Assignment - 7
A Job Ready Bootcamp in C++, DSA and IOT MySirG
Iterative Control Statements (Part - 2)

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

---

1. Write a program to find the Nth term of the Fibonnaci series.

```c
#include<stdio.h>
int main(){
    int f0=0, f1=1, fn=0, n;
    printf("Enter the nth term: ");
    scanf("%d", &n);
    if(n<=1)
        return n;
    for(int i=1; i<=n-1; i++){
        fn = f1+f0;
        f0=f1;
        f1=fn;
    }
    printf("nth term is: %d", fn);
    return 0;
}
```

2. Write a program to print first N terms of Fibonacci series

```c
#include<stdio.h>
int main(){
    int f0=0, f1=1, fn=0, n;
    printf("Enter the nth term: ");
    scanf("%d", &n);
    if(n<=1)
        return n;
    printf("%d ",f0);
    for(int i=1; i<=n-1; i++){
```

```
        fn = f1+f0;
        f0=f1;
        f1=fn;
        printf("%d ", f0);
    }
    return 0;
}
```

3. Write a program to check whether a given number is there in the Fibonacci series or not.

```
#include<stdio.h>
#include<math.h>
int main(){
    int x=0,a,b,n;
    printf("Enter a number: ");
    scanf("%d", &x);
    a=(5*(x*x)+4);
    b=(5*(x*x)-4);
    n=a>b?a:b;
    int flag=0;
    for(int i=1; i<=sqrt(n)+1; i++){
    if(a==(i*i) || b==(i*i)){
        flag = 1;
        break;
        }
    }
    if(flag == 1)
        printf("%d is in fibo series", x);
    else
        printf("%d is not in fibo series", x);
    return 0;
}
```

4. Write a program to calculate HCF of two numbers

```
#include<stdio.h>
int main(){
    int a,b,x;
    printf("\nConsidering b>=a\nEnter the two numbers { b
a }: ");
    scanf("%d %d", &a,&b);
    while(1){
        x=b%a;
        if(x==0){
            printf("%d", a);
            break;
            }
        else{
            b=a;
            a=x;
        }
    }
    return 0;
}
```

5. Write a program to check whether two given numbers are co-prime numbers or not

```
#include<stdio.h>
int main(){
    int a,b,x;
    printf("\nConsidering b>=a\nEnter the two numbers { b
a }: ");
    scanf("%d %d", &b, &a);
    while(1){
        x=b%a;
        if(x==0){
```

```
            break;
            }
        else{
            b=a;
            a=x;
        }
    }
    if (a==1)
        printf("\nco primes");
    else
        printf("\nNot co primes");
    return 0;
}
```

6. Write a program to print all Prime numbers under 100

```
#include<stdio.h>
int main(){
    int n=100, x=2, i;
    while(n){
        for(i=2; i<x; i++){
            if(x%i == 0)
                break;
        }
        if(x==i){
            printf("%d ", x);
            n--;
        }
        x++;
    }

}
```

7. Write a program to print all Prime numbers between two given numbers

```c
#include <stdio.h>
int main()
{
    int n, n1, i;
    printf("\nEnter the range of two numbers: ");
    scanf("%d %d", &n, &n1);
    while (n <= n1)
    {
        for (i = 2; i < n; i++)
        {
            if (n % i == 0)
                break;
        }
        if (n == i)
        {
            printf("%d ", n);
        }
        n++;
    }
}
```

8. Write a program to find next Prime number of a given number

```c
#include <stdio.h>
int main()
{
    int n, i, flag = 1;
    printf("\nEnter number to find next prime: ");
    scanf("%d", &n);
    n = n + 1;
    while (flag)
    {
```

```
        for (i = 2; i < n; i++)
        {
            if (n % i == 0)
                break;
        }
        if (n == i)
        {
            printf("%d ", n);
            flag = 0;
        }
        n++;
    }
}
```

9. Write a program to check whether a given number is an Armstrong number or not

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

int main()
{
    int n, t, dig, res = 0, count;
    printf("Enter a number to check Armstrong or not:");
    scanf("%d", &n);
    t = n;
    count = printf("%d", n);
    while (t > 0)
    {
        dig = t % 10;
        res = res + pow(dig, count);
        t = t / 10;
    }
```

```
    if (n == res)
        printf("\n%d is Armstrong Number", n);
    else
        printf("\n%d is not an Armstrong number", n);
    return 0;
}
```

10. Write a program to print all Armstrong numbers under 1000

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

int main()
{
    int n, res, dig;
    for (int i = 1; i <= 1000; i++)
    {
        n = i;
        res = 0;
        while (n > 0)
        {
            dig = n % 10;
            res += (dig * dig * dig);
            n = n / 10;
        }
        if (i == res)
            printf("%d\n", i);
    }
    return 0;
}
```