Assignment - 7

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

```
#include<stdio.h>
int main()
   {
      int n, t1 = 0, t2 = 1, nextTerm = 0, i;
      printf("Enter the n value: ");
      scanf("%d", &n);
      if(n == 0 || n == 1)
         printf("%d ", n);
      else
         nextTerm = t1 + t2;
      for (i = 3; i <= n; ++i)
         {
            t1 = t2;
            t2 = nextTerm;
            nextTerm = t1 + t2;
         }
      printf("%d", t2);
}
```

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

```
#include<stdio.h>
int main()
{

  int i, n;
  int t1 = 0, t2 = 1;
  int nextTerm = t1 + t2;
```

```
  printf("Enter the number of terms: ");
  scanf("%d", &n);
  printf("Fibonacci Series: %d, %d, ", t1, t2);

  for (i = 3; i <= n; ++i)
    {
        printf("%d, ", nextTerm);
        t1 = t2;
        t2 = nextTerm;
        nextTerm = t1 + t2;
    }

  return 0;
}
```

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

```
                #include<stdio.h>

int main()
{
    int f1,f2,f3,n;
    printf("Enter any number ");
    scanf("%d",&n);
    f1=0;
    f2=1;
    if(n==0)
        printf("Fibboncci number");
    f3=f1+f2;
    while(f3<n)
    {
```

```
        f1=f2;
        f2=f3;
        f3=f1+f2;

    }
    if(f3==n)
        printf("Fibbonacci series");
    else
        printf("Not fibbanacci number");

   return 0;
}
```

4. Write a program to calculate HCF of two numbers

```
#include<stdio.h>

int main()
{
    int num1 = 36, num2 = 60, hcf = 1;

    for(int i = 1; i <= num1 || i <= num2; i++) {
        if(num1 % i == 0 && num2 % i == 0)
            hcf = i;
    }

    printf("The HCF: %d", hcf);

    return 0;
}
```

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

```
#include<stdio.h>

int main()
{
	int num1, num2, hcf, i;
	printf("Enter two numbers:\n");
	scanf("%d%d", &num1, &num2);

	for(i=1;i<=num1;i++)
	{
	 if(num1%i==0 && num2%i==0)
	 {
	  hcf = i;
	 }
	}

	if(hcf == 1)
	{
	 printf("%d and %d are CO-PRIME NUMBERS.", num1, num2);
	}
	else
	{
	 printf("%d and %d are NOT CO-PRIME NUMBERS.", num1,
num2);
	}
	return 0;
}
```

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

```
#include <stdio.h>
int main()
{
    int i, num, count;
    printf("Prime Numbers 1 To 100 -: \n");
    for (num = 1; num <= 100; num++){
        count = 0;
        for (i = 2; i <= num/2; i++){
            if (num % i == 0){
                count++;
                break;
            }
        }

        if (count == 0 && num != 1)
            {
                printf("%d ", num);
            }
    }
    return 0;
}
```

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

```
#include<stdio.h>

int main()
{
    int lwr,upr,x,i;
```

```
    printf("Enter two numbers : ");
    scanf("%d %d",&lwr,&upr);
    for(x = lwr+1;x<upr-1;x++)
    {
        for(i=2;i<x;i++)
            if(x%i==0)
            break;

    if(i==x)
        printf("%d ",x);
    }
    return 0;
}
```

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

```
                #include<stdio.h>

int main()
{
    int i,j,n,count=0 ;
    printf("Enter the number ");
    scanf("%d",&n);
    for(i=n;i>0;i++)
    {
        count=0;
        for(j=1;j<=i;j++)
        {
            if(i%j==0)
                count++;

        }
```

```
        if(count==2)
        {
            printf("%d ",i);
            break;
        }
    }
    return 0;
}
```

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

```
                #include<stdio.h>
int main()
{
    int n,arm = 0,r,c;
    printf("Enter any number ");
    scanf("%d",&n);
    c=n;
    while(n>0)
    {
        r=n%10;
        arm = (r*r*r)+arm;
        n=n/10;
    }
    if(c==arm)
        printf("Armstrong Number ");
    else
        printf("Not Armstrong Number ");

    return 0;
}
```

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

```
#include<stdio.h>

int main()
{
    int n,r,x,s;
    printf("Armstrong Number = ");
    for(n=1;n<=1000;n++)
    {
        s=0;
        x = n;
        while(x!=0)
        {
            r= x%10;
            s=s+r*r*r;
            x = x/10;

        }
        if(s==n)
            printf("%d \n",n);
    }

    return 0;
}
```