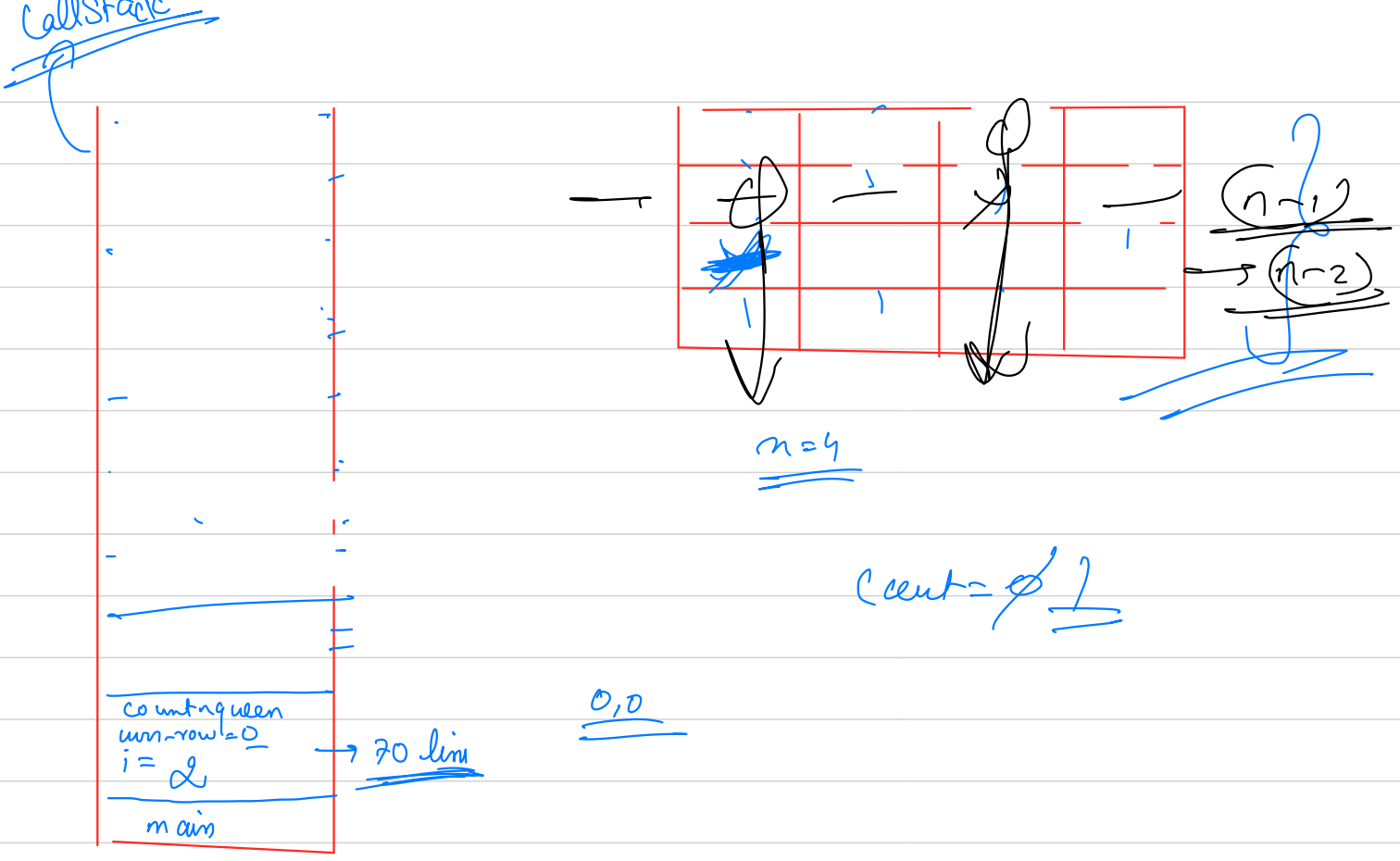
⇒ Let's try to write recursive solution.

Base Case ⇒

Recursion task → Recursively place N-1 queen in my N-1 rows

Self work → check if the spot is safe
if yes then place the queen

We will place our queen in current row


CallStack
countnqueen
unn-row=0
i = 2
→ 70 line
main
n=4
(n-1)
(n-2)
count=0
1
0,0

Space complexity → $O(n^2)$

$$T(n) = n \times T(n-1) + O(n^2)$$

$$T(n-1) = (n-1) \times T(n-2) + O(n^2)$$

$$T(n-2) = (n-2) \times T(n-3) + O(n^2)$$

$$\vdots \qquad \vdots \qquad \vdots$$

$$T(2) = 2 \times T(1) + O(n^2)$$

$T(n) = n \times T(n-1) + O(n^2)$

$T(n) = n\,(n-1) \times T(n-2) + O(n^2)\,) + O(n^2)$

$T(n) = n\,((n-1) \times ((n-2) \times T(n-3)$ - - - - -

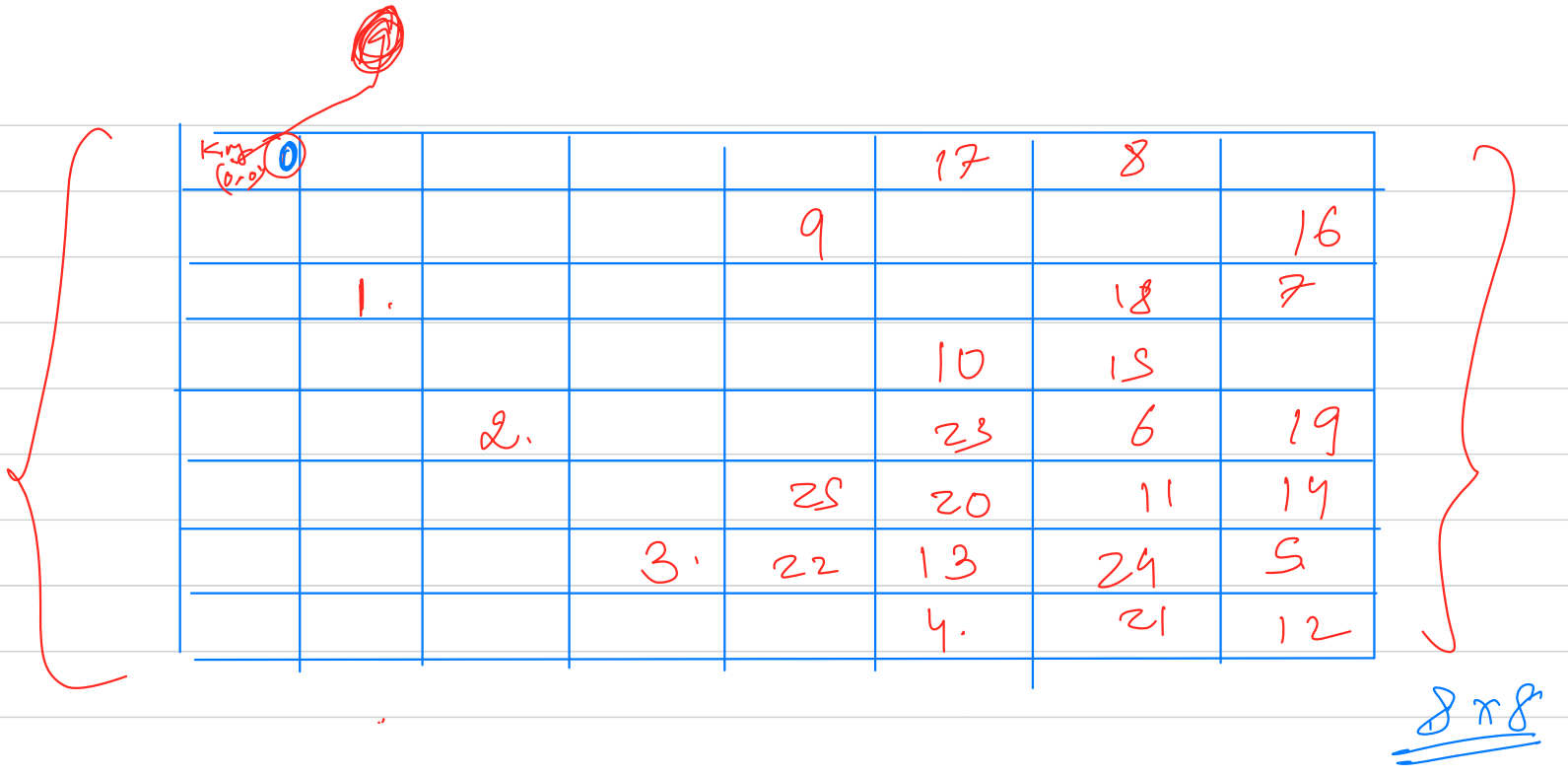
↳ $\Rightarrow O(n!)$

loose bound

$\Rightarrow n^n$

Tight bound

Q) Given a N * N board with a knight placed at (0, 0). Check if the knight can cover all the cells of the board, using it's usual chess moves without visiting any cell twice.


0
17
8
9
16
1.
18
7
10
15
2.
23
6
19
25
20
11
14
3.
22
13
24
5
4.
21
12
8 x 8