ASSIGNMENT-IX

**Question 1**

Given an integer n, return *true* if it is a power of two. Otherwise, return *false*.

An integer n is a power of two, if there exists an integer x such that n == 2x.

**Example 1:** Input: n = 1

Output: true

**Example 2:** Input: n = 16

Output: true

**Example 3:** Input: n = 3

Output: false

**Ans:**

def isPowerOfTwo(n):

if n <= 0:

return False

return (n & (n - 1)) == 0

**Question 2**

Given a number n, find the sum of the first natural numbers.

**Example 1:**

Input: n = 3

Output: 6

**Example 2:**

Input : 5

Output : 15

**Ans:**

def sumOfNaturalNumbers(n):

return (n \* (n + 1)) // 2

**Question 3**

\*\*\*\*Given a positive integer, N. Find the factorial of N.

**Example 1:**

Input: N = 5

Output: 120

**Example 2:**

Input: N = 4

Output: 24

**Ans:**

def factorial(N):

if N == 0:

return 1

factorial = 1

for i in range(1, N + 1):

factorial \*= i

return factorial

**Question 4**

Given a number N and a power P, the task is to find the exponent of this number raised to the given power, i.e. N^P.

**Example 1 :**

Input: N = 5, P = 2

Output: 25

**Example 2 :** Input: N = 2, P = 5

Output: 32

**Ans:**

def calculateExponent(N, P):

return N \*\* P

**Question 5**

Given an array of integers **arr**, the task is to find maximum element of that array using recursion.

**Example 1:**

Input: arr = {1, 4, 3, -5, -4, 8, 6}; Output: 8

**Example 2:**

Input: arr = {1, 4, 45, 6, 10, -8}; Output: 45

**Ans:**

def findMaximum(arr):

if len(arr) == 1:

return arr[0]

else:

mid = len(arr) // 2

left\_max = findMaximum(arr[:mid])

right\_max = findMaximum(arr[mid:])

return max(left\_max, right\_max)

**Question 6**

Given first term (a), common difference (d) and a integer N of the Arithmetic Progression series, the task is to find Nth term of the series.

**Example 1:**

Input : a = 2 d = 1 N = 5 Output : 6 The 5th term of the series is : 6

**Example 2:**

Input : a = 5 d = 2 N = 10 Output : 23 The 10th term of the series is : 23

**Ans:**

def findNthTerm(a, d, N):

return a + (N - 1) \* d

**Question 7**

Given a string S, the task is to write a program to print all permutations of a given string.

**Example 1:**

***Input:***

S = “ABC”

***Output:***

“ABC”, “ACB”, “BAC”, “BCA”, “CBA”, “CAB”

**Example 2:**

***Input:***

S = “XY”

***Output:***

“XY”, “YX”

**Ans:**

def generatePermutations(S, left, right):

if left == right:

print(''.join(S))

else:

for i in range(left, right + 1):

S[left], S[i] = S[i], S[left]

generatePermutations(S, left + 1, right)

S[left], S[i] = S[i], S[left] # backtrack

def printAllPermutations(S):

n = len(S)

generatePermutations(list(S), 0, n - 1)

**Question 8**

Given an array, find a product of all array elements.

**Example 1:**

Input : arr[] = {1, 2, 3, 4, 5} Output : 120

**Example 2:**

Input : arr[] = {1, 6, 3} Output : 18

**Ans:**

def productOfArray(arr):

product = 1

for num in arr:

product \*= num

return product