Assignment - 11
A Job Ready Bootcamp in C++, DSA and IOT MySirG
More on functions in C Language

Submitted By;
Vishal Shaw
bkr.vishalshaw@gmail.com

—----------

1. Write a function to calculate LCM of two numbers. (TSRS)

```c
#include <stdio.h>

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

int lcm(int n, int n1)
{
    int i = 1;
    while (1)
    {
        if (i % n == 0 && i % n1 == 0)
        {
            break;
        }
        else
            i++;
    }
```

```
    return i;
}
```

2. Write a function to calculate HCF of two numbers. (TSRS)

```
#include <stdio.h>

int hcf(int, int);
int main()
{
    int a, b;
    printf("Enter values of a and b: ");
    scanf("%d %d", &a, &b);
    printf("H.C.F of %d and %d is: %d", a, b, hcf(a, b));
    return 0;
}

int hcf(int n, int n1)
{
    int n3;
    while (1)
    {
        n3 = n % n1;
        if (n3 == 0)
            break;
        else
        {
            n = n1;
            n1 = n3;
        }
    }
    return n1;
}
```

3. Write a function to check whether a given number is Prime or not. (TSRS)

```c
#include <stdio.h>

int chekPrime(int);
int main()
{
    int a;
    printf("Enter a number: ");
    scanf("%d", &a);
    if (a == chekPrime(a))
    {
        printf("\n%d is Prime", a);
    }
    else
    {
        printf("\n%d is not Prime", a);
    }

    return 0;
}

int chekPrime(int n)
{
    int i;
    for (i = 2; i < n; i++)
    {
        if (n % i == 0)
            break;
    }

    return i;
}
```

4. Write a function to find the next prime number of a given number. (TSRS)

```c
#include <stdio.h>

int nextPrime(int);
int main()
{
    int a;
    printf("Enter a number: ");
    scanf("%d", &a);
    a = a + 1;
    printf("\nnext prime number is: %d", nextPrime((a)));
    return 0;
}

int nextPrime(int n)
{
    int flag = 1, i;
    while (flag)
    {
        for (i = 2; i < n; i++)
        {
            if (n % i == 0)
            {
                break;
            }
        }
        (n == i) ? flag = 0 : n++;
    }
    return n;
}
```

5. Write a function to print first N prime numbers (TSRN)

```c
#include <stdio.h>

void allPrime(int);
int main()
{
    int n1;
    printf("Enter end range value: ");
    scanf("%d", &n1);
    allPrime(n1);
    return 0;
}

void allPrime(int n1)
{
    int i, n = 2;
    while (n1 > 1)
    {
        for (i = 2; i < n; i++)
        {
            if (n % i == 0)
                break;
        }
        if (n == i)
            printf("%d ", n);
        n++;
        --n1;
    }
}
```

6. Write a function to print all Prime numbers between two given numbers. (TSRN)

```c
#include <stdio.h>

void allPrime(int, int);
int main()
{
    int n, n1;
    printf("Enter start and end range value: \n");
    scanf("%d %d", &n, &n1);
    printf("prime numbers between %d and %d are: \n", n,
n1);
    allPrime(n, n1);
    return 0;
}

void allPrime(int n, int n1)
{
    int i;
    while (n1 > n)
    {
        n++;
        for (i = 2; i < n; i++)
        {
            if (n % i == 0)
                break;
        }
        if (n == i)
            printf("%d ", n);
    }
}
```

7. Write a function to print first N terms of Fibonacci series (TSRN)

```c
#include <stdio.h>

void fib(int, int, int);
int main()
{
    int f0 = 0, f1 = 1, n;
    printf("\nEnter value of n: ");
    scanf("%d", &n);
    fib(f0, f1, n);
    return 0;
}

void fib(int f0, int f1, int n)
{
    int f;
    printf("%d %d ", f0, f1);
    for (int i = 1; i <= n - 2; i++)
    {
        f = f0 + f1;
        f0 = f1;
        f1 = f;
        printf("%d ", f);
    }
}
```

8. Write a function to print PASCAL Triangle. (TSRN)

```c
#include <stdio.h>

int fact(int);
int combi(int, int);
void printPascal(int);

int main()
{
    int x;
    printf("Enter a number: ");
    scanf("%d", &x);
    printPascal(x);
    return 0;
}

int fact(int n)
{
    if (n == 0)
        return 1;
    return (n * fact(n - 1));
}

int combi(int n, int r)
{
    return (fact(n) / fact(n - r) / fact(r));
}

void printPascal(int line)
{
    int i, j, k = 1, r;
    for (i = 1; i <= line; i++)
```

```
    {
        k = 1;
        r = 0;
        for (j = 1; j <= 2 * line - 1; j++)
        {
            if ((j >= line + 1 - i) && (j <= line - 1 +
i) && k)
            {
                printf("%d", combi(i - 1, r));
                k = 0;
                r++;
            }
            else
            {
                printf(" ");
                k = 1;
            }
        }
        printf("\n");
    }
}
```

9. Write a program in C to find the square of any number using the function.

```
#include <stdio.h>

int sq(int);
```

```c
int main()
{
    int n = 1;
    do
    {
        int a;
        printf("Enter a number to get square of it: \n");
        scanf("%d", &a);
        printf("Square of %d is: %d", a, sq(a));
        printf("\nDo you want to check again [0-no]: ");
        scanf("%d", &n);
    } while (n);
    return 0;
}

int sq(int n)
{
    return (n * n);
}
```

10. Write a program in C to find the sum of the series 1! /1+2!/2+3!/3+4!/4+5!/5 using the function.

```c
#include <stdio.h>

int sumSeries(int);
int series(int);
int main()
{
    int n;
    printf("Enter value of n: ");
    scanf("%d", &n);
```

```
    printf("Sum of series 1!/1 to %d!/%d is: %d", n, n,
sumSeries(n));
    return 0;
}

int series(int n)
{
    int res = 1, i = 1;
    while (i <= n)
    {
        for (i = 1; i <= n; i++)
        {
            res = res * i;
        }
    }
    return res;
}

int sumSeries(int n)
{
    int a = 1, var = 0;
    for (int i = 1; i <= n; i++)
    {
        var = var + (series(i) / i);
    }
    return var;
}
```