**ASSIGNMENT** **2:**

1.Write a program to print the following pattern

1

2\*2

3\*3\*3

4\*4\*4\*4

Sol:

#include<stdio.h>

void main()

{

int i,j,n;

printf("\n Enter the value of n:");

scanf("%d",&n);

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

{

for(j=1;j<=i;j++)

{

printf("%d",i);

}

printf("\n");

}

getch();

}

2.Write a program to print the following pattern

1

01

101

0101

10101

Sol:

#include<stdio.h>

#include<conio.h>

void main()

{

int i,j,n;

clrscr();

printf("\n Enter the value of n:");

scanf("%d",&n);

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

{

for(j=1;j<i;j++)

{

if((i+j)%2==0)

{

printf("\t 0");

}

else

{

printf("\t 1");

}

}

printf("\n");

}

getch();

}

3.Write the similarity and difference between array name and a pointer variable?

Sol: Similarity between array and a pointer:

Arrays and pointers are closely related in C. In fact an array declared as int A[10]; can be accessed using its pointer representation. The name of the array A is a constant pointer to the first element of the array.

Differences between array and a pointer:

* An array is a collection of elements of similar data type whereas the pointer is a variable that stores the address of another variable.
* An array size decides the number of variables it can store whereas; a pointer variable can store the address of only one variable in it.
* Arrays can be initialized at the definition, while pointers cannot be initialized at the definition.
* Arrays are static in nature which means once the size of the array is declared, it cannot be resized according to users requirement. Whereas pointers are dynamic in nature, which means the memory allocated can be resized later at any point in time.
* Arrays are allocated at compile time while pointers are allocated at runtime.