

# # 'Arrays' :-

## '1-D Array' :-

$\text{int } a[10] = \{10, 20, 30, \dots, 100\}$  /\* it is an array of 10 elements, all are int \*/

1000	1002	1004	1006	1008	1010	1012	1014	1016	1018
10	20	30	40	50	60	70	80	90	100
0	1	2	3	4	5	6	7	8	9

$\rightarrow 1018-1010$

Ex: (1) :-

$$\text{Loc}(a[8]) = 1000 + (8-0) \times 2$$

$\uparrow$  int

$$= 1000 + 16 \Rightarrow \underline{1016}$$

\* Array size =  $\frac{j-i+1}{2}$

$\frac{1019-1000+1}{2}$

$\Rightarrow \underline{20}$  int

$\frac{20}{2} \Rightarrow 10$  element

$$\text{Loc}(a[4]) = 1000 + (4-0) \times 2$$

$$= 1000 + 8 \Rightarrow \underline{1008}$$

random access possible (due to this formula).  $O(1)$  time

Ex: (2) :-

$a[150 \dots 700]$  } Total elements =  $(j-i+1)$

$$= 700 - 150 + 1$$

$$= 550 + 1 = \underline{551}$$

Base addr = 1000

Size of element =  $\frac{10}{C}$

$$\text{Loc}(A[575]) = 1000 + (575-150) \times 10$$

$$= 1000 + 4250 = \underline{5250}$$

Ex: (3)

$a[-500 \dots -25]$  } total elements

$$= \underline{500 - 25 + 1}$$

$$= -25 + 500 + 1$$

$$= 475 + 1 = \underline{476}$$

B.A. = 0      C = 5

$$\text{Loc}(A[-150]) = 0 + (-150 + 500) \times 5$$

$$= (\underline{350}) \times 5$$

$$= \underline{1750}$$

Note :-

A [ lb ..... ub ]

# of elements :-  $\frac{ub - lb + 1}{1}$

→  $\underline{B \cdot A} = \underline{BA}$  element size = 'C'  
 Base Addr. ↗

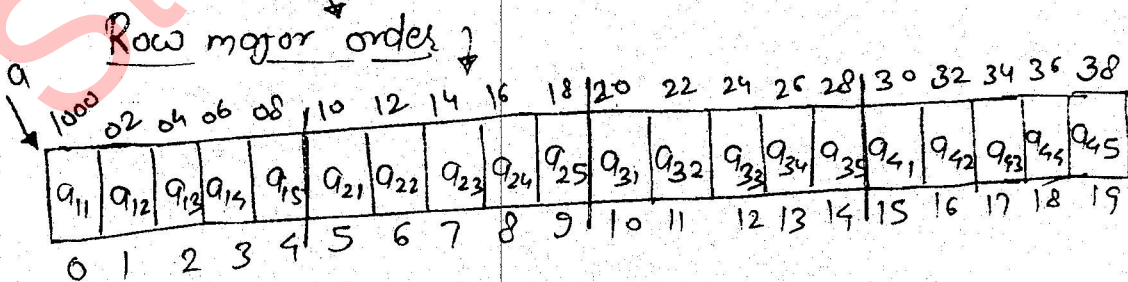
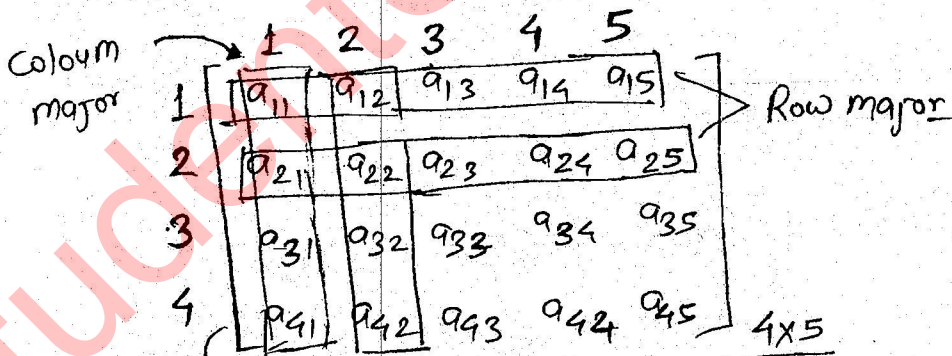
$$\text{Loc}(A[i]) = B \cdot A + [i - lb] \times C$$

\* Note :-

By default array index will always start from '0' but not from '1' because no need to calculate offset value everytime.

# "2D-Arrays" :-

int a [ 1 --- 4, 1 --- 5 ]  
 ↓ ↓ ⇒ 20 elements  
 4 rows 5 columns  
 [4-1+1] [5-1+1]



$$\text{Loc}(A[4][3]) = 1000 + \left[ \overset{\downarrow \text{Row}}{(4-1) \times 5} + \overset{\downarrow \text{Column}}{(3-1)} \right] \times 2$$

$$= 1000 + 34 = \underline{1034} \quad \text{element}$$

$$\text{Loc}(A[2][5]) = 1000 + [(2-1) \times 5 + (5-1)] \times 2$$

$$= 1000 + 18 = \underline{1018}$$

Ex: ② :-  $A [ 25 \dots 250, 79 \dots 527 ]$     Row = 226    Column =  $\frac{448+1}{449}$

$B.A. = 1000, \quad C = 10B$     Row major order

$$\text{Loc}(A[201][423]) = 1000 + [(201-25) \times 449 + (423-79)] \times 10$$

$$= 1000 + 79360$$

$$= \underline{79460}$$

Ex: ③ :-  $A [ -200 \dots 200, -75 \dots +75 ]$     Row =  $\frac{200+200}{+1} = 401$     Column =  $\frac{75+75}{+1} = 151$

$BA = 0 \quad C = 5B$     Row major.

$$\text{Loc}(A[175][3]) = 0 + [(175+200) \times 151 + (3+75)] \times 5$$

$$= \underline{(283515)}$$

Note :-  $A [ lb_1 \dots ub_1, lb_2 \dots ub_2 ]$     Column =  $\frac{ub_2 - lb_2 + 1}{}$

Base Addr. = BA    Size (element) = C    Row major

$$\text{Loc}(A[i][j]) = B.A. + [(i - lb_1) \times (ub_2 - lb_2 + 1) + (j - lb_2)] \times C$$

nc  
neg ul

∴ Column major order :-

Ex:  $A [25 \dots 250, 79 \dots 527]$

<u>Row</u>	<u>Column</u>
$250 - 25 + 1$	$527 - 79$
$= 226$	$+ 1$
	$= \underline{449}$

Base addr. = 1000     $C = 10B$

Column major order

↓ go to the col 1st  
 ↓ Row

$$\text{Loc}(A[201][423]) = \frac{B \cdot A}{1000} + ((423 - 79) \times 226 + (201 - 25) \times 10)$$

then Row →

$$= 1000 + 779200 = \underline{780200}$$

Ex: (2)  $A [-200 \dots +200, -75 \dots +75]$

<u>Row</u>	<u>Col</u>
$200 + 200$	$75 + 75$
$+ 1$	$+ 1$
$\underline{401}$	$\underline{151}$

$B \cdot A = 0, C = 5B$     Column major

$$\text{Loc}(A[175][3]) = 0 + ((3 + 75) \times 401 + (175 + 200) \times 5)$$

→ rows  
 get col 1st ←  
 → then row

$$= \underline{158265}$$

\* Note :-  $A [lb_1 \dots Ub_1, lb_2 \dots Ub_2]$   
 $B \cdot A, C, \text{Col major order}$

$$\text{Loc}(A[i][j]) = B \cdot A + ((j - lb_2) \times \underbrace{(Ub_1 - lb_1 + 1)}_{nr}) + (i - lb_1) \times C$$

\* By default C Language follows "Row major order"  
 or  
 other programming Language

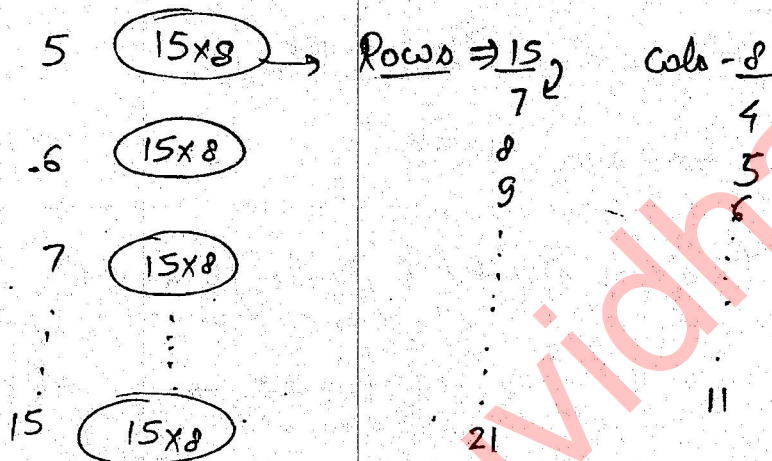
# 3-D Array

int a [ 5 ..... 15, 7 ..... 21, 4 ..... 11 ]

3-D:- Collection of 2D's

int a [ 11 ] [ 15 ] [ 8 ]

11 2D's, each of size (15x8)



int a [ 5 ..... 15, 7 ..... 21, 4 ..... 11 ]

B.A = 1000, C = 10B Row major

$$\text{Loc } (A[12][20][7]) = BA +$$

$$= ((12-5) \times 15 \times 8 + (20-7) \times 8$$

$$+ (7-4)) \times 10$$

$$= 10470$$

Colo major:-

$$\begin{array}{r} 780 \\ + 1000 \\ \hline 1780 \\ + 1000 \\ \hline 2780 \end{array}$$

$$\text{Loc } (A[12][20][7]) = 1000 + ((12-5) \times 15 \times 8$$

$$+ ((7-4) \times 15) + (20-7) \times 10$$

$$= 9980$$

Note

i)  $A [ lb_1, \dots, ub_1, lb_2, \dots, ub_2, lb_3, \dots, ub_3 ]$   
 BA, C, Row major  
 (required 2D)  $\rightarrow$  # of 2D's

$$\text{Loc}(A[i][j][k]) = BA + ((i - lb_1) \times nr \times nc) + (j - lb_2) \times nc + (k - lb_3) \times c$$

$\uparrow$  required row     $\uparrow$  col     $\uparrow$  element size

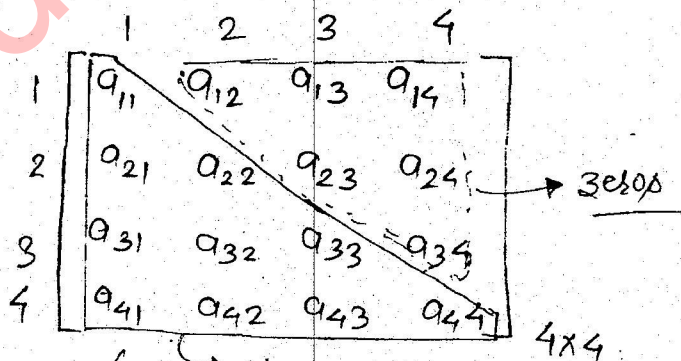
ii)  $A [ lb_1, \dots, ub_1, lb_2, \dots, ub_2, lb_3, \dots, ub_3 ]$   
 BA, C, ~~Row~~ col major

$$\text{Loc}(A[i][j][k]) = BA + ((i - lb_1) \times nr \times nc) + (k - lb_3) \times nr + (j - lb_2) \times c$$

$\rightarrow$  required 2D  
 $\rightarrow$  required col  
 $\downarrow$  Row

# "Lower Triangular matrix" :-

int a[1...4, 1...4]



$\rightarrow$  Non-zero

$$\frac{\text{elements (Non zero)}}{2} = \frac{n(n+1)}{2} \Rightarrow \frac{4(4+1)}{2} = \underline{10}$$

∴ All Triangular matrices are square matrices.

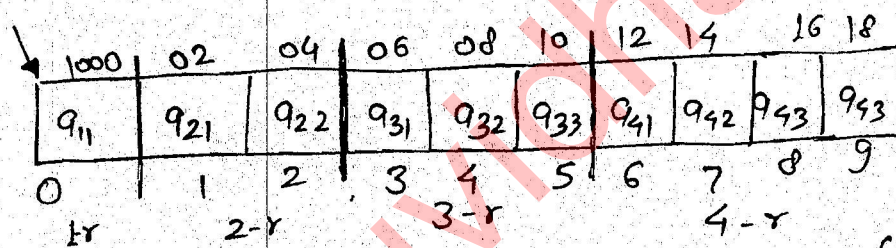
∴ Storing zero :- unnecessary space wastage.

\* :-  $a[i][j]$   $[j \neq i \leq j]$   $\rightarrow$  element belongs to lower triangle.

\* :-  $a[i][j]$   $[j \neq j > i]$   $\rightarrow$  upper triangle return 0;

∴ Starting position of Row & Column must be same.

∴ Storing in 'Row major order' :-  
 Ex:  $A[25 \dots 150, 25 \dots 150]$



Ex: ①

$$\begin{aligned} \text{Loc}(A[4][3]) &= \frac{B.A}{1000} + \left[ (4-1) \times \frac{3 \times 4}{2} \right] \\ &= 1000 + \left[ \frac{(4-1)(4-1+1)}{2} + (3-1) \right] \times 2 \\ &= 1016 \end{aligned}$$

Ex: ②

$$A[25 \dots 150, 25 \dots 150]$$

$\downarrow 126$                        $\rightarrow 126$

Base addr. = 1000      C = 10, Lower Triangular Matrix, Row major

$$\begin{aligned} \text{Loc}(A[125][105]) &= 1000 + \left( \frac{(125-25)(125-25+1)}{2} + (105-25) \right) \times 10 \\ &= 1000 + \left( \frac{50 \times 101}{2} + 80 \right) \times 10 \\ &= 52300 \end{aligned}$$

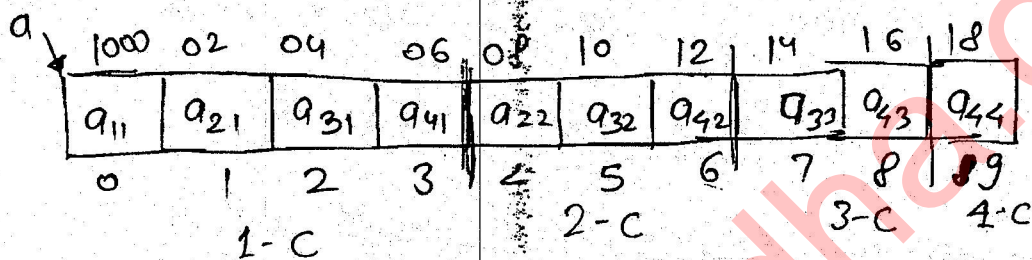
Note  $A [lb_1 \dots ub_1, lb_2 \dots ub_2]$

$B.A.$ ,  $C$ , Lower triangular matrix, Row major

$$\text{Loc}(A[i][j]) = B.A. + \left( \frac{(i-lb_1)(i-lb_1+1)}{2} + (j-lb_2) \right) \times C$$

If  $i \geq j$  else return 0;

Column major



$$\text{Loc}(A[4][3]) = 1000 + \left[ \frac{(3-1)(3-1+1)}{2} + (4-1) \right] \times 2$$

~~1000 + [ (3-1)(3-1+1) ]~~ → all col crossed (now on right side)

$$\text{Loc}(A[4][3]) = 1000 + \left[ \left( \frac{4 \times 5}{2} - \frac{2 \times 3}{2} \right) + (4-3) \right] \times 2$$

$$= 1016$$

Ex - ②  $A [25 \dots 176, 25 \dots 200]$

$B.A. = 1000$ ,  $C = 10B$ , Lower Triangular matrix & Column major

$$\text{Loc}(A[150][90]) = 1000 + \left[ \left( \frac{176 \times 177}{2} - \frac{111 \times 112}{2} \right) + (150-90) \right] \times 10$$

$$= 95200$$

$$\frac{176-150}{25}$$

or

$$= 1000 + \left[ \frac{176 \times 177}{2} - \frac{(200-90+1)(200-90+2)}{2} + (150-90) \right] \times 10$$



\* Note :-

$$X [ lb_1 \dots ub_1, lb_2 \dots ub_2 ]$$

B.A, C, Lower Triangle matrix, Coloum major

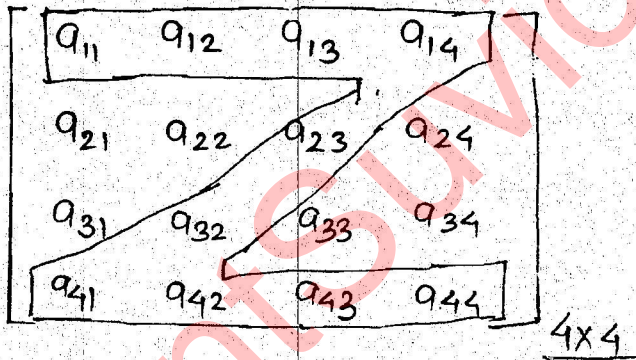
$$Loc(A[i][j][j]) = B.A + \left[ \frac{(ub_2 - lb_2)(ub_2 - lb_2 + 1)}{2} \right.$$

Lower triangle so  $i \geq j$

$$- \frac{(ub_2 - j)(ub_2 - j + 1)}{2}$$

$$+ (i - j) ] \times C$$

$$= B.A + \left[ \frac{(nc)(nc+1)}{2} - \frac{(ub_2 - j + 1)(ub_2 - j + 2)}{2} + (i - j) \right] \times C$$



$$Z = \frac{3n - 2}{2} \text{ elements}$$

Row major

$$\frac{n + 1 + 1 + \dots + n}{1^{st} \text{ last}}$$

$$50^{th} \text{ Row} \Rightarrow \frac{49 \text{ rows}}{N + 48}$$

Coloum major

$$\frac{2 + 3 + 3 + 3 + \dots + 2}{1^{st} \text{ last}}$$

$$50^{th} \text{ Coloum} \Rightarrow \frac{48 \times 3 + 2}{2}$$