1) find the sum of first 10 natural numbers. (Using for loop)

```
#include <stdio.h>
void main()
{
   int  j, sum = 0;

   printf("The first 10 natural number is :\n");

   for (j = 1; j <= 10; j++)
   {
      sum = sum + j;
      printf("%d ",j);
   }
   printf("\nThe Sum is : %d\n", sum);
}
```

2) display the multiplication table of a given integer (Using while loop)

```
#include <stdio.h>
int main()
{
  int n, i;

   printf("Enter a Number ");
   scanf("%d",&n);
   i=1;
   while(i<=10){

      printf("%d * %d = %d \n", n, i, n*i);
      ++i;
   }

   getch();

}
```

3) display the n terms of odd natural number and their sum (Using do…while loop)

```
#include<stdio.h>
```

```
int main()
{
Int i=1, n, sum=0;
Printf(“input number of  terms: “);
Scanf(%d”,&n);
Printf(“\nthe odd numbers are: “);
Do
{
Printf(“%d”,2*i-1);
Sum+=2*i-1;
I++;
}
While(i <=n);
Printf(“\n the bsum of odd natural number upto %d terms : %d \n”,n,sum);
return 0;
}
```

4) display the pattern like right angle triangles. (Using for loop)

```
*
**
***
****
```

```
#include <stdio.h>
void main()
{
  int i,j,rows;
  printf("Input number of rows : ");
  scanf("%d",&rows);
  for(i=1;i<=rows;i++)
  {
          for(j=1;j<=i;j++)
            printf("*");
          printf("\n");
  }
}
```

5) display the pattern like right angle triangles. (Using while loop)

1
2 3
4 5 6
7 8 9 10

```
#include<stdio.h>
int main()
{
int i=1, j=1, k=1;
while(i<=4){
j=1;
while(j<=i){
printf(“%d”,k++);
j++;
}
I++;
Printf(“\n”);
}
Return 0;
}
```

6) make such a pattern like a pyramid with numbers (Using do…while loop)

1
2 3
4 5 6
7 8 9 10

```
#include <stdio.h>
int main(){
   int i=1,j,k,n,t=1,g;
   printf("Enter the value for n:");
   scanf("%d",&n);
   g=n+4-1;
   do
   {
      for(k=g;k>=1;k--){
         printf(" ");
      }
      for(j=1;j<=i;j++)
```

```
            printf("%d",t++);
        printf("\n");
        g--;
        i++;
    }
    while(i<=n);
    return 0;
}
```

7) display Pascal's triangle. (Using for loop)

```
1
 1 1
 1 2 1
 1 3 3 1
1 4 6 4 1
```

```
#include <stdio.h>

void main()
{
    int no_row,c=1,blk,i,j;
    printf("Input number of rows: ");
    scanf("%d",&no_row);
    for(i=0;i<no_row;i++)
    {
        for(blk=1;blk<=no_row-i;blk++)
        printf("  ");
        for(j=0;j<=i;j++)
        {
            if (j==0||i==0)
                c=1;
            else
               c=c*(i-j+1)/j;
            printf("% 4d",c);
        }
        printf("\n");
}
}
```

8) display the first n terms of Fibonacci series. (Using for loop)

```
#include <stdio.h>

void main()
{
  int prv=0,pre=1,trm,i,n;
  printf("Input number of terms to  display : ");
  scanf("%d",&n);
  printf("Here is the Fibonacci series upto  to %d terms : \n",n);
  printf("% 5d % 5d", prv,pre);

 for(i=3;i<=n;i++)
  {
   trm=prv+pre;
   printf("% 5d",trm);
   prv=pre;
   pre=trm;
  }
  printf("\n");
}
```

9) check whether a given number is a perfect number or not. (Using while loop)

```
#include<stdio.h>

int main()
{
   int num, count = 1, sum = 0;

   printf("Enter a number\n");
   scanf("%d", &num);

   while(count < num)
   {
      if(num%count == 0)
      {
         sum = sum + count;
```

```
        }
        count++;
    }

    if(sum == num)
    {
        printf("\n%d is a perfect number\n", num);
    }
    else
    {
        printf("\n%d is not a perfect number\n", num);
    }

    return 0;
}
```

10) find the Armstrong number for a given range of number. (Using while loop)

```
#include <stdio.h>
int main() {
    int num, originalNum, remainder, result = 0;
    printf("Enter a three-digit integer: ");
    scanf("%d", &num);
    originalNum = num;

    while (originalNum != 0) {

        remainder = originalNum % 10;

       result += remainder * remainder * remainder;


      originalNum /= 10;
    }

    if (result == num)
        printf("%d is an Armstrong number.", num);
    else
        printf("%d is not an Armstrong number.", num);
```

```
    return 0;
}
```

11) determine whether a given number is prime or not. (Using do…while loop)

```
#include<stdio.h>
#include<conio.h>
#include<math.h>

void main()
{
    int n, i, flag=0;
    clrscr();

    printf("\n Enter a positive integer value: ");
    scanf("%d",&n);


    do
    {
        if((n!=2) && (n%i==0))
        {
            flag=1;
            break;
        }
        i++;
    }while(i<=sqrt(n));


    if (flag==0)
        printf("\n %d is a prime number.",n);
    else
        printf("\n %d is not a prime number.",n);
    getch();
}
```

12) display the number in reverse order. (Using do…while loop)

```
#include<stdio.h>
#include<conio.h>
void main()
{
   int n,a,r,s=0;
   clrscr();

   printf("\n  Enter The Number:");
   scanf("%d",&n);
   a=n;


   do
   {
      r=n%10;
      s=s*10+r;
      n=n/10;
   }while(n>0);

   printf("\n  The Reverse Number of %d is %d",a,s);
   getch();
}
```

13) display the sum of the series [9 + 99 + 999 + 9999 ...] (Using for loop)

```
#include <stdio.h>

void main()
{   long int n,i,t=9;
          int sum =0;
          printf("Input the number or terms :");
          scanf("%ld",&n);
          for (i=1;i<=n;i++)
          { sum +=t;
            printf("%ld   ",t);
            t=t*10+9;
          }
          printf("\nThe sum of the series = %d \n",sum);
}
```

14) find the sum of the series [1-X^2/2!+X^4/4!- .....]. (Using while loop)

```
#include<stdio.h>
#include<math.h>
int main()
{
float x, sum, t, d;
int i, n;
printf("input the value of x :  ");
printf("input the number of terms :  ");
scanf("%d", &n);
sum=1; t=1;
i=1;
while(i<n){
d = (2*i)*(2*i-1);
t= -t*x*x/d;
sum= sum+t;
i++;
}
Printf("\nsum = %f", sum);
return 0;
}
```

15) find the sum of the series [x - x^3 + x^5 + ...]. (Using do…while loop)

```
#include<stdio.h>
#include<math.h>
int main()
{
```

```
int x, sum, ctr;
int i=1,n, m, mm, nn;
printf(“input the value of x :”);
scanf(“%d”,&x);
printf(“input number of terms : “);
scanf(“%d”,&n);
sum= x; m= -1;
printf(“the values of the series: \n”);
printf(“%d\n”,x);
do{
ctr = (2* i+1);
mm= pow(x, ctr);
nn= mm*m;
printf(“%d \n”,nn);
sum= sum+nn;
m= m* (-1);
i++;
}while(i<n);
printf(“\n the sum =- %d\n”, sum);
return 0;
}
```