In [1]:

```
"""
Given a sequence of n values x1, x2, ..., xn and a window size k>0, the k-th m
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
"""
import numpy as np
def Moving_Average(x,K):
    res=[]
    N=x.shape[0] # getting the number of elements of an array into the variabl
    elmnt=0
    count=N-K+1

    for i in range(K,N+1):
        sumelmnt=0
        for j in range (elmnt,i):
            sumelmnt=sumelmnt+x[j]
        lst=sumelmnt/3
        res.append(lst)
        elmnt=elmnt+1
    print("Given a sequence of"+str(N)+" valuess :"+str(x)+"\nThe moving avera
    for i in range(1,count+1):
        print(str(i)+":\t"+str(res[i-1])+"\n")
    print("the moving average sequence has n-k+1=" +str(N)+"-"+str(K)+"+1="+st

inpt = np.array([3, 5, 7, 2, 8, 10, 11, 65, 72, 81, 99, 100, 150])
K= int(input("Enter the Moving average"))
Moving_Average(inpt,K)
```

```
Enter the Moving average3
Given a sequence of13 valuess :[  3   5   7   2   8  10  11  65  72  81  99 10
0 150]
The moving averages with k=3
1:      5.0

2:      4.666666666666667

3:      5.666666666666667

4:      6.666666666666667

5:      9.666666666666666
```

```
6:      28.666666666666668

7:      49.333333333333336

8:      72.66666666666667

9:      84.0

10:     93.33333333333333

11:     116.33333333333333

the moving average sequence has n-k+1=13-3+1=11values
```

In [ ]:
```
1
```