Q1. What is array? Explain memory representation of 1-D & 2-D array.

Array - Array is the collection of similar data elements.

declaration of array - data type array name [arr-size];

```
int arr[10];
```

initialization in array - Array में values देने के लिए array declaration के समय पर ही values निर्धारित की जाती है.

ex. `int a[5] = {1,2,3,4,5};`

Types of Array - Array दो प्रकार का होता है

1. 1-D array
2. 2-D array

1-D array - 1-D array में एक ही पंक्ति में values store होती हैं।

Ex- int a[10]; is an example of 1-D array.

Array के सभी elements memory में continously store होते हैं।

int a[5]

a[0]	a[1]	a[2]	a[3]	a[4]
1	2	3	4	5

2-D array - 2-D array में Rows & columns होते हैं।

declaration of 2-D array -

data type array name [No. of rows] [no. of col]

int a[5][5];

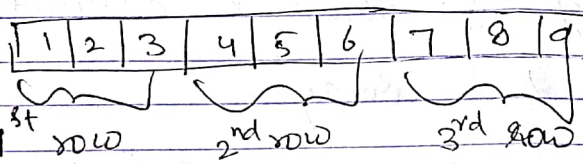
no. of rows, no. of col.

memory rep. of 2-D array- 2-D array को भी memory में continously store किया जाता है। इसके दो तरीके है -

1. Row major ordering
2. column major ordering

Row major - इस method में एरे के सभी elements memory में continously row-wise store होते हैं।

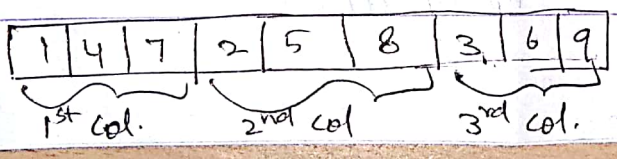
ex. int a[3][3] में यदि $\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$ है तो memory में



store होगा।

Column major - इस method में array के सभी elements column wise store होते हैं।

ex. int a[3][3] यदि $\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$ है तो memory में



store होगा।

Q2. WAP to check wheather a no. is palindrome or not.

```
#include <stdio.h>
#include <conio.h>
void main()
  {
   int n, m, s=0, a;
   printf("Enter any number");
   scanf("%d", &n);
    m=n;
   while(n>0)
     {
        a= n%10;
        s= s*10+a;
        n= n/10;
     }
   if(m==s)
     {
        printf("This is a palindrome no.");
     }
     else
     printf("This is not a palindrome");
   getch();
}
```

Q2: WAP to calculate factorial of a given int. no.

```
#include <stdio.h>
#include <conio.h>
void main()
{
    int n, c;
    long f;
printf("enter any number");
scanf("%d", &n);
    f=1;
for(c=1; c<=n; c++)
    f= f*c;
printf("%d", f);
    getch();
}
```

Q.3. WAP to print this pattern—

```
1
1 2
1 2 3
1 2 3 4
1 2 3 4 5
```

```
#include <stdio.h>
#include <conio.h>
void main()
{
    int i,j;
    for(i=1; i<=5; i++)
    {
        for(j=1; j<=i; j++)
        {
            printf("%d", j);
        }
        printf("\n");
    }
getch();
}
```

Q.3

Program to add two matrices.

```
#include<stdio.h>
#include<conio.h>
void main()
{
int a[3][3], b[3][3], c[3][3], i, j;
printf("Enter value in a matrix");
for(i=0; i<3; i++)
{
for(j=0; j<3; j++)
scanf("%d", &a[i][j]);
}
printf("enter value in b matrix");
for(i=0; i<3; i++)
{
for(j=0; j<3; j++)
scanf("%d", &b[i][j]);
}

for(i=0; i<3; i++)
{
for(j=0; j<3; j++)
{
c[i][j] = a[i][j]+b[i][j];
printf("%d",c[i][j]);
}
printf("\n");
}
getch();
}
```