ASSIGNMENT – 13

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

#include<stdio.h>

int nat\_sum(int num)

{

    if(num)

        return num + nat\_sum(num-1);

}

int main()

{

    int s,sum;

    printf("Enter a number : ");

    scanf("%d",&s);

    sum=nat\_sum(s);

    printf("Sum of first %d Natural numbers is : %d",s,sum);

    return 0;

}

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

#include<stdio.h>

int sum=0;

int nat\_oddsum(int num)

{

    if(num>0)

    {

        nat\_oddsum(num-1);

        if(num%2!=0)

            sum+=num;

    }

    return sum;

}

int main()

{

    int s;

    printf("Enter a number : ");

    scanf("%d",&s);

    sum=nat\_oddsum(2\*s);

    printf("Sum of first %d Odd Natural numbers is : %d",s,sum);

    return 0;

}

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

#include<stdio.h>

int sum=0;

int nat\_evensum(int num)

{

    if(num>0)

    {

        if(num%2==0)

            sum+=num;

        nat\_evensum(num-1);

    }

    return sum;

}

int main()

{

    int s;

    printf("Enter a number : ");

    scanf("%d",&s);

    sum=nat\_evensum(2\*s);

    printf("Sum of first %d Even Natural numbers is : %d",s,sum);

    return 0;

}

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

#include<stdio.h>

int sum=0;

int nat\_squaresum(int num)

{

    if(num>0)

    {

        nat\_squaresum(num-1);

        sum+=num\*num;

    }

    return sum;

}

int main()

{

    int s;

    printf("Enter a number : ");

    scanf("%d",&s);

    sum=nat\_squaresum(s);

    printf("Sum of squares first %d Natural numbers is : %d",s,sum);

    return 0;

}

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

#include<stdio.h>

int sum=0;

int digit\_sum(int num)

{

    if(num>0)

    {

        digit\_sum(num/10);

        num=num%10;

        sum+=num;

    }

    return sum;

}

int main()

{

    int s;

    printf("Enter a number : ");

    scanf("%d",&s);

    sum=digit\_sum(s);

    printf("Sum of digits of %d is : %d",s,sum);

    return 0;

}

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

#include<stdio.h>

int fact(int num)

{

    if(num==0)

        return 1;

    else

        return num \* fact(num-1);

}

int main()

{

    int s;

    printf("Enter a number : ");

    scanf("%d",&s);

    printf("Factorial of %d is : %d",s,fact(s));

    return 0;

}

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

#include<stdio.h>

int gcd(int x, int y)

{

    if(y==0)

        return x;

    else

        return gcd(y,x%y);

}

int main()

{

    int s,m;

    printf("Enter two numbers : ");

    scanf("%d %d",&s,&m);

    printf("HCF of %d and %d is : %d",s,m,gcd(s,m));

    return 0;

}

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

#include<stdio.h>

int fibbo(int n)

{

    if(n==0)

        return(0);

    else if(n==1)

        return 1;

    else

        return fibbo(n-1)+fibbo(n-2);

}

int main()

{

    int s;

    printf("Enter a number : ");

    scanf("%d",&s);

    printf("%d terms of the Fibonacci series are : \n",s);

    for(int i=0;i<s;i++)

        printf("%d ",fibbo(i));

    return 0;

}

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

#include<stdio.h>

int count=0;

int dig\_sum(int n)

{

    if(n)

    {

        dig\_sum(n/10);

        count++;

    }

    return count;

}

int main()

{

    int s;

    printf("Enter a number : ");

    scanf("%d",&s);

    printf("Total number of digits in %d is : %d",s,dig\_sum(s));

    return 0;

}

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

#include<stdio.h>

int power(int n,int m)

{

    if(m==0)

        return 1;

    else

        return n\* power(n,m-1);

}

int main()

{

    int s,m;

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

    scanf("%d %d",&s,&m);

    printf("Value of %d to the power %d is : %d",s,m,power(s,m));

    return 0;

}