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

```c
#include <stdio.h>

int sumNat(int);
int main()
{
    int a;
    printf("Enter the value of a: ");
    scanf("%d", &a);
    printf("%d", sumNat(a));
    return 0;
}
int sum = 0;
int sumNat(int n)
{
    if (n == 1)
        return (sum = sum + n);
    else
    {
        sumNat(n - 1);
        sum += n;
    }
    return sum;
}
```

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

```c
#include <stdio.h>

int sumOdd(int);
int main()
{
    int a;
    printf("Enter the value of a: ");
    scanf("%d", &a);
    printf("sum of first %d odd numbers is: %d", a,
sumOdd(a));
    return 0;
}
int sum = 0;
int sumOdd(int n)
{
    if (n == 1)
        sum += (2 * n) - 1;
    else
    {
        sumOdd(n - 1);
        sum += (2 * n) - 1;
    }
    return sum;
}
```

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

```c
#include <stdio.h>

int sumOdd(int);
int main()
{
    int a;
    printf("Enter the value of a: ");
    scanf("%d", &a);
    printf("sum of first %d odd numbers is: %d", a,
sumOdd(a));
    return 0;
}
int sum = 0;
int sumOdd(int n)
{
    if (n == 1)
        sum += (2 * n) - 1;
    else
    {
        sumOdd(n - 1);
        sum += (2 * n) - 1;
    }
    return sum;
}
```

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

```c
#include <stdio.h>

int sumSqr(int);
int main()
{
    int a;
    printf("Enter the value of a: ");
    scanf("%d", &a);
    printf("sum of squares of first %d natural numbers
is: %d", a, sumSqr(a));
    return 0;
}
int sum = 0;
int sumSqr(int n)
{
    if (n == 1)
        sum += n * n;
    else
    {
        sumSqr(n - 1);
        sum += n * n;
    }
    return sum;
}
```

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

```c
#include <stdio.h>

int sumOfDigits(int);
int main()
{
    int a;
    printf("Enter the value of a: ");
    scanf("%d", &a);
    printf("sum of digits of given numbers %d is: %d", a,
sumOfDigits(a));
    return 0;
}
int sum = 0;
int sumOfDigits(int n)
{
    if (n >= 1)
    {
        sum += (n % 10);
        sumOfDigits(n / 10);
    }

    else
        return sum;
    return sum;
}
```

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

```c
#include <stdio.h>

int fact(int);
int main()
{
    int a;
    printf("Enter the value of a: ");
    scanf("%d", &a);
    printf("Factorial of %d is: %d", a, fact(a));
    return 0;
}
int res = 1;
int fact(int n)
{
    if (n == 1)
        return 1;

    else
    {
        fact(n - 1);
        res = res * n;
    }
    return res;
}
```

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

```c
#include <stdio.h>

int hcf(int, int);
int main()
{
    int a, b;
    printf("Enter the value of a and b: ");
    scanf("%d %d", &a, &b);
    printf("\nHCf of %d and %d is %d", a, b, hcf(a, b));
    return 0;
}

int n, res = 0;
int hcf(int a, int b)
{
    n = b % a;
    if (n == 0)
    {
        res = a;
    }
    else
    {
        b = a;
        a = n;
        hcf(a, b);
    }
    return res;
}
```

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

```c
#include <stdio.h>

int fibonacci(int);
int main()
{
    int n, m = 0, i;
    printf("Enter Total terms: ");
    scanf("%d", &n);
    printf("Fibonacci series terms are: ");
    for (i = 1; i <= n; i++)
    {
        printf("%d ", fibonacci(m));
        m++;
    }
    return 0;
}
int fibonacci(int n)
{
    if (n == 0 || n == 1)
        return n;
    else
        return (fibonacci(n - 1) + fibonacci(n - 2));
}
```

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

```c
#include <stdio.h>

int countDigit(int);
int main()
```

```
{
    int x;
    printf("\nEnter a number: ");
    scanf("%d", &x);
    printf("\nCount of number is : %d", countDigit(x));
    return 0;
}

int count = 0;
int countDigit(int n)
{
    if (n > 0)
    {
        count++;
        countDigit(n / 10);
    }
    else
    {
        return count;
    }

    return count;
}
```

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

```
#include <stdio.h>

int power(int, int);

int main()
```

```
{
    int base, exponent;
    printf("Enter the base\n");
    scanf("%d", &base);
    printf("Enter the exponent\n");
    scanf("%d", &exponent);
    printf("Power(%d^%d) = %d\n", base, exponent,
power(base, exponent));
    return 0;
}

int power(int b, int e)
{
    if (e == 0)
        return 1;
    else
        return (b * power(b, e - 1));
}
```