) One - Dimensional Array :

By the previous definition of 1-Dimensional array, we can say that the compiler limits the storage region to storing set of element, and the first location is individual element of array , and this called the **Base Address**.

For example :

let's be as 500. Base Address (501) and like for the all elements and used the index I, by its value are range 1<= I => N according to Base Index (500), by using this relation:

Location (X[I]) = Base Address + (I-1)

For example :

When the requirement is to bound the forth element, I = 4 :

Location(X[4])= 500+(4-1)
= 500 +3
= 503

So the address of forth element is 503 because the first element in 500.

When the program indicate or dealing with element of array in any instruction like (write (X [I]), read (X [I])), the compiler depend on going relation to bounding the requirement address.

2) Two - Dimensional Array :

A two dimensional Array A is the collection of 'm X n' elements. Programming language stores the two dimensional array in one dimensional memory in either of two ways -

1) Row Major Order:

First row of the array occupies the first set of memory locations reserved for the array; Second row occupies the next set, and so forth.

To determine element address A[i,j]:

Location (A[i,j]) =Base Address + (N x (I - 1)) + (j - 1)

For example :

Given an array [1…5,1…7] of integers. Calculate address of element T[4,6], where BA=900.

Solution:- I = 4 , J = 6 ,M= 5 , N= 7
Location (T [4,6]) = BA + (7 x (4-1)) + (6-1)
= 900+ (7 x 3) +5
= 900+ 21 + 5
= 926

2) Column Major Order:

Order elements of first column stored linearly and then comes elements of next column.

To determine element address A[i,j]:

Location (A[i,j]) =Base Address + (M x (j - 1)) + (i - 1)

For example :

Given an array [1…6,1…8] of integers. Calculate address element T[5,7], where BA=300.

Solution:- I = 5 , J = 7, M= 6 , N= 8

Location (T [4,6]) = BA + (6 x (7-1)) + (5-1)

= 300+ (6 x 6) +4

= 300+ 36+4

= 340

2) Three - Dimensional Array :

In three - dimensional array also address is calculated through two methods i.e; row-major order and column-major method.

To calculate address of element X[i,j,k] using row-major order :

Location (X[i,j,k])=BA + MN (k-1) + N (i-1) + (j-1)

To calculate address of element X[i,j,k] using column-major order

Location (X[i,j,k])=BA + MN (k-1) + M (j-1) + (i-1)

For example :

Given an array [1..8, 1..5, 1..7] of integers. Calculate address of element A[5,3,6], by using rows and columns methods, if BA=900?

Solution:- The dimensions of A are :

M=8 , N=5, R=7, i=5, j=3, k=6

Rows - wise :

Location (A[i,j,k]) = BA + MN(k-1) + N(i-1) + (j-1)

Location(A[5,3,6])= 900 + 8x5(6-1) + 5(5-1) + (3-1)

= 900 + 40 x 5 +5 x 4 + 2

= 900 + 200 +20 +2

= 1122

Columns - wise :

Location (A[i,j,k]) = BA + MN(k-1) + M(j-1) + (i-1)

Location (A[5,3,6]) = 900 + 8x5(6-1) + 8(3-1) + (5-1)

= 900 + 40 x 5 +8 x 2 + 4

= 900 + 200 +16 +4

= 1120