Agenda → Bitmasking based optimization for N queen

↳ New DS

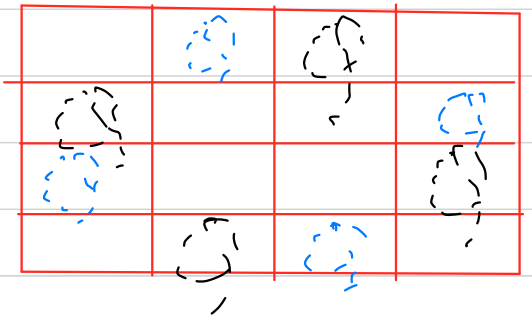
N Knights

Backtracking problems

N queen problem ⇒ Given a N×N board, place N queens such that no queen attacks the other.

Ex

basic backtracking

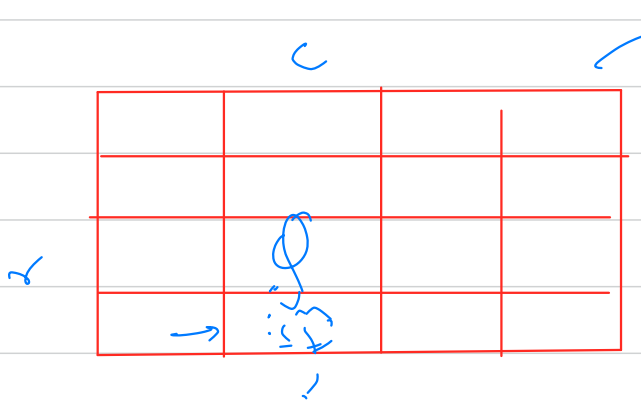


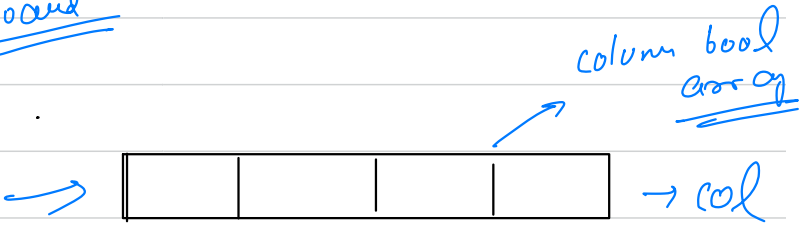
4×4

N = 4

1 → row → queen →

Boolean array → which represents column.





col [c] = true

col [c] = false

if (col [c] == true)

O(1)

	0	1	2	3
0	0,0	0,1	0,2	0,3
1	1,0	1,1	1,2	1,3
2	2,0	2,1	2,2	2,3
3	3,0	3,1	3,2	3,3

$0,1 \rightarrow -1$

$1,2 \rightarrow -1$

$2,3 \rightarrow -1$

$0,2 \rightarrow -2$

$1,3 \rightarrow -2$

$0,3 = -3$

(1) Left upper diagnol (O(1))

$0,0 \rightarrow 0$

$1,1 \rightarrow 0$

$2,2 \rightarrow 0$

$3,3 \rightarrow 0$

$1,0 \rightarrow 1$

$2,1 \rightarrow 1$

$3,2 \rightarrow 1$

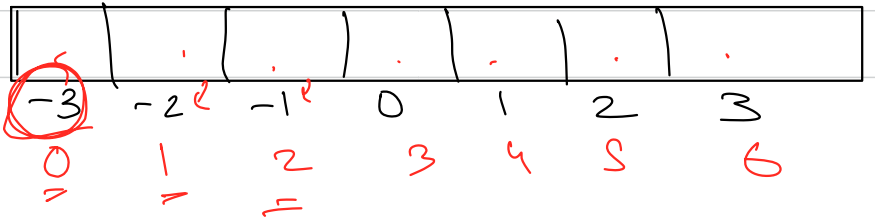
$2,0 \rightarrow 2$

$3,1 \rightarrow 2$

$3,0 \rightarrow 3$

$(-n-1) \leftrightarrow (n-1)$

Shifting

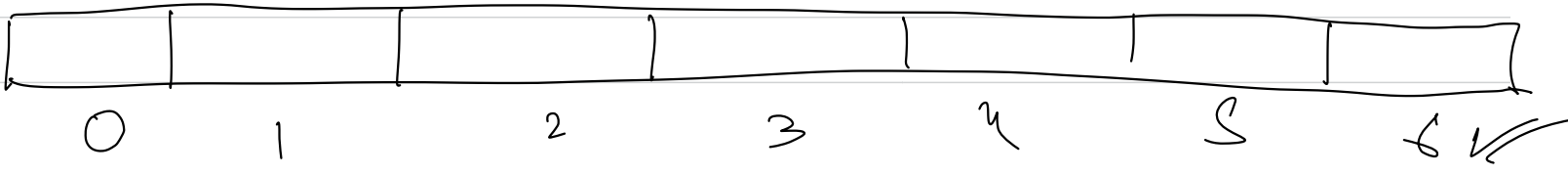


Size = 2N-1

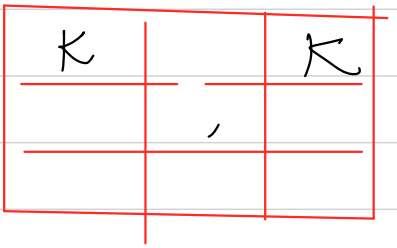
Right upper diagonal → O(1)

	0	1	2	3
0	0,0	0,1	0,2	0,3
1	1,0	1,1	1,2	1,3
2	2,0	2,1	2,2	2,3
3	3,0	3,1	3,2	3,3

0,0 → 0 | 1,0 → 1, 0,1 → 1 | 2,0 → 2, 1,1 → 2, 0,2 → 2 | 3,0 → 3, 2,1 → 3, 1,2 → 3, 0,3 → 3 | 3,1 → 4, 2,2 → 4, 1,3 → 4 | 3,2 → 5, 2,3 → 5 | 3,3 → 6

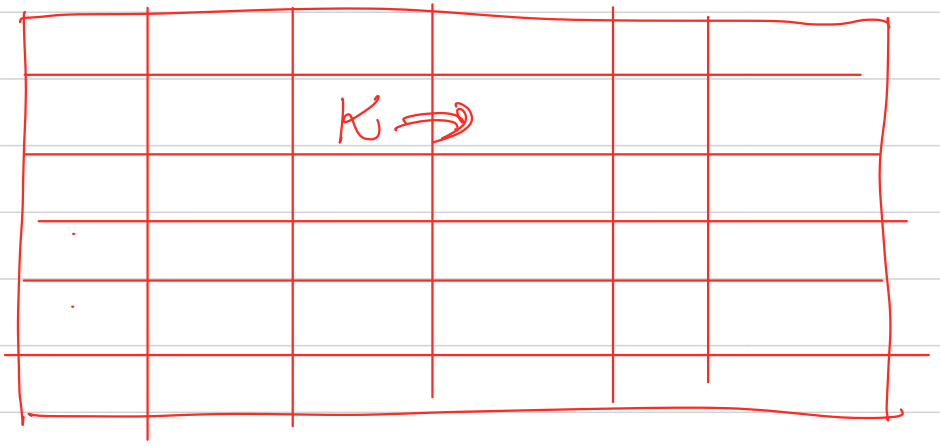


Q:) Given an $N \times N$ board, place N knights such that no knight attack each other



$N=3$

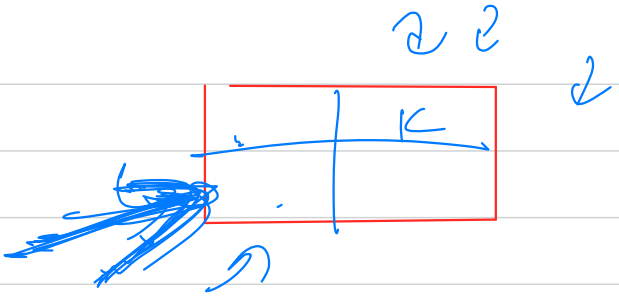
3×3



Curr Col + 1

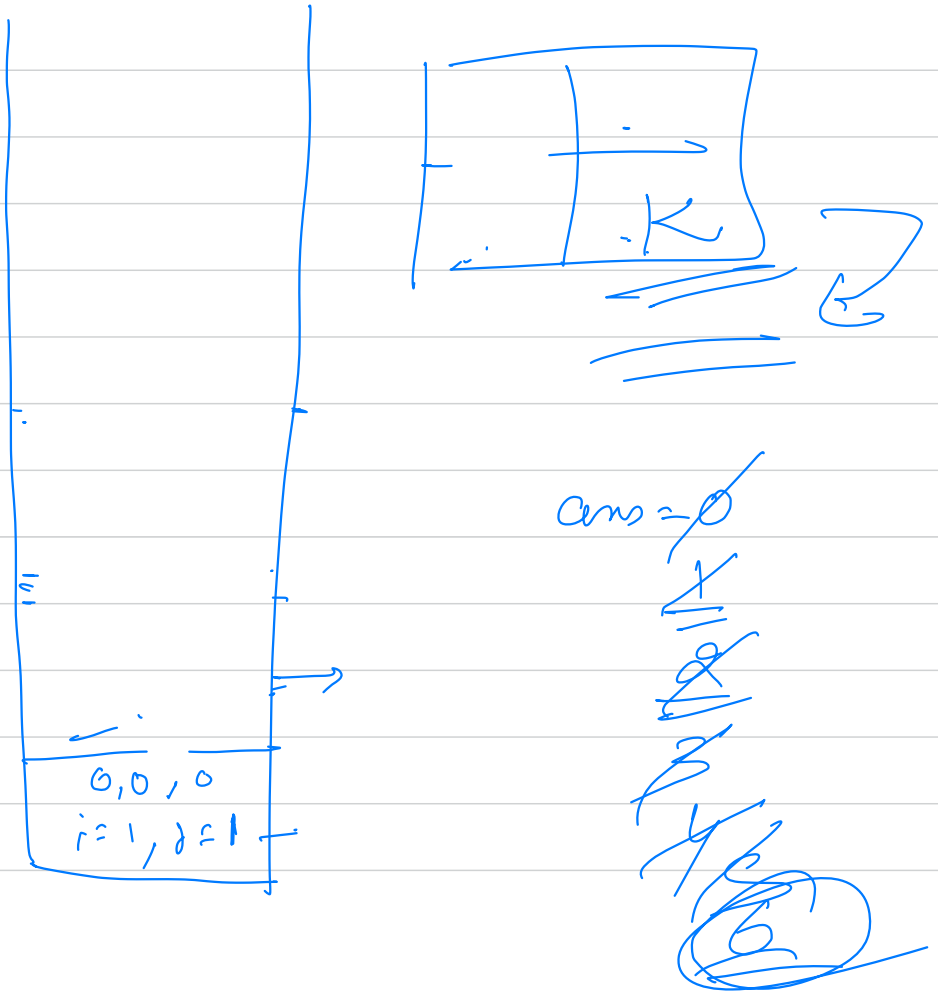
for (——

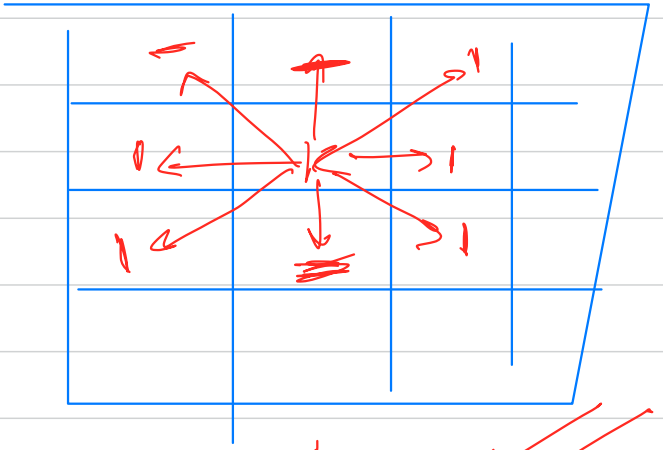
for (——



ans = ∅ 1

cc


K
ans = 0
1
2
3
4
5
0,0,0
i=1, j=1


C+1
C+2

Q.

a	b	a	g
o	c	n	i
z	q	d	z
y	n	d	z

character grid

["band", "can", "and", "cozy"] ← list of string