ASSIGNMENT 7.1

Student: K. Anandaranga

---

1. Introduction

This assignment will help you to consolidate the concepts learnt in the session.

2. Problem Statement (a)

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 (b)

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.

3. Output

Solution

Note: I wrote a function and used that to show the output for both questions (a) and (b) by feeding it different sequences and window sizes.

ACD MDS Assignment 7.1 (a)

Student: K. Anandaranga - Mar 2018 batch

In [116]:
```
def calc_mov_avg():
    ''' Given a sequence l of n values, and window size k, calculate the moving average '''

    l = [int(x) for x in input("Enter the integer sequence (example: 2 4 5 10 12) :").split()]  # User provided sequence
    window_size = int (input("Enter preferred moving avg window size : "))     # User provided value for window size

    n = len(l)                                  # Count of number of values in sequence
    max_values = n - window_size + 1            # For any given combination of n, and window_size, this is the max values
    m_avg = []                                  # Empty list

    for i in range (max_values):                # Loop based on the count of final calculated values
        mov_tot = 0
        for j in range (window_size):           # Each calculation should only include per window size
            select_index = i + j                # The starting point for each window is determined here
            mov_tot = mov_tot + l[select_index] # Rolling totals are captured
        m_avg.append (mov_tot/window_size)    # Avg of rolling total is then appended at the end of each iteration

    print ("_"*60)
    print ("Total number of values = {} and Window size = {}".format(n, window_size))
    print ("_"*60)
    for i in range (max_values):                # Use the loop to print out the values in each window and the ...
        values = []                             # ... corresponding moving average value
        for j in range (window_size):
            values.append (l[i+j])
        print ("Moving Avg Value#", i+1, " = ", m_avg[i], "for values", values )
    return m_avg
```

In [120]:
```
calc_mov_avg()
```

```
Enter the integer sequence (example: 2 4 5 10 12) :10 20 30 40 50 60 70 80 90 100
Enter preferred moving avg window size : 4
------------------------------------------------------------
Total number of values = 10 and Window size = 4
------------------------------------------------------------
Moving Avg Value# 1  =  25.0 for values [10, 20, 30, 40]
Moving Avg Value# 2  =  35.0 for values [20, 30, 40, 50]
Moving Avg Value# 3  =  45.0 for values [30, 40, 50, 60]
Moving Avg Value# 4  =  55.0 for values [40, 50, 60, 70]
Moving Avg Value# 5  =  65.0 for values [50, 60, 70, 80]
Moving Avg Value# 6  =  75.0 for values [60, 70, 80, 90]
Moving Avg Value# 7  =  85.0 for values [70, 80, 90, 100]
```

Out[120]:
```
[25.0, 35.0, 45.0, 55.0, 65.0, 75.0, 85.0]
```

In [ ]:

In the above, I used the sequence (10 20 30 …. 100) with moving average window size of 4.

ACD MDS Assignment 7.1 (b)

Student: K. Anandaranga - Mar 2018 batch

In [116]:
```
def calc_mov_avg():
    ''' Given a sequence l of n values, and window size k, calculate the moving average '''

    l = [int(x) for x in input("Enter the integer sequence (example: 2 4 5 10 12) :").split()]  # User provided sequence
    window_size = int (input("Enter preferred moving avg window size : "))     # User provided value for window size

    n = len(l)                                  # Count of number of values in sequence
    max_values = n - window_size + 1            # For any given combination of n, and window_size, this is the max values
    m_avg = []                                  # Empty list

    for i in range (max_values):                # Loop based on the count of final calculated values
        mov_tot = 0
        for j in range (window_size):           # Each calculation should only include per window size
            select_index = i + j                # The starting point for each window is determined here
            mov_tot = mov_tot + l[select_index] # Rolling totals are captured
        m_avg.append (mov_tot/window_size)    # Avg of rolling total is then appended at the end of each iteration

    print ("_"*60)
    print ("Total number of values = {} and Window size = {}".format(n, window_size))
    print ("_"*60)
    for i in range (max_values):                # Use the loop to print out the values in each window and the ...
        values = []                             # ... corresponding moving average value
        for j in range (window_size):
            values.append (l[i+j])
        print ("Moving Avg Value#", i+1, " = ", m_avg[i], "for values", values )
    return m_avg
```

In [121]:
```
calc_mov_avg()
```

```
Enter the integer sequence (example: 2 4 5 10 12) :3 5 7 2 8 10 11 65 72 81 99 100 150
Enter preferred moving avg window size : 3
------------------------------------------------------------
Total number of values = 13 and Window size = 3
------------------------------------------------------------
Moving Avg Value# 1  =  5.0 for values [3, 5, 7]
Moving Avg Value# 2  =  4.666666666666667 for values [5, 7, 2]
Moving Avg Value# 3  =  5.666666666666667 for values [7, 2, 8]
Moving Avg Value# 4  =  6.666666666666667 for values [2, 8, 10]
Moving Avg Value# 5  =  9.666666666666666 for values [8, 10, 11]
Moving Avg Value# 6  =  28.666666666666668 for values [10, 11, 65]
Moving Avg Value# 7  =  49.333333333333336 for values [11, 65, 72]
Moving Avg Value# 8  =  72.66666666666667 for values [65, 72, 81]
Moving Avg Value# 9  =  84.0 for values [72, 81, 99]
Moving Avg Value# 10  =  93.33333333333333 for values [81, 99, 100]
Moving Avg Value# 11  =  116.33333333333333 for values [99, 100, 150]
```

Out[121]:
```
[5.0,
 4.666666666666667,
 5.666666666666667,
 6.666666666666667,
 9.666666666666666,
 28.666666666666668,
 49.333333333333336,
 72.66666666666667,
 84.0,
 93.33333333333333,
 116.33333333333333]
```

In the above screenshot, I've provided the input sequence and window size as per the assignment.