2D arrays

2 Dimensional Arrays. → __Matrixes__
↓
we will have rows & columns.

rows ↓

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

→ cols

arr →

(0,1)   (1,1)

| 1 | 2 | 3 | 4 |

(0,0)   (1,0)

100 108 10C

100
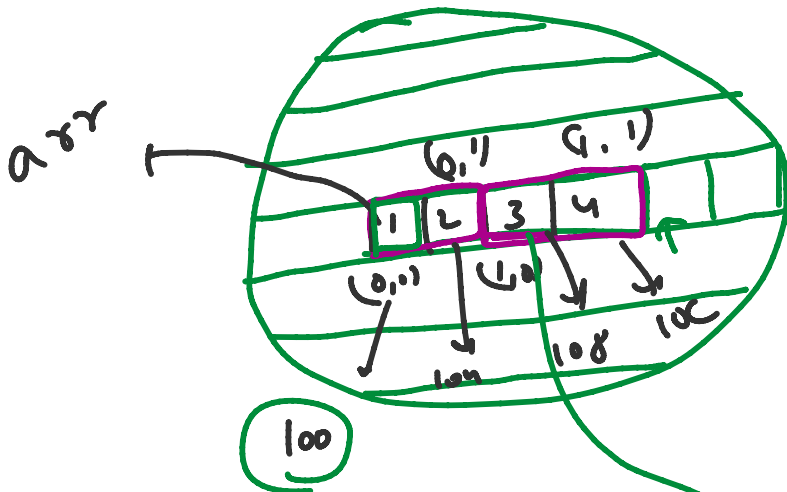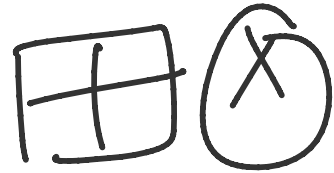
cout << arr
cout << arr[i] → second
bucket or arr

$arr[2][2]$
↓
2rows 2cols.

2x2 = 4 buckets

$arr[2][2]$
= 4

arr → error

arr[][]
$= \{ \{1, 2\}, \{3, 4\} \};$

X

r=1   c=4
r=2,3,4 c= 3 4
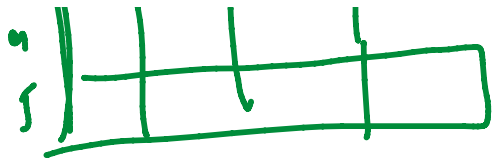
r = 2, 3, 4 c = 3 4

r = 5 , c = 4

→ rectangular

In 2D array   (row)   (matrix)

$arr[1]$ → 2nd bucket of arr

$arr[1][3]$ → 4th bucket of $arr[1]$

$arr[2]$ → $*(arr + 2)$

$arr[2][2]$ → ??

1	2	3	4

0,0 0,1 1,0 1,1

100 104 104 10C

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

$arr[i]$ → In case of 2D arrays.

↓

$\{ arr + 1 \times 4 \times columns \}$

or in general sense.

or in general sense.

$$\boxed{arr}[i] = \left(arr + i \times (\text{sizeof datatype}) \times \underline{(\text{total columns in a row})} \right)$$

int type 2D array

$$arr[i] = 100 + 1 \times 4 \times 2 = \boxed{108}$$

$arr \longrightarrow 100$

$$arr[1][1] = arr + 1 \times 4 \times 2 + 4 = \boxed{arr + 12}$$

$$arr[0][3] = arr + 3 \times 4 = \boxed{arr + 12}$$