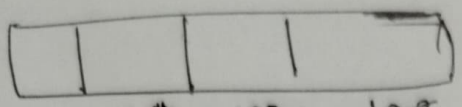
2D Array in Memory

char



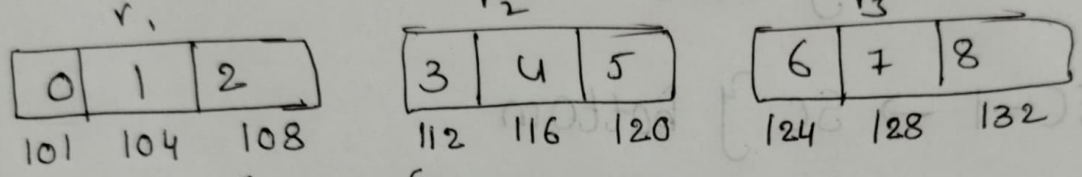
→ Addresses

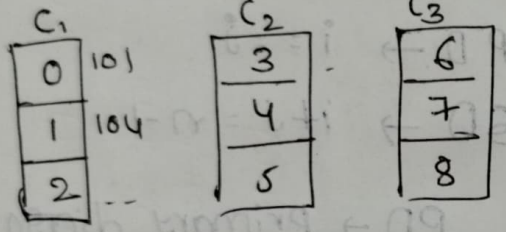
→ row Major

0	1	2
3	4	5
6	7	8

3x3
i j

i = row
j = col





Spiral Matrix

Approach

$\frac{n}{2}$ iterations

Start row
end row
Start col
end col

1st	2nd
→ 0	1
→ 3(n-1)	2
→ 0	1
3(n-1)	2

SR = 0

EC = n-1 = 3

1	2	3	4
5	6	7	8
9	10	11	12
13	14	15	16

SR = 1

ER = 2

ER = n-1 = 3

Start → ++

end → --

loop

while( )
{
1) top
2) Right

3) bottom

4) left

}

SR
SC → EC } TOP

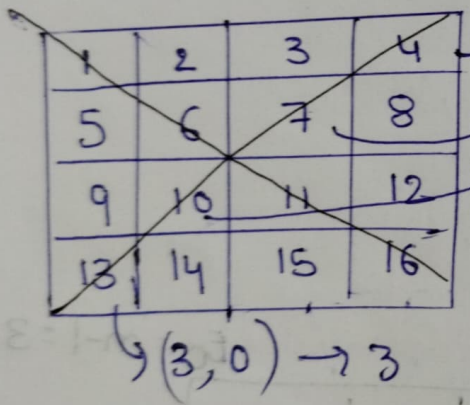
SR+1 → ER } Right

EC-1 → SC } bottom

left

ER-1 → SR+1

Diagonal Sum



PD → i == j

SD → i+j = n-1

PD → Primary diagonal

SD → Secondary diagonal

PD : 1 + 6 + 11 + 16 = 34

SD : 4 + 7 + 10 + 13 = 34

Secondary Method

i + j = n - 1

↓

j = n - i - 1