**Day 09**

**ARRAYS**

ARRAYS are stored in a consequetive memory location

Arrays are homogeneous that means similar datatypes are stored

**syntax:** datatype arrayname[capacity]

**base address + (indexvalue \* size of (datatype))**

**storing**

address of [base address + (indexvalue \* size of (datatype))]

contents of [base address + (indexvalue \* size of (datatype))]

**4 types of arrays**

**static array :** the size of array is known before to the compilation time

**ex:**int arr[5]

**dynamic array :** the size of array is known at run time

**ex:** malloc,calloc,realloc these are defined instdlib.h

**stretchable array :** size of the array is increased or decreased depending on the need for dynamic array

**malloc ,calloc,realloc**

**mutable array :** the size of the array is known or alloted at the time of linking and before the execution.

**/\***

**2D-array demo**

\*/

#include <stdio.h>

#define ROW 2

#define COL 3

int main()

{

int a1[ROW][COL] = {{1,2,3},{4,5,6}};

int a2[ROW][COL] = {1,2,3,4,5,6};

int riv,civ;

printf("\n%d\n",a2[0][2]);

for(riv=0;riv<ROW;riv++)

{

for(civ=0;civ<COL;civ++)

scanf("%d",&a2[riv][civ]);

}

for(riv=0;riv<ROW;riv++)

{

for(civ=0;civ<COL;civ++)

printf("%d ",a2[riv][civ]);

printf("\n");

}

printf("\n\n");

return 0;

}

**Printing odd and even numbers from 51 to103**

#include <stdio.h>

#define CAP 100

int main()

{

int a[CAP], i;

int mid, countOdd,countEven;

int m=51, n=103;

mid = (n-m)/2;

// printf("\nmid=%d",mid);

for(i=m,countOdd = 0, countEven=mid;i<n;i++)

{

if(i%2 != 0)

{

//odd

a[countOdd] = i;

// printf("\na[%d]=%d",countOdd,a[countOdd]);

countOdd++;

}

else

{

//even

a[countEven] = i;

// printf("\na[%d]=%d",countEven,a[countEven]);

countEven++;

}

}

for(i=0;i<countEven;i++)

printf("%d ",a[i]);

printf("\n\n");

return 0;

}

51 53 55 57 59 61 63 65 67 69 71 73 75 77 79 81 83 85 87 89 91 93 95 97 99 101 52 54 56 58 60 62 64 66 68 70 72 74 76 78 80 82 84 86 88 90 92 94 96 98 100 102