Q1

class Solution:

def isPowerOfTwo(self, n: int) -> bool:

if n < 1:

return False

if n==1:

return True

return n%2 == 0 and self.isPowerOfTwo(n/2)

Q2

def findSum(n):

sum = 0

x = 1

while x <= n:

sum = sum + x

x = x + 1

return sum

n = 5

print findSum(n)

Q3

def recur\_factorial(n):

if n == 1:

return n

elif n < 0:

return None

else:

return n\*recur\_factorial(n-1)

num = int(input("Enter a number: "))

Q4

def recur\_factorial(n):

if n == 1:

return n

elif n < 0:

return None

else:

return n\*recur\_factorial(n-1)

num = int(input("Enter a number: "))

Q5

def power(base,exp):

if(exp==1):

return(base)

if(exp!=1):

return(base\*power(base,exp-1))

base=int(input("Enter base: "))

exp=int(input("Enter exponential value: "))

print("Result:",power(base,exp))

Q6

def Nth\_of\_AP(a, d, N) :

return (a + (N - 1) \* d)

a = 2

d = 1

N = 5

print( "The ", N ,"th term of the series is : ",

Nth\_of\_AP(a, d, N))

Q7

def toString(List):

    return ''.join(List)

 def permute(a, l, r):

    if l == r:

        print(toString(a))

    else:

        for i in range(l, r):

            a[l], a[i] = a[i], a[l]

            permute(a, l+1, r)

            a[l], a[i] = a[i], a[l]

 string = "ABC"

n = len(string)

a = list(string)

 permute(a, 0, n)

Q8

arr=[1,2,3,4,5]

product=1

i=0

j=len(arr)-1

while(i<j):

    product\*=arr[i]\*arr[j]

    i+=1

    j-=1

if(i==j):

    product\*=arr[i]

print(product)