Question 1

Sum of first N natural numbers

In [99]:
```
from elementaryLibrary import sumN

#The following code is just to take care of any exceptional input.

N = float(input("Enter any natural number: "))
if N <= 0:
    print(str(float(N)) + " is not a natural number!")
elif N.is_integer():
    sumN(int(N))
else:
    print(str(N) + " is not a natural number!")
```

```
Enter any natural number: 10
55 is the sum of first 10 natural numbers.
```

In [100…
```
from elementaryLibrary import sumN

#The following code is just to take care of any exceptional input.

N = float(input("Enter any natural number: "))
if N <= 0:
    print(str(float(N)) + " is not a natural number!")
elif N.is_integer():
    sumN(int(N))
else:
    print(str(N) + " is not a natural number!")
```

```
Enter any natural number: 23
276 is the sum of first 23 natural numbers.
```

In [108…
```
from elementaryLibrary import sumN

#The following code is just to take care of any exceptional input.

N = float(input("Enter any natural number: "))
if N <= 0:
    print(str(float(N)) + " is not a natural number!")
elif N.is_integer():
    sumN(int(N))
else:
    print(str(N) + " is not a natural number!")
```

```
Enter any natural number: 12
78 is the sum of first 12 natural numbers.
```

In [106…
```
from elementaryLibrary import sumN

#The following code is just to take care of any exceptional input.

N = float(input("Enter any natural number: "))
if N <= 0:
    print(str(float(N)) + " is not a natural number!")
elif N.is_integer():
    sumN(int(N))
else:
    print(str(N) + " is not a natural number!")
```

```
Enter any natural number: -4
-4.0 is not a natural number!
```

In [105…
```
from elementaryLibrary import sumN

#The following code is just to take care of any exceptional input.

N = float(input("Enter any natural number: "))
if N <= 0:
    print(str(float(N)) + " is not a natural number!")
elif N.is_integer():
    sumN(int(N))
else:
    print(str(N) + " is not a natural number!")
```

```
Enter any natural number: 3.4
3.4 is not a natural number!
```

Sum of first N odd numbers

In [1]:
```
from elementaryLibrary import sumOdd

#The following code is just to take care of any exceptional input.

N = float(input("Enter any natural number: "))
if N <= 0:
    print(str(float(N)) + " is not a natural number!")
elif N.is_integer():
    sumOdd(int(N))
else:
    print(str(N) + " is not a natural number!")
```

```
Enter any natural number: 13
169 is the sum of first 13 odd numbers.
```

In [2]:
```
from elementaryLibrary import sumOdd

#The following code is just to take care of any exceptional input.

N = float(input("Enter any natural number: "))
if N <= 0:
    print(str(float(N)) + " is not a natural number!")
elif N.is_integer():
    sumOdd(int(N))
else:
    print(str(N) + " is not a natural number!")
```

```
Enter any natural number: 125
15625 is the sum of first 125 odd numbers.
```

In [3]:
```
from elementaryLibrary import sumOdd

#The following code is just to take care of any exceptional input.

N = float(input("Enter any natural number: "))
if N <= 0:
    print(str(float(N)) + " is not a natural number!")
elif N.is_integer():
    sumOdd(int(N))
else:
    print(str(N) + " is not a natural number!")
```

```
Enter any natural number: -8
-8.0 is not a natural number!
```

In [4]:
```
from elementaryLibrary import sumOdd

#The following code is just to take care of any exceptional input.

N = float(input("Enter any natural number: "))
if N <= 0:
    print(str(float(N)) + " is not a natural number!")
elif N.is_integer():
    sumOdd(int(N))
else:
    print(str(N) + " is not a natural number!")
```

```
Enter any natural number: 0
0.0 is not a natural number!
```

In [5]:
```
from elementaryLibrary import sumOdd

#The following code is just to take care of any exceptional input.

N = float(input("Enter any natural number: "))
if N <= 0:
    print(str(float(N)) + " is not a natural number!")
elif N.is_integer():
    sumOdd(int(N))
else:
    print(str(N) + " is not a natural number!")
```

```
Enter any natural number: 7.8
7.8 is not a natural number!
```

Question 2

Arithmetic Progression

In [42]:
```
from elementaryLibrary import sumAP

a = float(input("Enter the first term of your arithmetic progression: "))
n = int(input("Enter the number of terms you need the summation of: "))

sumAP(a, 1.5, n)
```

```
Enter the first term of your arithmetic progression: 13
Enter the number of terms you need the summation of: 15
352.5 is the sum of first 15 terms of the arithmetic progression with first term 13.0 and common difference 1.
5.
```

In [6]:
```
from elementaryLibrary import sumAP

a = float(input("Enter the first term of your arithmetic progression: "))
n = int(input("Enter the number of terms you need the summation of: "))

sumAP(a, 1.5, n)
```

```
Enter the first term of your arithmetic progression: -8
Enter the number of terms you need the summation of: 16
52.0 is the sum of first 16 terms of the arithmetic progression with first term -8.0 and common difference 1.5.
```

In [7]:
```
from elementaryLibrary import sumAP

a = float(input("Enter the first term of your arithmetic progression: "))
n = int(input("Enter the number of terms you need the summation of: "))

sumAP(a, 1.5, n)
```

```
Enter the first term of your arithmetic progression: 0
Enter the number of terms you need the summation of: 54
2146.5 is the sum of first 54 terms of the arithmetic progression with first term 0.0 and common difference 1.
5.
```

In [8]:
```
from elementaryLibrary import sumAP

a = float(input("Enter the first term of your arithmetic progression: "))
n = int(input("Enter the number of terms you need the summation of: "))

sumAP(a, 1.5, n)
```

```
Enter the first term of your arithmetic progression: 90
Enter the number of terms you need the summation of: 5
465.0 is the sum of first 5 terms of the arithmetic progression with first term 90.0 and common difference 1.5.
```

In [9]:
```
from elementaryLibrary import sumAP

a = float(input("Enter the first term of your arithmetic progression: "))
n = int(input("Enter the number of terms you need the summation of: "))

sumAP(a, 1.5, n)
```

```
Enter the first term of your arithmetic progression: -4.45
Enter the number of terms you need the summation of: 15
90.74999999999999 is the sum of first 15 terms of the arithmetic progression with first term -4.45 and common d
ifference 1.5.
```

Geometric Progression

In [26]:
```
from elementaryLibrary import sumGP

a = float(input("Enter the first term of your geometric progression: "))
n = int(input("Enter the number of terms you need the summation of: "))

sumGP(a, 0.5, n)
```

```
Enter the first term of your geometric progression: 3
Enter the number of terms you need the summation of: 13
5.999267578125 is the sum of first 13 terms of the geometric progression with first term 3.0 and common ratio
0.5.
```

In [10]:
```
from elementaryLibrary import sumGP

a = float(input("Enter the first term of your geometric progression: "))
n = int(input("Enter the number of terms you need the summation of: "))

sumGP(a, 0.5, n)
```

```
Enter the first term of your geometric progression: 0
Enter the number of terms you need the summation of: 13
0.0 is the sum of first 13 terms of the geometric progression with first term 0.0 and common ratio 0.5.
```

In [11]:
```
from elementaryLibrary import sumGP

a = float(input("Enter the first term of your geometric progression: "))
n = int(input("Enter the number of terms you need the summation of: "))

sumGP(a, 0.5, n)
```

```
Enter the first term of your geometric progression: 7.5
Enter the number of terms you need the summation of: 13
14.9981689453125 is the sum of first 13 terms of the geometric progression with first term 7.5 and common ratio
0.5.
```

In [12]:
```
from elementaryLibrary import sumGP

a = float(input("Enter the first term of your geometric progression: "))
n = int(input("Enter the number of terms you need the summation of: "))

sumGP(a, 0.5, n)
```

```
Enter the first term of your geometric progression: 12
Enter the number of terms you need the summation of: 15
23.999267578125 is the sum of first 15 terms of the geometric progression with first term 12.0 and common ratio
0.5.
```

In [13]:
```
from elementaryLibrary import sumGP

a = float(input("Enter the first term of your geometric progression: "))
n = int(input("Enter the number of terms you need the summation of: "))

sumGP(a, 0.5, n)
```

```
Enter the first term of your geometric progression: 16
Enter the number of terms you need the summation of: 4
30.0 is the sum of first 4 terms of the geometric progression with first term 16.0 and common ratio 0.5.
```

Harmonic Progression

In [30]:
```
from elementaryLibrary import sumHP

a = float(input("Enter the first term of your harmonic progression: "))
n = int(input("Enter the number of terms you need the summation of: "))

sumHP(a, 1.5, n)
```

```
Enter the first term of your harmonic progression: 6
Enter the number of terms you need the summation of: 3
0.4111111111111111 is the sum of first 3 terms of the harmonic progression with first term 6.0 and common diffe
rence 1.5.
```

In [14]:
```
from elementaryLibrary import sumHP

a = float(input("Enter the first term of your harmonic progression: "))
n = int(input("Enter the number of terms you need the summation of: "))

sumHP(a, 1.5, n)
```

```
Enter the first term of your harmonic progression: 5
Enter the number of terms you need the summation of: 10
1.0038406865766296 is the sum of first 10 terms of the harmonic progression with first term 5.0 and common diff
erence 1.5.
```

In [15]:
```
from elementaryLibrary import sumHP

a = float(input("Enter the first term of your harmonic progression: "))
n = int(input("Enter the number of terms you need the summation of: "))

sumHP(a, 1.5, n)
```

```
Enter the first term of your harmonic progression: 0
Enter the number of terms you need the summation of: 12
A harmonic progression cannot have zero as one of its term!
```

In [16]:
```
from elementaryLibrary import sumHP

a = float(input("Enter the first term of your harmonic progression: "))
n = int(input("Enter the number of terms you need the summation of: "))

sumHP(a, 1.5, n)
```

```
Enter the first term of your harmonic progression: 15
Enter the number of terms you need the summation of: 10
0.47918093545028534 is the sum of first 10 terms of the harmonic progression with first term 15.0 and common di
fference 1.5.
```

In [17]:
```
from elementaryLibrary import sumHP

a = float(input("Enter the first term of your harmonic progression: "))
n = int(input("Enter the number of terms you need the summation of: "))

sumHP(a, 1.5, n)
```

```
Enter the first term of your harmonic progression: -4
Enter the number of terms you need the summation of: 15
2.0855009182368613 is the sum of first 15 terms of the harmonic progression with first term -4.0 and common dif
ference 1.5.
```

Question 3

In [48]:
```
from elementaryLibrary import fact

#The following code is just to take care of any exceptional input.

N = float(input("Enter any natural number: "))
if N < 0:
    print(str(float(N)) + " is not a natural number!")
elif N.is_integer():
    result = fact(int(N))
    print("The factorial of " + str(int(N)) + " is {}.".format(result))
else:
    print(str(N) + " is not a natural number!")
```

```
Enter any natural number: 13
The factorial of 13 is 6227020800.
```

In [18]:
```
from elementaryLibrary import fact

#The following code is just to take care of any exceptional input.

N = float(input("Enter any natural number: "))
if N < 0:
    print(str(float(N)) + " is not a natural number!")
elif N.is_integer():
    result = fact(int(N))
    print("The factorial of " + str(int(N)) + " is {}.".format(result))
else:
    print(str(N) + " is not a natural number!")
```

```
Enter any natural number: 0
The factorial of 0 is 1.
```

In [19]:
```
from elementaryLibrary import fact

#The following code is just to take care of any exceptional input.

N = float(input("Enter any natural number: "))
if N < 0:
    print(str(float(N)) + " is not a natural number!")
elif N.is_integer():
    result = fact(int(N))
    print("The factorial of " + str(int(N)) + " is {}.".format(result))
else:
    print(str(N) + " is not a natural number!")
```

```
Enter any natural number: 7
The factorial of 7 is 5040.
```

In [20]:
```
from elementaryLibrary import fact

#The following code is just to take care of any exceptional input.

N = float(input("Enter any natural number: "))
if N < 0:
    print(str(float(N)) + " is not a natural number!")
elif N.is_integer():
    result = fact(int(N))
    print("The factorial of " + str(int(N)) + " is {}.".format(result))
else:
    print(str(N) + " is not a natural number!")
```

```
Enter any natural number: 6.5
6.5 is not a natural number!
```

In [21]:
```
from elementaryLibrary import fact

#The following code is just to take care of any exceptional input.

N = float(input("Enter any natural number: "))
if N < 0:
    print(str(float(N)) + " is not a natural number!")
elif N.is_integer():
    result = fact(int(N))
    print("The factorial of " + str(int(N)) + " is {}.".format(result))
else:
    print(str(N) + " is not a natural number!")
```

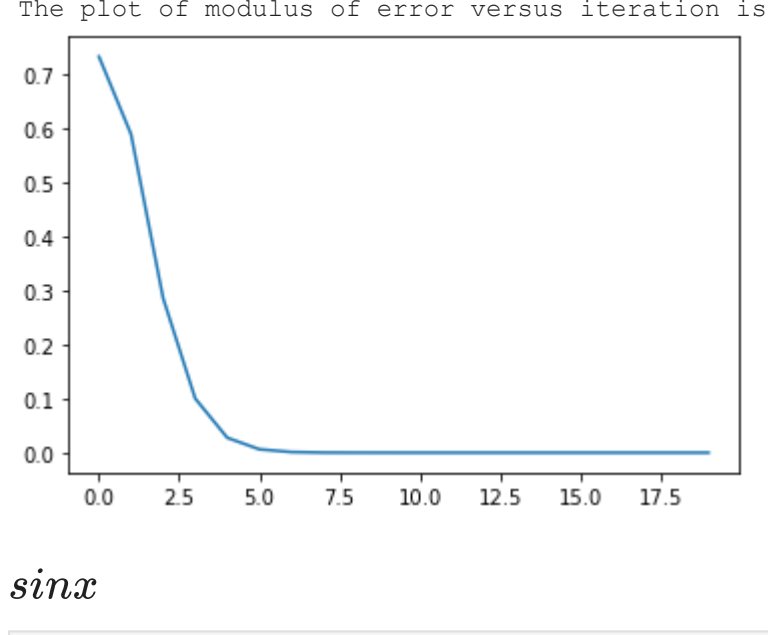
```
Enter any natural number: -4
-4.0 is not a natural number!
```

Question 4

e^{-x}

In [4]:
```
import math
from math import exp
from elementaryLibrary import fact
from matplotlib import pyplot as plt

x1 = float(input("Enter the number whose negative exponential you want to calculate: ")) #take input

#making a list which will contain the value on the X-Axis
x = []
for p in range(20):
    x.append(p)

def mynegexp(x):
    myexplist = []
    s = 0
    for i in range(20):
        s = s + ((-1)**i)*(x1)**i/fact(i)
        myexplist.append(s) #needed to construct the Y-axis data points
    j = 0
    while abs(myexplist[j] - exp(-(x1))) >= 10**(-5):
        j = j + 1 #this will make sure at least the first 4 decimal places are accurate
    print("The value accurate upto 4 decimal places is " + str(myexplist[j]) + ".")
    k = 0
    y = []
    for k in range(20):
        z = abs(myexplist[k] - exp(-(x1))) #constructing Y-axis data points
        y.append(z)
    print("The plot of modulus of error versus iteration is below:")
    plt.plot(x, y) #plotting the error vs iteration graph
    plt.show()

0mynegexp(x)
```

```
Enter the number whose negative exponential you want to calculate: 1.324
The value accurate upto 4 decimal places is 0.26606482750331445.
The plot of modulus of error versus iteration is below:
```



$sinx$

In [27]:
```
import math
from math import sin
from elementaryLibrary import fact
from matplotlib import pyplot as plt

x2 = float(input("Enter the number whose sine you want to calculate: ")) #take input

#making a list which will contain the value on the X-Axis
x = []
for p in range(20):
    x.append(p)

def mysin(x):
    mysinlist = []
    s = 0
    for i in range(20):
        s = s + ((-1)**i)*(x2)**(2*i + 1)/fact(2*i + 1)
        mysinlist.append(s) #needed to construct the Y-axis data points
    j = 0
    while abs(mysinlist[j] - sin(x2)) >= 10**(-10): #the sine function near boundary inputs like pi/2 can be ve
        j = j + 1 #this will make sure at least the first 4 decimal places are accurate
    print("The value accurate upto 4 decimal places is " + str(mysinlist[j]) + ".")
    k = 0
    y = []
    for k in range(20):
        z = abs(mysinlist[k] - sin(x2)) #constructing Y-axis data points
        y.append(z)
    print("The plot of modulus versus iteration is below: ")
    plt.plot(x, y) #plotting the error vs iteration graph
    plt.show()

mysin(x)
```

```
Enter the number whose sine you want to calculate: 3.442
The value accurate upto 4 decimal places is -0.29590933493770305.
The plot of modulus versus iteration is below:
```