ADDRESS CALCULATION ARRAY.

ADDRESS CALCULATION ROW-MAJOR ORDER -3- D

$$\boxed{\text{Address of } arr[i][j][K] = \text{Base Address} + [(i-L1).D_2.D_3 + (j-L_2).D_3 + (K-L_3)]W}$$

W= ELEMENT SIZE

Ex→ Base Address = 400 , Element Size = 2 Bytes

→ Size of an Array = [1:9 , -4:1 , 5:10]

L_1 U_1 L_2 U_2 L_3 U_3

→ arr [5][-1][8]

i j K ← Indicies of desired elements.

→ L_1, L_2, L_3 → Lower Bound of each dimension

→ D_1, D_2, D_3 → are Size of each dimension.

Dimension

D_1 = [1:9] = $U_1 - L_1 + 1$ = 9-1+1 = 9

D_2 = [-4:1] = $U_2 - L_2 + 1$ = 1-(-4)+1 = 6

D_3 = [5:10] = $U_3 - L_3 + 1$ = 10-5+1 = 6

Calculation :→

$400 + [(5-1)*6*6 + (-1-(-4))*6 + (8-5)]$. Element Size(

$400 + [165].2$

$400 + 330$

730-ANSWER.

ADDRESS CALCULATION COLUMN-MAJOR ORDER - 3-D ARRAY.

$$arr[i][j][k] = \text{Base Address} + [(k-L_3).D_1.D_2 + (j-L_2).D_1 + (i-L_1)].\text{Element size.}$$

2-D ARRAY.

Row-major Order - 2-D array

$$arr[i][j] = \text{Base Address} + [(i-L_1)D_2 + (j-L_2)].\text{Element Size}$$

Column-major Order - 2-D array

$$arr[i][j] = \text{Base Address} + [(j-L_2).D_1 + (i-L_1)].\text{Element Size}$$

4-D ARRAY.

Row-major Order

$$arr[i][j][k][l] = \text{Base Address} + [(i-L_1)D_2D_3D_4 + (j-L_2)D_3D_4 + (k-L_3).D_4 + (l-L_4)]\text{Element Size}$$

Column-Major Order

$$arr[i][j][k][l] = \text{Base Address} + [(l-L_4).D_3D_2D_1 + (k-L_3)D_2D_1 + (j-L_2)D_1 + (i-L_1)]\text{Element Size}$$