**Assignment - 13 A Job Ready Bootcamp in C++, DSA and IOT MySirG**

**More on Recursion in C Language**

1. Write a recursive function to calculate sum of first N natural numbers

Sol – 1.

#include<stdio.h>

#include<conio.h>

#include<math.h>

int sumN(int);

int main()

{

int x;

printf("Enter a number : ");

scanf("%d",&x);

printf("sum of first %d natural numbers is %d",x,sumN(x));

getch();

return 0;

}

int sumN(int n)

{

static int sum=0;

if(n!=0)

{

sum=n+sumN(n-1);

}

return sum;

}

1. Write a recursive function to calculate sum of first N odd natural numbers

Sol – 2.

#include<stdio.h>

#include<conio.h>

#include<math.h>

int sum\_odd(int);

int main()

{

int x;

printf("Enter a number : ");

scanf("%d",&x);

printf("Sum of first %d odd natural numbers is %d",x,sum\_odd(x));

getch();

return 0;

}

int sum\_odd(int n)

{

static int sum=0;

if(n>0)

{

sum=(2\*n-1)+sum\_odd(n-1);

}

return sum;

}

1. Write a recursive function to calculate sum of first N odd natural numbers

Sol – 3.

#include<stdio.h>

#include<conio.h>

#include<math.h>

int sum\_odd(int);

int main()

{

int x;

printf("Enter a number : ");

scanf("%d",&x);

printf("Sum of first %d odd natural numbers is %d",x,sum\_odd(x));

getch();

return 0;

}

int sum\_odd(int n)

{

static int sum=0;

if(n>0)

{

sum=(2\*n-1)+sum\_odd(n-1);

}

return sum;

}

1. Write a recursive function to calculate sum of squares of first n natural numbers

Sol – 4.

#include<stdio.h>

#include<conio.h>

#include<math.h>

int sum\_sqr(int);

int main()

{

int x;

printf("Enter a number : ");

scanf("%d",&x);

printf("Sum of square of first %d natural numbers is %d",x,sum\_sqr(x));

getch();

return 0;

}

int sum\_sqr(int n)

{

static int sum=0;

if(n>0)

{

sum=n\*n+sum\_sqr(n-1);

}

return sum;

}

1. Write a recursive function to calculate sum of digits of a given number

Sol – 5.

#include<stdio.h>

#include<conio.h>

#include<math.h>

int sum\_dig(int);

int main()

{

int x;

printf("Enter a number : ");

scanf("%d",&x);

printf("Sum of digits of %d is %d",x,sum\_dig(x));

getch();

return 0;

}

int sum\_dig(int n)

{

static int sum=0;

if(n>0)

{

sum=n%10+sum\_dig(n/10);

}

return sum;

}

1. Write a recursive function to calculate factorial of a given number

Sol – 6.

#include<stdio.h>

#include<conio.h>

#include<math.h>

int fact(int);

int main()

{

int x;

printf("Enter a number : ");

scanf("%d",&x);

printf("factorial of %d is %d",x,fact(x));

getch();

return 0;

}

int fact(int n)

{

int factorial=1;

if(n>1)

{

factorial=n\*fact(n-1);

}

return factorial;

}

1. Write a recursive function to calculate HCF of two numbers

Sol – 7.

#include<stdio.h>

#include<conio.h>

#include<math.h>

int hcf(int,int);

int main()

{

int x,y;

printf("Enter two numbers : ");

scanf("%d%d",&x,&y);

printf("HCF of %d and %d is %d",x,y,hcf(x,y));

getch();

return 0;

}

int hcf(int x,int y)

{

int c=x>y?y:x;

if(x%y==0||y%x==0)

return c;

if(x>y)

hcf(x%y,y);

else

hcf(x,y%x);

}

1. Write a recursive function to print first N terms of Fibonacci series

Sol – 8.

#include<stdio.h>

#include<conio.h>

#include<math.h>

void fiboN(int);

int main()

{

int x;

printf("Enter a number : ");

scanf("%d",&x);

printf("First %d terms of fibonacci series are : ",x);

fiboN(x);

getch();

return 0;

}

void fiboN(int x)

{

static int a=-1,b=1,c;

c=a+b;

if(x>0)

{

printf("%d ",c);

a=b;

b=c;

fiboN(x-1);

}

}

1. Write a program in C to count the digits of a given number using recursion.

Sol – 9.

#include<stdio.h>

#include<conio.h>

#include<math.h>

void dig(int);

int main()

{

int x;

printf("Enter a number : ");

scanf("%d",&x);

printf("Digits in %d : ",x);

dig(x);

getch();

return 0;

}

void dig(int n)

{

static int count=0;

if(n!=0)

{

count++;

dig(n/10);

}

else

printf("%d",count);

}

1. Write a program in C to calculate the power of any number using recursion.

Sol – 10.

#include<stdio.h>

#include<conio.h>

#include<math.h>

int power(int,int);

int main()

{

int x,a;

printf("Enter a number and power: ");

scanf("%d%d",&x,&a);

printf("%d^%d : %d",x,a,power(x,a));

getch();

return 0;

}

int power(int y,int n)

{

if(y==0)

return 0;

if(n!=0)

{

return y\*power(y,n-1);

}

else

return 1;

}