Numpy Assignment :-

1. Write a function so that the columns of the output matrix are powers of the input vector. The order of the powers is determined by the increasing Boolean argument. Specifically, when increasing is False, the i-th output column is the input vector raised element-wise to the power of N - i - 1.

HINT: Such a matrix with a geometric progression in each row is named for Alexandre-

Theophile Vandermonde.

Solution 1:

```
import numpy as np
x=np.array([1,2,3,4,5])
N=5
matrix = np.column_stack([x**(N-i-1) for i in range(N)])
print(matrix)
```

Problem Statement 2:

Given a sequence of n values x1, x2, ..., xn and a window size k>0, the k-th moving average of

the given sequence is defined as follows:

The moving average sequence has n-k+1 elements as shown below.

The moving averages with k=4 of a ten-value sequence (n=10) is shown below

i 1 2 3 4 5 6 7 8 9 10

===== == == == == == == == == == ==

Input 10 20 30 40 50 60 70 80 90 100

y1 25 = (10+20+30+40)/4

y2 35 = (20+30+40+50)/4

y3 45 = (30+40+50+60)/4

y4 55 = (40+50+60+70)/4

y5 65 = (50+60+70+80)/4

y6 75 = (60+70+80+90)/4

y7 85 = (70+80+90+100)/4

Thus, the moving average sequence has n-k+1=10-4+1=7 values.

Question: Write a function to find moving average in an array over a window:

Test it over [3, 5, 7, 2, 8, 10, 11, 65, 72, 81, 99, 100, 150] and window of 3.

Solution:
```
def mov_avg_val(mylist, N):

    cumsum, moving_aves = [0], []
    for i, x in enumerate(mylist, 1):
        cumsum.append(cumsum[i-1] + x) # summing up all the values one by one
and appending the list
        if i>=N:
            moving_ave = round(((cumsum[i] - cumsum[i-N])/N),2) #computing the
moving average using temp array sum array cumsum
            #can do stuff with moving_ave here
            moving_aves.append(moving_ave) # listing the moving average as an
single array
    print("Moving average values list: ", moving_aves)
    print("Length of the list l-N+1: ", len(moving_aves))


mylist=[3, 5, 7, 2, 8, 10, 11, 65, 72, 81, 99, 100, 150]
N=3
mov_avg_val(mylist, N)
```