Two-Dimensional Arrays

- A 2-dimensional array is a collection of elements placed in m-rows and n-columns.
- The Syntax used to declare a 2-D array includes two subscripts, of which one specifies the number of rows and the other specifies the number of columns of an array.

Ex arr[3][4] is a 2-D array containing 3 rows and 4 columns and arr[0][2] is an element placed at 0th row and 2nd column in the array. The two dimensional array is also called a matrix.

- The pictorial representation of a matrix is shown below.

COLUMN

↓ \ →	0	1	2	3
0	12	1	-9	23
1	14	7	11	121
2	6	78	15	34

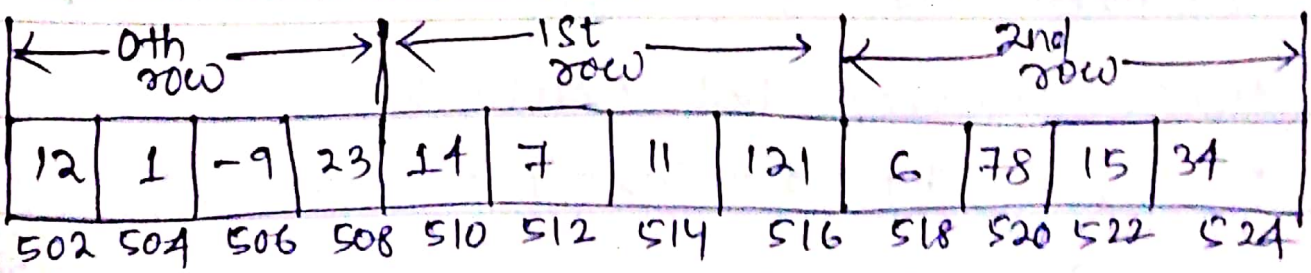
ROW

Row Major and Column Major Arrangement

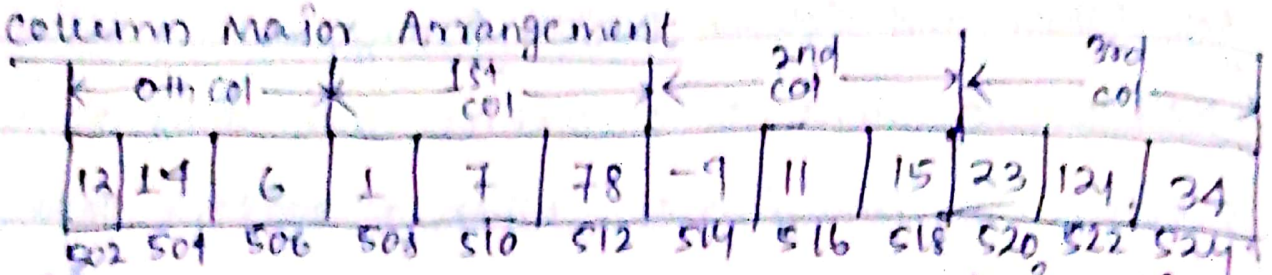
- Row and columns of a matrix are only a matter of imagination. When a matrix get stored in memory all its elements are linearly since computer's memory can only be viewed as consecutive units of memory locations. This leads to two possibles arrangements of elements in memory - Row major arrangement and column Major Arrangement.

Ex
```
int a[3][4] = { {12,1,-9,23},
                {14,7,11,121},
                {6,78,15,34}
              };
```

Row Major Arrangement :-



Columnn Major Arrangement



- Since the array elements are stored in adjacent memory locations we can access any element of the array once we know the base address (starting address) of the array and number of rows and columns present in the array.

Ex - If the base address of the array is 502 and we wish to refer the element 121, then the calculation involved would be as follows:

Row Major Arrangement

Element 121 is present at a[1][3].

Hence location 121 would be

= 502 + 1*4 + 3 = 502 + 7 = 516

- In general, for any array a[m][n] the address of element a[i][j] would be Base address + i*n + j.

Column Major Arrangement

- Element 121 is Present at a[1][3]. Hence location of 121 would be ~~502+3*3+1~~
  = 502 + 3*3 + 1 = 502 + 10 = 522
- In general for an array a[m][n] the address of element a[i][j] would be Base address + j*m + i
- Note that C language permits only Row Major arrangement.

Common Matrix operation:

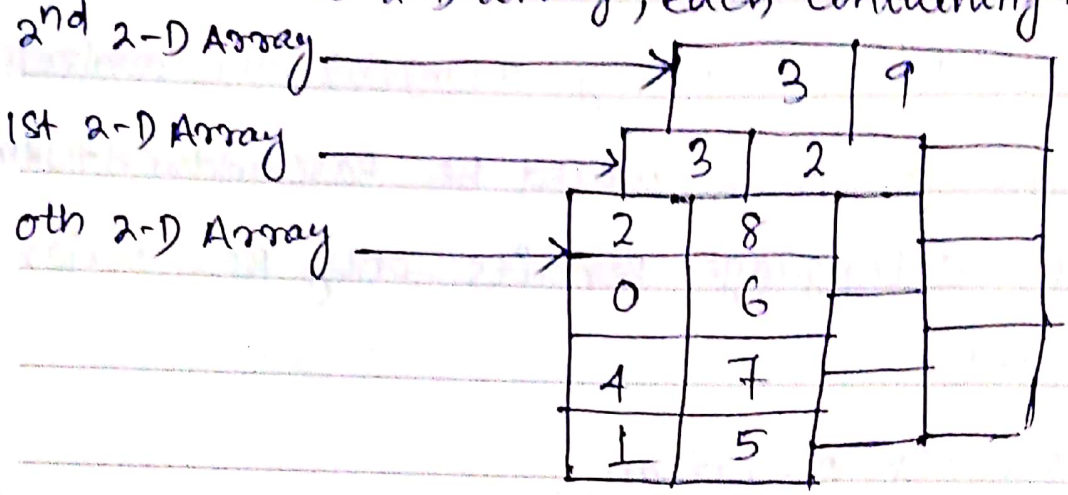
- Common matrix operations are addition, Multiplication and transposition.

Assignment

- Write a C Program of matrix operations which includes addition, Multiplication and transpose.

Multidimensional Arrays

- A 3-dimensional array can be thought of as an array of arrays of arrays.
- The below fig shows a 3D Array, which is a collection of three 2-D array, each containing 4 rows and 2 columns.

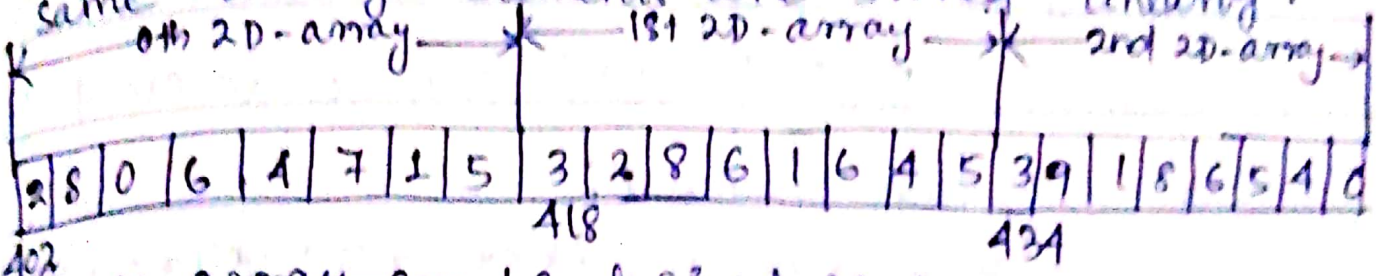


- This array can be defined as;

```
int a[3][4][2] = {
                    { {2,8}, {0,6}, {4,7}, {1,5} },
                    { {3,2}, {8,6}, {1,6}, {4,5} },
                    { {3,9}, {1,8}, {6,5}, {4,0} }
                  };
```

- The outer array has three elements, each of which is a 2D-array, which in turn holds four 1D arrays containing two integers each.

- Note that the arrangement shown in figure is only conceptually true. In memory the same array elements are stored linearly.



- ~~This array can be defined as :~~

~~int a[3][4][2]~~

- As we know C permits only a Row Major arrangement for multi dimensional arrays.
- Let us determine the location of element 9 in the array shown in above figure.
- Element 9 is present at a[2][0][1] indicating that it present in 0th row, 1st column of 2nd 2-D array. Hence address of 9 would be :

$$402 + 2*4*2 + 0*2 + 1 = 402 + 17 = 436$$

- for any 3-D array a[x][y][z] arranged in row major fashion the element a[i][j][k] can be accessed using Base address + i*y*z + j*z + k

- The formula for column Major arrangement would be Base address + i*y*z + k*y + j.
- Similarly for a 4-D array a[w][x][y][z] the element a[i][j][k][l] can be accessed using following formula:

Row Major: Base address + i*x*y*z + j*y*z + k*z + l

Column Major: Base address + i*x*y*z + j*y*z + l*y*k

25 Sunday

Assignment

1. (a) find the location of the element a[1][2][2][1] ~~from~~ from a 4-D integer array a[4][3][4][3] if the base address of the array is 1002.

(b) Write a program to find out the maximum and the second maximum number from an array of integers.

(c) There are two arrays A and B. A contains 25 elements, where as, B contains 30 elements. write a function to create an array c that contains only those elements that are common to A and B.

(d) Write a Program to delete duplicate elements from an array of 20 integers.

(e) A square matrix is symmetric if for all values of i and j a[i][j] = a[j][i]. write a Program, which verifies whether a given 5x5 matrix is symmetric or not.