In [1]:

```
import math

def is_natural(n):                      #checks if n is natural
        try:
            int(n)
            a=True
        except ValueError:
            a=False

        if a==True and int(n)>0:
            return (int(n))
        else:
            print("Input is not a Natural number.")
            return (1)
```

In [2]:

```
def is_float(n):                       #function to return n as float
    try:
        float(n)
        a = True
    except:
        a = False

    if a==True:
        return(float(n))
    else:
        print("The input is not as desired.")
        return (1)
```

In [3]:

```
def sum_natural_numbers(n):                         # finds sum of n natural numbers using recursion
    if n <= 1:
        return n
    return n +  sum_natural_numbers(n - 1)


    print("The sum of first "+ str(n) + " natural numbers is " + str(sum_natural_numbers(n)))


def sum_odd_numbers(n):                            # function for sum of first n odd numbers

    sum=0
    i=1
    j=1
    while i<=n:
        if j%2!=0:
            sum=sum+j
            i+=1
            j+=1
        else:
            j+=1
    print("The sum of first " + str(n) + " odd numbers is " + str(sum))
```

In [4]:

```
def sum_numbers_AP(a,n):                          # function for sum of n terms of an AP
    d=1.5                                          # with first term and number of terms taken as input
    sum=0
    for i in range(n):
        sum=sum+a
        a=a+d
    print("\nThe sum of first " + str(n) + " terms of an AP is " + str(sum))




def sum_numbers_GP(a,n):                               # function for sum of n terms of a GP
    r=0.5                                              # with first term and number of terms taken as input
    sum=0
    for i in range(n):
        sum=sum+a
        a=a*r
    print("\nThe sum of first " + str(n) + " terms of an GP is " + str(sum))




def sum_numbers_HP(a,n):                              # function for sum of n terms of a HP
    r=1.5                                             # with first term and number of terms taken as input
    sum=0
    for i in range(n):
        sum=sum+1/a
        a=a+r
    print("\nThe sum of first " + str(n) + " terms of an HP is " + str(sum))
```

In [5]:

```
def factorial(n):                                   # function for finding factorial of a number
    f=1
    while n>0:
        f=f*n
        n-=1
    return f
```

In [6]:

```
# sine function

def sin_func(x,n):                          # with argument of sine and number of terms in its taylor expansion taken as input
    sum=0
    for i in range(1,n+1):                              # starting the index with i=1 because factorial of -1 is not defined
        t=(-1)**(i-1) * (x**(2*i-1)/factorial(2*i-1))  # taylor series terms
        sum=sum+t
    return sum


# exponential function

def exp_func(x,n):                          # with argument of sine and number of terms in its taylor expansion taken as input
    sum=0
    for i in range(0,n):
        t=(-1)**i * x**i/factorial(i)        # taylor series terms
        sum=sum+t
    return sum
```

In [ ]:

In [ ]: