**Problem Statement**

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.

**Problem Statement**

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.

**Code:**

import numpy as np

def moving\_average(a,k):#a is the input array and k is the window#

temp=np.cumsum(a,dtype=float)

temp[3:]=temp[3:]-temp[:-3]

return temp[3-1:]/3

lst=[3,5,7,2,8,10,11,65,72,81,99,100,150]#input#

a = np.array(lst)

op\_array = moving\_average(a,k=3)#output#

print("Moving average of",lst,"are:\n")

print("Output:\n",op\_array)

**Output**

Moving average of [3, 5, 7, 2, 8, 10, 11, 65, 72, 81, 99, 100, 150] are:

Output:

[ 5. 4.66666667 5.66666667 6.66666667 9.66666667

28.66666667 49.33333333 72.66666667 84. 93.33333333

116.33333333]