Loop 2

1. Take a number from the user and calculate its factorial.

INPUT	OUTPUT
5	5!=120
0	0!=1
1	1!=1

/* Take two number from user and calculat it's factoriul */

```cpp
#include <iostream>
using namespace std;

int main ()
{
    int i,n,sum;
    sum=1;              /* In the case of multification must use 1 not 0 */
    cout<<"Enter a number :";
    cin>>n;

    for(i=1; i<=n; i=i+1)      /* there is same case i=1 */
    {
        sum=sum*i;
    }

    cout<<"n! = "<<sum;

    return 0;
}
```

2. Take the value of x and y from the user and calculate x^y without using the pow() function (x can be a decimal number but y must be a positive integer).

INPUT		OUTPUT
2.5	5	97.65625
10	0	1
2	10	1024
-3	3	-27

/* Take the value of x and y from user and calculat x^y with out using prower function.
x can be decimal number and y must be positive integer */

```cpp
#include <iostream>
using namespace std;

int main()
{
    int i,y;
    float x,sum;
    sum=1;                    /* In the case of multification must use 1 not 0 */

    cout<<"Enter x :";
    cin>>x;
    cout<<"Enter y :";
    cin>>y;

    for(i=1; i<=y; i=i+1)     /* there is same case i=1 */
    {
        sum=sum*x;

    }

    cout<<sum;

    return 0;
}
```

3. Take two integers from the user and print their highest common factor (HCF).

INPUT		OUTPUT
75	100	25
50	49	1
121	99	11

```cpp
/* Take two integer number from user and print the hight common factor (HCF) */

#include <iostream>
using namespace std;

int main()
{
   int i,x,y,sum,swap;
      sum=1;

      cout<<"Enter 1.Number :";
      cin>>x;
      cout<<"Enter 2.Number :";
      cin>>y;

   if(x > y )
   {
      swap=x;
      x=y;                        /* if user enter big number in x that's why we use this */
      y=swap;
   }

   for(i=x; i>0; i=i-1)           /* we want HFC that's why start from x and i=i-1 */
   {
      if((x%i==0) && (y%i==0))    /* here we search i which divisibal by x and y */
      {
         cout<<"HCF ( "<<x<<", "<<y<<" )="<<i<<endl;
         break;
      }

   }

   return 0;
}
```

4. Take two integers from the user and print their lowest common multiple (LCM).

INPUT		OUTPUT
15	10	30
13	5	65
33	99	99

```cpp
/* Take two integer number from user and print the lowest common factor (LCF) */

#include <iostream>
using namespace std;

int main()
{
    int i,x,y,sum,swap;
        sum=1;

        cout<<"Enter 1.Number :";
        cin>>x;
        cout<<"Enter 2.Number :";
        cin>>y;

    if(x > y )
    {
        swap=x;
        x=y;                    /* if user enter big number in x that's why we use this */
        y=swap;
    }

    for(i=y; i<=x*y ; i=i+1)
    {
        if((i%x==0) && (i%y==0))        /* here we search x and y which divisibal by i */
        {
            cout<<"LCF ( "<<x<<", "<<y<<" )="<<i<<endl;
            break;
        }

    }

    return 0;
}
```

5. Take a number from the user and check if it is a prime number. A prime number is a positive integer divisible by exactly two numbers, 1 and the number itself.

INPUT	OUTPUT
5	Prime
9	Not prime
121	Not prime
1	Not prime
17	Prime

```cpp
/* Take a number from user and chack it is a prime number */

#include <iostream>
using namespace std;
int
main ()
{
  int n, i, isprime;
  isprime = 1;            /* dore isprime=1 hoca prime */

  cout << "Enter a Number :";
  cin >> n;

  for (i = 2; i < n; i = i + 1)
    {
      if (n % i == 0)            /* check kora hoca bag jaikina jodi jai tahola isprime=0 hoba  */
        {
          isprime = 0;     /* isprime=0 hoca ata not prime */
          break;
        }

    }
  if (isprime == 1)
    {
      cout << n << " prime";
    }
  else
    cout << n << " not prime";

  return 0;
}
```

6. Take a number from the user and check if it is a perfect number. A perfect number is a positive integer that is equal to the sum of its positive divisors excluding the number itself. 28 is a perfect number because the sum of its divisors 1+2+4+7+14 is equal to 28.

INPUT	OUTPUT
6	Perfect
10	Not perfect
496	Perfect

```cpp
/* Take a number from user and chack if it is a perfect number */

#include<iostream>
using namespace std;
int main ()
{
   int i, num, x, sum=0;

   cout << "Enter the number : ";
   cin >> num;

   for (i=1; i < num; i=i+1)
   {
      x = num % i;
      if (x == 0)
      {
         sum = sum + i;
      }
   }
   if (sum == num)
      cout<< num <<" perfect.";
   else
      cout<< num <<"  not perfect.";
   return 0;
}
```

7. Take a number from the user and check if it is an Armstrong number. An Armstrong number is an integer such that the sum of the cubes of its digits is equal to the number itself. 153 is an Armstrong number because $1^3+5^3+3^3$ is equal to 153.

INPUT	OUTPUT
0	Armstrong number
12	Not an Armstrong number
370	Armstrong number
100	Not an Armstrong number

```cpp
/* Take a number from user and chack if it is a armstorng number */

#include <iostream>
using namespace std;

int main(){

  int n,r,sum,temp;
  sum=0;
   cout<<"Enter the Number=  ";
   cin>>n;

  temp=n;

  while(n>0)       /* 153 = (1*1*1)+(5*5*5)+(3*3*3) */
  {
     r=n%10;
     n=n/10;
     sum=sum+(r*r*r);
  }
  if(temp==sum)
   cout<<"Armstrong Number.";
  else
   cout<<"Not Armstrong Number.";
return 0;
}
```

10. Take the number of terms, n, from the user and display the following series up to the nth term and its sum.

$$1+2+4+7 \cdots + n^{th} \text{ term} = ?$$

INPUT	OUTPUT
5	1+2+4+7+11=25
8	1+2+4+7+11+16+22+29=92

/* Take the number of turms n from user and disolay the flowing sris 1+2+4+7+11+..+n turms=? */

```cpp
#include<iostream>
using namespace std;

int main()
{
    int n, i, x=1, sum=0;

    cout<<"Enter a Number :";
    cin>>n;

    for(i=0; i<n; i=i+1)        /* i=0 karon 0 thaka start korlam and x=1  */
    {

        x=x + i;                        /* sum korar por x print detace x=1+0 =1 */
        cout<<x;                         // x=1+1 =2 // x=2+2 =4 // x=4+3 =7

        sum=sum+x;                  /* x ka sum kora hoca */

        if(i+1==n)                     /* i ar man jokon n ar shoman hoba thokon = print hoba */
        {
            cout<<" = ";
        }
        else
        {
            cout<<" + ";
        }
    }

    cout<<sum;

    return 0;
}
```

11. Takes the number of terms, n, from the user and display the following series up to the nth term and its sum.

$$1-2+3-4\cdots \pm n^{th}\text{ term} = ?$$

INPUT	OUTPUT
10	1-2+3-4+5-6+7-8+9-10=-5
3	1-2+3=2

/* Take the number of turms n from user and disolay the flowing sris 1-2+3-4+5-..+n turms=? */

```cpp
#include<iostream>
using namespace std;

int main(){
    int n, sum=0, a=0, b=0, i;

    cout<<"Enter a Number :";
    cin>>n;

    for(i=1;i<=n;i=i+1)
    {
        if(i%2==0)              // if i=1,2,3,4
        {
            cout<<i<<"+";       // 2+4+
            a=a+i;              // a=0+2+4=6
        }
        else
        {
            cout<<i<<"-";       // 1-3-
            b=b+i;              // b=0+1+3=4

        }
        if(a>b)
        {
            sum=a-b;            // sum= 6-4 = 2
        }
        else
        {
            sum=b-a;
        }
    }

    cout<<"\b="<<sum;       // cout sum
    return 0;}
```

12. Write a program that takes the number of terms, n, from the user and prints the first n terms of the Fibonacci sequence. In the Fibonacci sequence, any term is equal to the sum of the previous two terms (except for the first two terms).

0, 1, 1, 2, 3, 5, 8, 13, 21…

INPUT	OUTPUT
5	0, 1, 1, 2, 3
11	0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55

/* Take a number from user n and display the following fibonacci sequence like 0 1 1 2 3 5 8 13 21 */

```cpp
#include <iostream>
using namespace std;
int
main ()
{
  int n, i, f1, f2, f3;
  f1 = 0;
  f2 = 1;

  cout << "Enter a Number :";
  cin >> n;

  cout<<f1<<" "<<f2<<" ";

  for (i = 3; i <= n; i = i + 1)
  {
    f3 = f1 + f2;                  /* f3=0+1 =1 */

    cout << f3 << " ";

    f1 = f2;                       /* f1=1 here f2 is convat f1 */
    f2 = f3;                       /* f2=1 here f3 is convat f2 */

  }

  return 0;
}
```

13. Take a number from the user and check if it is a term in the Fibonacci sequence. If it is one of the terms in the Fibonacci sequence, print the term number.

INPUT	OUTPUT
8	7th term of the Fibonacci sequence
15	Does not exist in the Fibonacci sequence
55	11th term of the Fibonacci sequence

```cpp
/* Take number from user and check if it's a term in the fibonacci sequence.
if it is one of the terms in the fibonacci sequence, print the term number */

#include<iostream>
using namespace std;

int main ()
{
    int a, b, c, num;

    cout<<"Enter any number: ";
    cin>>num;

    if ((num == 0) || (num == 1)){          /* if num = 3 */
        cout<<num<<" term of the Fibonacci sequence";
    }
    else
        a = 0;
        b = 1;
        c = a + b;                          /* c=0+1 =1 */
        while (c < num)                     /* 1<3 */
            {
            a = b;                          /* a=1 b=1 c=2 */
            b = c;
            c = a + b;
        }
    if (c == num) {
        cout<<num<<" term of the Fibonacci sequence";
    }
     else

        cout<<num<<" Does not exist in the Fibonacci sequence";

    return 0;

}
```

14. Take a number from the user and display its divisors.

INPUT	OUTPUT
12	1,2,3,4,6,12
23	1,23

/* Take number from user and display it's divisore */

```cpp
#include <iostream>
using namespace std;

int main()
{
  int i, n;

  cout<<"Enter a Number :";
  cin>>n;

  for(i=1; i<n; i=i+1)
  {
    if(n % i == 0)          /* find what is divisore if n=12 than 12%1=0   12/2=0 than it's cout i */
    {
      cout<<i<<" , ";
    }
  }
  cout<<i;
  return 0;
}
```

15. Write a program that asks the user to enter how many inputs to take, then takes the specified number of inputs and counts the number of positive, negative, zero, odd and even numbers.

INPUT	OUTPUT
Number of inputs: 5 Enter 5 numbers: -7 0 4 7 0	Positive: 2 Negative: 1 Zero: 2 Odd: 2 Even: 3
Number of inputs: 10 Enter 10 numbers: 6 23 0 -1 88 5 -10 31 2 0	Positive: 6 Negative: 2 Zero: 2 Odd: 4 Even: 6

/* write a program that aske the user how many input take, than take the specified number of input and count the number of positive , negative , x=zero , odd and even number. */

```cpp
#include <iostream>
using namespace std;

int main()
{
   int i,x,m,p,n,z,o,e;
   p=0;
   z=0;
   n=0;
   e=0;
   o=0;

       cout<<"Number of input : ";
       cin>>m;

   for(i=0; i<m; i=i+1)
   {
      cout<<"Enter  a integer ["<<(i+1)<<"] = ";
      cin>>x;

       if (x>0) {
           p=p+1;          /* count positive Number */
       }
       else if(x<0){
          n=n+1;          /* count Negative Number */
       }
        else if (x==0)
        {
           z=z+1;        /* count Zero */
        }
```

```cpp
        if(x%2==0) {
            e=e+1;          /* count Even */
        }
        else
            o=o+1;          /* count Odd */
    }

        cout<<" positive : "<<p<<endl;
        cout<<" Negative : "<<n<<endl;
        cout<<" Even : "<<e<<endl;
        cout<<" Zero : "<<z<<endl;
        cout<<" Odd : "<<o<<endl;
    return 0;
}
```