210411100085_Modul6_InsertionSort

May 17, 2022

1 Ascending

=> Data [10, 2, 4, 5, 20, 8, 15] Akan diurutkan secara Ascending

I key = 8, Data pengurutan = 8, 15

8 15 (1)

II key = 20, Data pengurutan = 20, 8, 15

20 8 15 (1) | 8 15 20

8 8 15 (2)

8 15 15

III key = 5, Data Pengurutan = 5, 8, 15, 20

5 8 15 20 (1)

IV key = 4, Data pengurutan = 4, 5, 8, 15, 20

4 5 8 15 20 (1)

V key = 2, Data pengurutan = 2, 4, 5, 8, 15, 20

2 4 5 8 15 20 (1)

VI key = 10, Data Pengurutan = 10, 2, 4, 5, 8, 15, 20

10 2 4 5 8 15 20 (1)

2 2 4 5 8 15 20 (2)

2 4 4 5 8 15 20 (3)

2 4 5 5 8 15 20 (4)

2 4 5 8 8 15 20 (5)

2 4 5 8 10 15 20

=> Data Terurut = [2, 4, 5, 8, 10, 15, 20]

Data yang dibandingkan adalah key dan yang dilingkari

[12]:
```
def insertionSortAscending(nums) :
    maxIdx = len(nums)-1

    for i in range(maxIdx,0,-1):

        print("Data : ",nums)
        key = nums[i-1]
        print("key, data[%d] : %d" % (i-1,key) )
        iAfter = i

        while iAfter <= maxIdx and key>=nums[iAfter]  :
            nums[iAfter-1] = nums[iAfter]
            print("Inner Sorting = ",nums)
            iAfter+=1
        nums[iAfter-1] = key


    return nums


nums   = [10,2,4,5,20,8,15]
print("Sorted Data : ",insertionSortAscending(nums))
```

```
Data :  [10, 2, 4, 5, 20, 8, 15]
key, data[5] : 8
Data :  [10, 2, 4, 5, 20, 8, 15]
key, data[4] : 20
Inner Sorting =  [10, 2, 4, 5, 8, 8, 15]
Inner Sorting =  [10, 2, 4, 5, 8, 15, 15]
Data :  [10, 2, 4, 5, 8, 15, 20]
key, data[3] : 5
Data :  [10, 2, 4, 5, 8, 15, 20]
key, data[2] : 4
Data :  [10, 2, 4, 5, 8, 15, 20]
key, data[1] : 2
Data :  [10, 2, 4, 5, 8, 15, 20]
key, data[0] : 10
Inner Sorting =  [2, 2, 4, 5, 8, 15, 20]
Inner Sorting =  [2, 4, 4, 5, 8, 15, 20]
Inner Sorting =  [2, 4, 5, 5, 8, 15, 20]
Inner Sorting =  [2, 4, 5, 8, 8, 15, 20]
Sorted Data :  [2, 4, 5, 8, 10, 15, 20]
```

1.1 Descending

=> Data = [10, 9, 1, 3, 8, 7, 11] akan diurutkan secara descending

I key = 7, Data pengurutan = 7, 11

7 (11)[1]
11 11

7 11

Data yang dibandingkan adalah key dan data yang dilingkari

II key = 8, Data pengurutan = 8, 11, 7

8 (11)[1] 7
11 11 (7)[2]
11 8 7

III key = 3, Data Pengurutan = 3, 11, 8, 7

3 (11)[1] 8 7
11 11 (8)[2] 7
11 8 8 (7)[3]
11 8 7 7
11 8 7 3

IV key = 1, Data Pengurutan = 1, 11, 8, 7, 3

1 (11)[1] 8 7 3
11 11 (8)[2] 7 3
11 8 8 (7)[3] 3
11 8 7 7 (3)[4]

11 8 7 3 3
11 8 7 3 1

V key = 9, Data pengurutan = 9, 11, 8, 7, 3, 1

9 (11)[1] 8 7 3 1
11 11 (8)[2] 7 3 1
11 9 8 7 3 1

VI key = 10, Data Pengurutan = 10, 11, 9, 8, 7, 3, 1

10 (11)[1] 9 8 7 3 1
11 11 9 8 7 3 1
11 10 9 8 7 3 1

[13]:
```
def insertionSortDescending(nums) :
    maxIdx = len(nums)-1

    for i in range(maxIdx,0,-1):

        print("Data : ",nums)
        key = nums[i-1]
        print("key, data[%d] : %d" % (i-1,key) )
        iAfter = i

        while iAfter <= maxIdx and key<=nums[iAfter]  :
            nums[iAfter-1] = nums[iAfter]
            print("Inner Sorting = ",nums)
            iAfter+=1
        nums[iAfter-1] = key


    return nums


nums   = [10,9,1,3,8,7,11]
print("Sorted Data : ",insertionSortDescending(nums))
```

```
Data :  [10, 9, 1, 3, 8, 7, 11]
key, data[5] : 7
Inner Sorting =  [10, 9, 1, 3, 8, 11, 11]
Data :  [10, 9, 1, 3, 8, 11, 7]
key, data[4] : 8
Inner Sorting =  [10, 9, 1, 3, 11, 11, 7]
Data :  [10, 9, 1, 3, 11, 8, 7]
key, data[3] : 3
Inner Sorting =  [10, 9, 1, 11, 11, 8, 7]
Inner Sorting =  [10, 9, 1, 11, 8, 8, 7]
Inner Sorting =  [10, 9, 1, 11, 8, 7, 7]
Data :  [10, 9, 1, 11, 8, 7, 3]
key, data[2] : 1
Inner Sorting =  [10, 9, 11, 11, 8, 7, 3]
Inner Sorting =  [10, 9, 11, 8, 8, 7, 3]
Inner Sorting =  [10, 9, 11, 8, 7, 7, 3]
Inner Sorting =  [10, 9, 11, 8, 7, 3, 3]
Data :  [10, 9, 11, 8, 7, 3, 1]
key, data[1] : 9
Inner Sorting =  [10, 11, 11, 8, 7, 3, 1]
Data :  [10, 11, 9, 8, 7, 3, 1]
key, data[0] : 10
Inner Sorting =  [11, 11, 9, 8, 7, 3, 1]
Sorted Data :  [11, 10, 9, 8, 7, 3, 1]
```