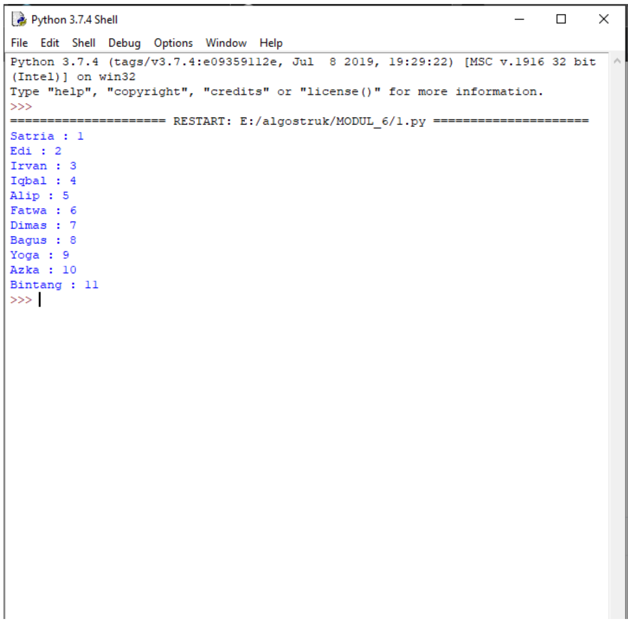
Praktikum Algostruk
Modul 6

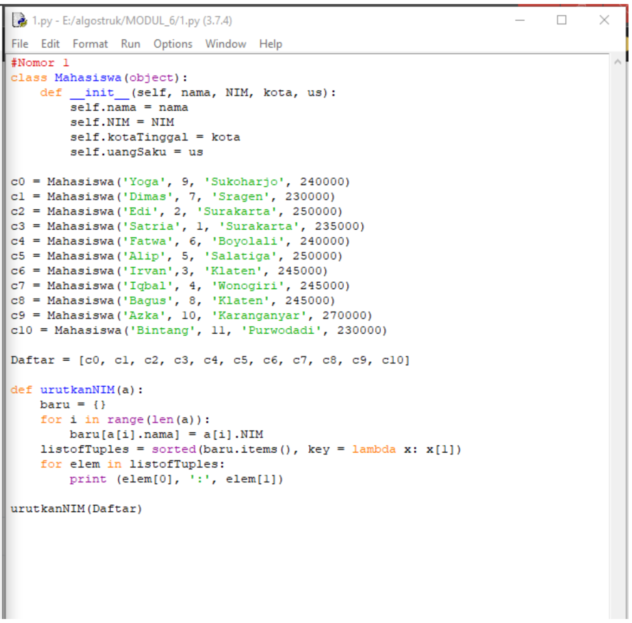
Nama : Edi Supriyanto
NIM : L200180002
Kelas : A

Tugas

1.

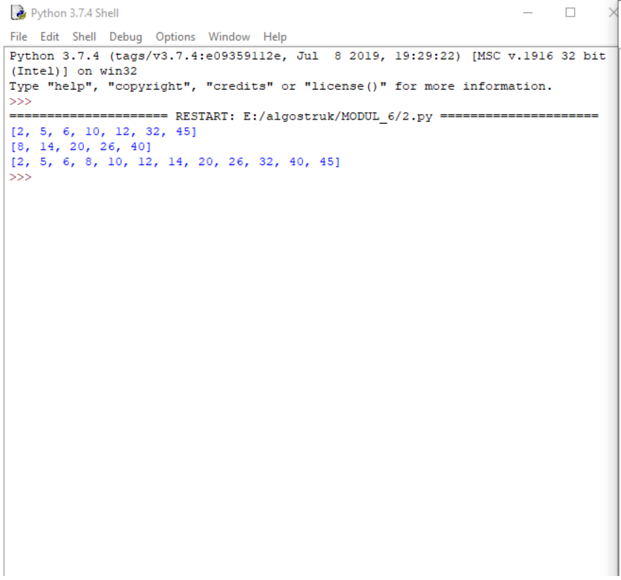


```
Python 3.7.4 (tags/v3.7.4:e09359112e, Jul  8 2019, 19:29:22) [MSC v.1916 32 bit
(Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> 
===================== RESTART: E:/algostruk/MODUL_6/1.py =====================
Satria : 1
Edi : 2
Irvan : 3
Iqbal : 4
Alip : 5
Fatwa : 6
Dimas : 7
Bagus : 8
Yoga : 9
Azka : 10
Bintang : 11
>>> 
```

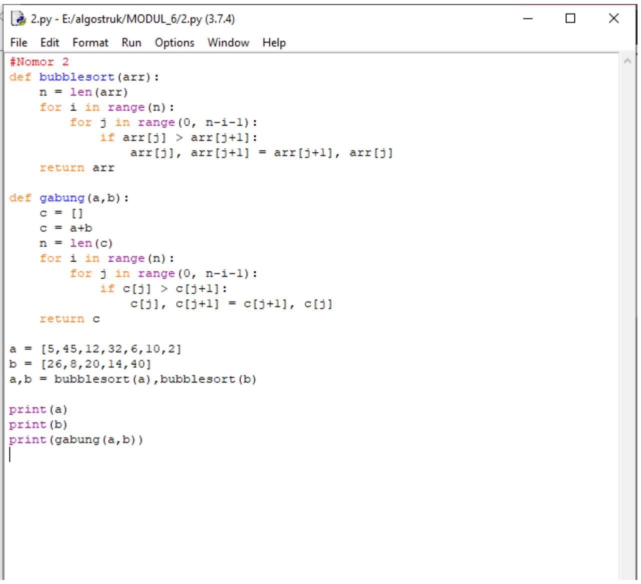

1.py - E:/algostruk/MODUL_6/1.py (3.7.4)
File Edit Format Run Options Window Help

```python
#Nomor 1
class Mahasiswa(object):
    def __init__(self, nama, NIM, kota, us):
        self.nama = nama
        self.NIM = NIM
        self.kotaTinggal = kota
        self.uangSaku = us

c0 = Mahasiswa('Yoga', 9, 'Sukoharjo', 240000)
c1 = Mahasiswa('Dimas', 7, 'Sragen', 230000)
c2 = Mahasiswa('Edi', 2, 'Surakarta', 250000)
c3 = Mahasiswa('Satria', 1, 'Surakarta', 235000)
c4 = Mahasiswa('Fatwa', 6, 'Boyolali', 240000)
c5 = Mahasiswa('Alip', 5, 'Salatiga', 250000)
c6 = Mahasiswa('Irvan',3, 'Klaten', 245000)
c7 = Mahasiswa('Iqbal', 4, 'Wonogiri', 245000)
c8 = Mahasiswa('Bagus', 8, 'Klaten', 245000)
c9 = Mahasiswa('Azka', 10, 'Karanganyar', 270000)
c10 = Mahasiswa('Bintang', 11, 'Purwodadi', 230000)

Daftar = [c0, c1, c2, c3, c4, c5, c6, c7, c8, c9, c10]

def urutkanNIM(a):
    baru = {}
    for i in range(len(a)):
        baru[a[i].nama] = a[i].NIM
    listofTuples = sorted(baru.items(), key = lambda x: x[1])
    for elem in listofTuples:
        print (elem[0], ':', elem[1])

urutkanNIM(Daftar)
```

2.

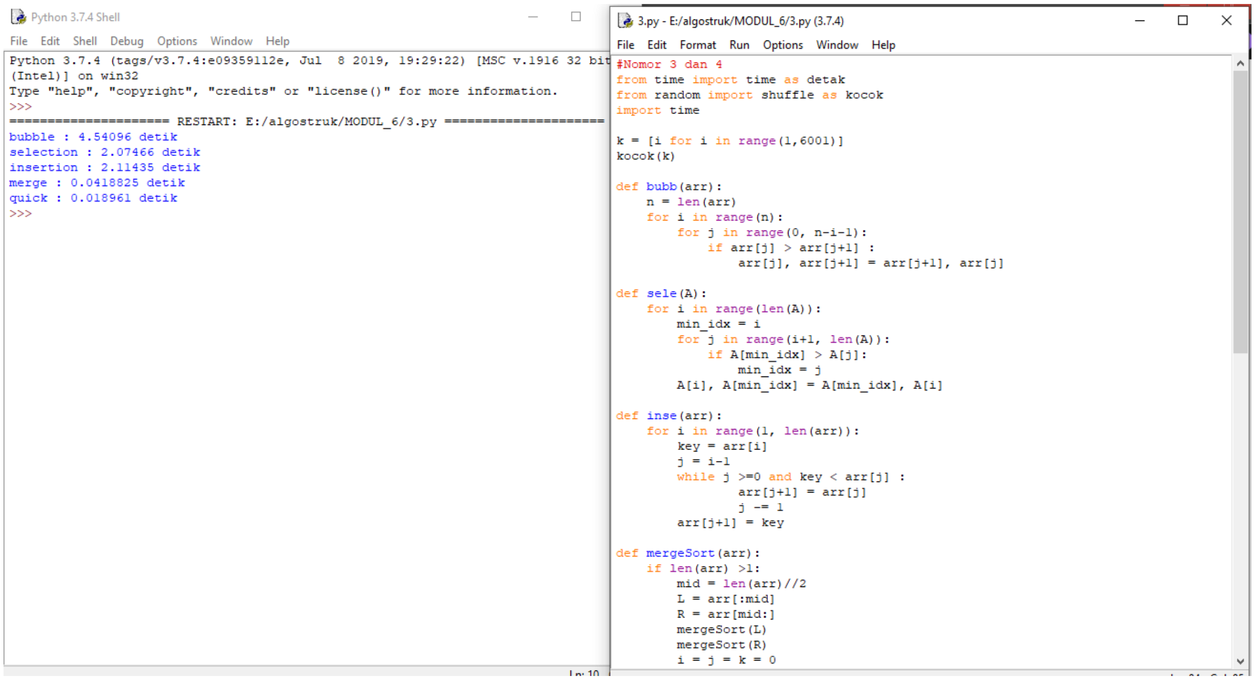


```
Python 3.7.4 (tags/v3.7.4:e09359112e, Jul  8 2019, 19:29:22) [MSC v.1916 32 bit
(Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> 
===================== RESTART: E:/algostruk/MODUL_6/2.py =====================
[2, 5, 6, 10, 12, 32, 45]
[8, 14, 20, 26, 40]
[2, 5, 6, 8, 10, 12, 14, 20, 26, 32, 40, 45]
>>> 
```


2.py - E:/algostruk/MODUL_6/2.py (3.7.4)
File Edit Format Run Options Window Help

```python
#Nomor 2
def bubblesort(arr):
    n = len(arr)
    for i in range(n):
        for j in range(0, n-i-1):
            if arr[j] > arr[j+1]:
                arr[j], arr[j+1] = arr[j+1], arr[j]
    return arr

def gabung(a,b):
    c = []
    c = a+b
    n = len(c)
    for i in range(n):
        for j in range(0, n-i-1):
            if c[j] > c[j+1]:
                c[j], c[j+1] = c[j+1], c[j]
    return c

a = [5,45,12,32,6,10,2]
b = [26,8,20,14,40]
a,b = bubblesort(a),bubblesort(b)

print(a)
print(b)
print(gabung(a,b))
```

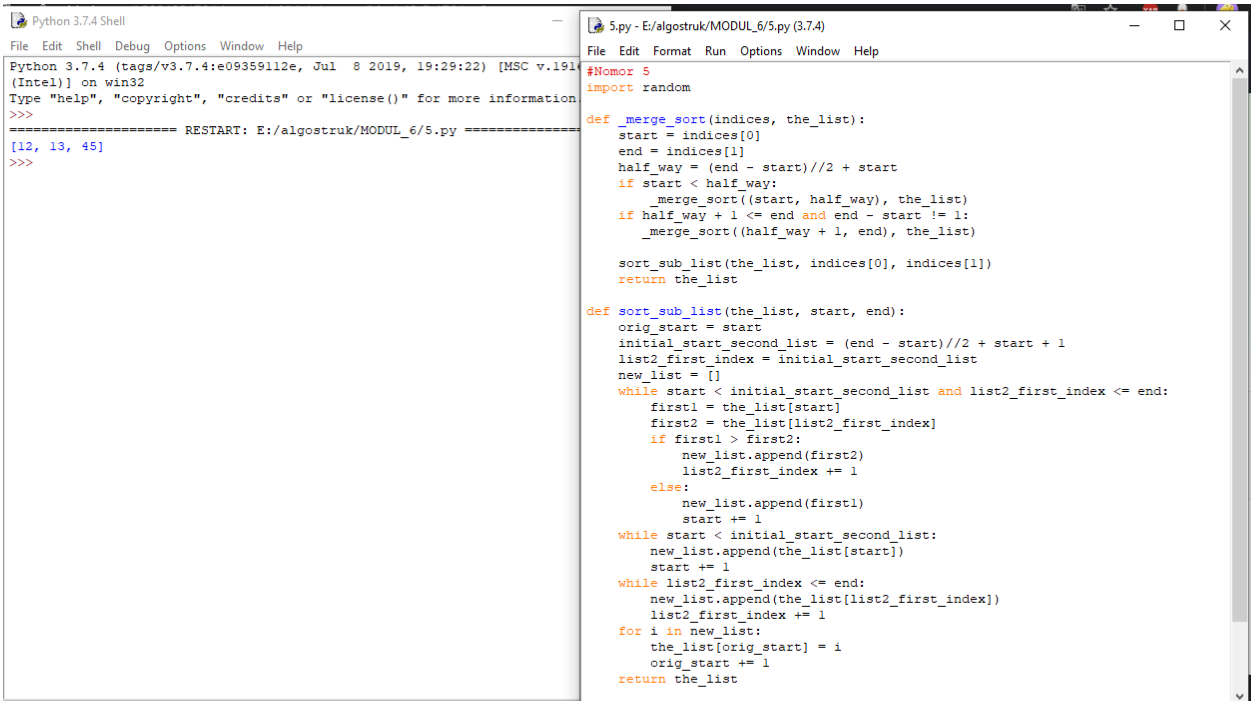
3. 4.


Python 3.7.4 Shell
File Edit Shell Debug Options Window Help
3.py - E:/algostruk/MODUL_6/3.py (3.7.4)
File Edit Format Run Options Window Help

```
Python 3.7.4 (tags/v3.7.4:e09359112e, Jul  8 2019, 19:29:22) [MSC v.1916 32 bit
(Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> 
==================== RESTART: E:/algostruk/MODUL_6/3.py ====================
bubble : 4.54096 detik
selection : 2.07466 detik
insertion : 2.11435 detik
merge : 0.0418825 detik
quick : 0.018961 detik
>>> 
```

```python
#Nomor 3 dan 4
from time import time as detak
from random import shuffle as kocok
import time

k = [i for i in range(1,6001)]
kocok(k)

def bubb(arr):
    n = len(arr)
    for i in range(n):
        for j in range(0, n-i-1):
            if arr[j] > arr[j+1] :
                arr[j], arr[j+1] = arr[j+1], arr[j]

def sele(A):
    for i in range(len(A)):
        min_idx = i
        for j in range(i+1, len(A)):
            if A[min_idx] > A[j]:
                min_idx = j
        A[i], A[min_idx] = A[min_idx], A[i]

def inse(arr):
    for i in range(1, len(arr)):
        key = arr[i]
        j = i-1
        while j >=0 and key < arr[j] :
                arr[j+1] = arr[j]
                j -= 1
        arr[j+1] = key

def mergeSort(arr):
    if len(arr) >1:
        mid = len(arr)//2
        L = arr[:mid]
        R = arr[mid:]
        mergeSort(L)
        mergeSort(R)
        i = j = k = 0
```

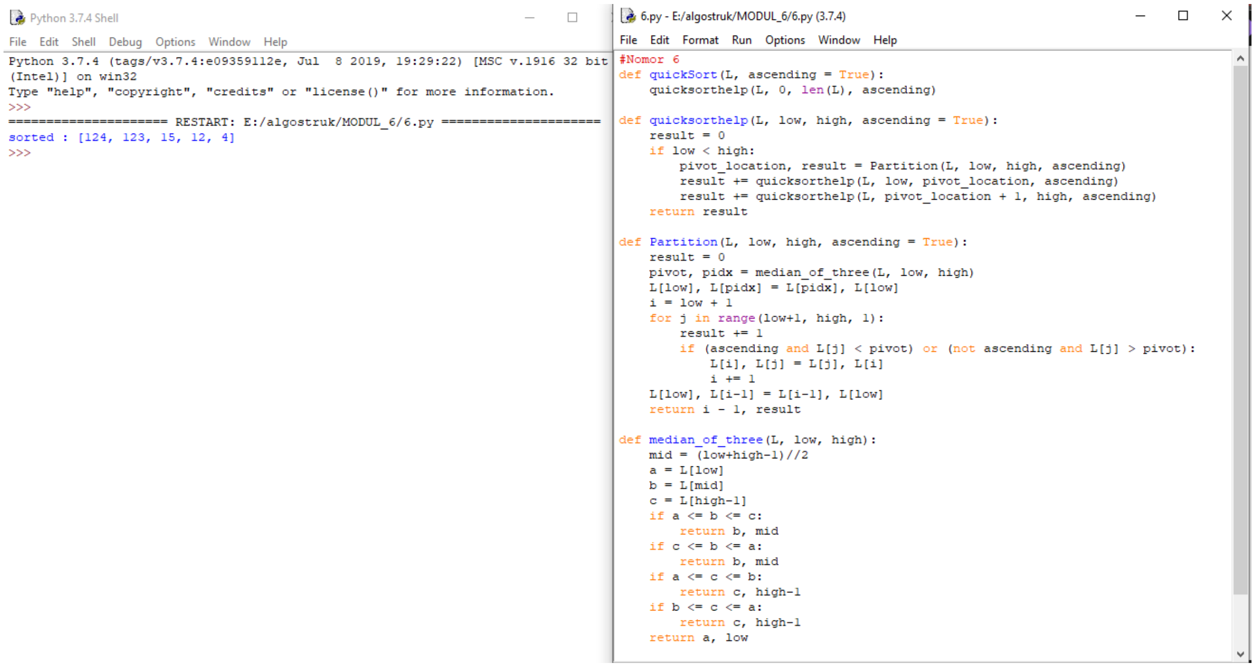
5.


Python 3.7.4 Shell
File Edit Shell Debug Options Window Help
5.py - E:/algostruk/MODUL_6/5.py (3.7.4)
File Edit Format Run Options Window Help

```
Python 3.7.4 (tags/v3.7.4:e09359112e, Jul  8 2019, 19:29:22) [MSC v.191
(Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information
>>> 
===================== RESTART: E:/algostruk/MODUL_6/5.py ==============
[12, 13, 45]
>>> 
```

```python
#Nomor 5
import random

def _merge_sort(indices, the_list):
    start = indices[0]
    end = indices[1]
    half_way = (end - start)//2 + start
    if start < half_way:
        _merge_sort((start, half_way), the_list)
    if half_way + 1 <= end and end - start != 1:
        _merge_sort((half_way + 1, end), the_list)

    sort_sub_list(the_list, indices[0], indices[1])
    return the_list

def sort_sub_list(the_list, start, end):
    orig_start = start
    initial_start_second_list = (end - start)//2 + start + 1
    list2_first_index = initial_start_second_list
    new_list = []
    while start < initial_start_second_list and list2_first_index <= end:
        first1 = the_list[start]
        first2 = the_list[list2_first_index]
        if first1 > first2:
            new_list.append(first2)
            list2_first_index += 1
        else:
            new_list.append(first1)
            start += 1
    while start < initial_start_second_list:
        new_list.append(the_list[start])
        start += 1
    while list2_first_index <= end:
        new_list.append(the_list[list2_first_index])
        list2_first_index += 1
    for i in new_list:
        the_list[orig_start] = i
        orig_start += 1
    return the_list
```

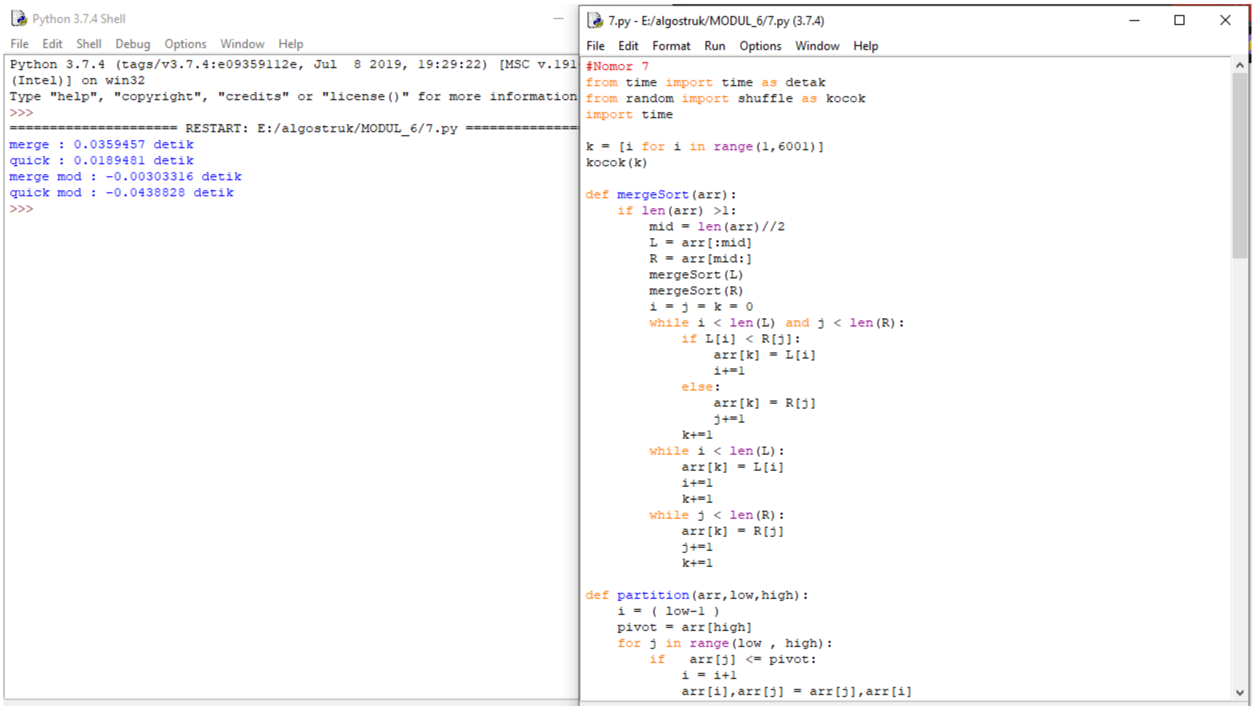
6.


Python 3.7.4 Shell
File Edit Shell Debug Options Window Help
6.py - E:/algostruk/MODUL_6/6.py (3.7.4)
File Edit Format Run Options Window Help

```
Python 3.7.4 (tags/v3.7.4:e09359112e, Jul  8 2019, 19:29:22) [MSC v.1916 32 bit
(Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> 
==================== RESTART: E:/algostruk/MODUL_6/6.py ====================
sorted : [124, 123, 15, 12, 4]
>>> 
```

```
#Nomor 6
def quickSort(L, ascending = True):
    quicksorthelp(L, 0, len(L), ascending)

def quicksorthelp(L, low, high, ascending = True):
    result = 0
    if low < high:
        pivot_location, result = Partition(L, low, high, ascending)
        result += quicksorthelp(L, low, pivot_location, ascending)
        result += quicksorthelp(L, pivot_location + 1, high, ascending)
    return result

def Partition(L, low, high, ascending = True):
    result = 0
    pivot, pidx = median_of_three(L, low, high)
    L[low], L[pidx] = L[pidx], L[low]
    i = low + 1
    for j in range(low+1, high, 1):
        result += 1
        if (ascending and L[j] < pivot) or (not ascending and L[j] > pivot):
            L[i], L[j] = L[j], L[i]
            i += 1
    L[low], L[i-1] = L[i-1], L[low]
    return i - 1, result

def median_of_three(L, low, high):
    mid = (low+high-1)//2
    a = L[low]
    b = L[mid]
    c = L[high-1]
    if a <= b <= c:
        return b, mid
    if c <= b <= a:
        return b, mid
    if a <= c <= b:
        return c, high-1
    if b <= c <= a:
        return c, high-1
    return a, low
```

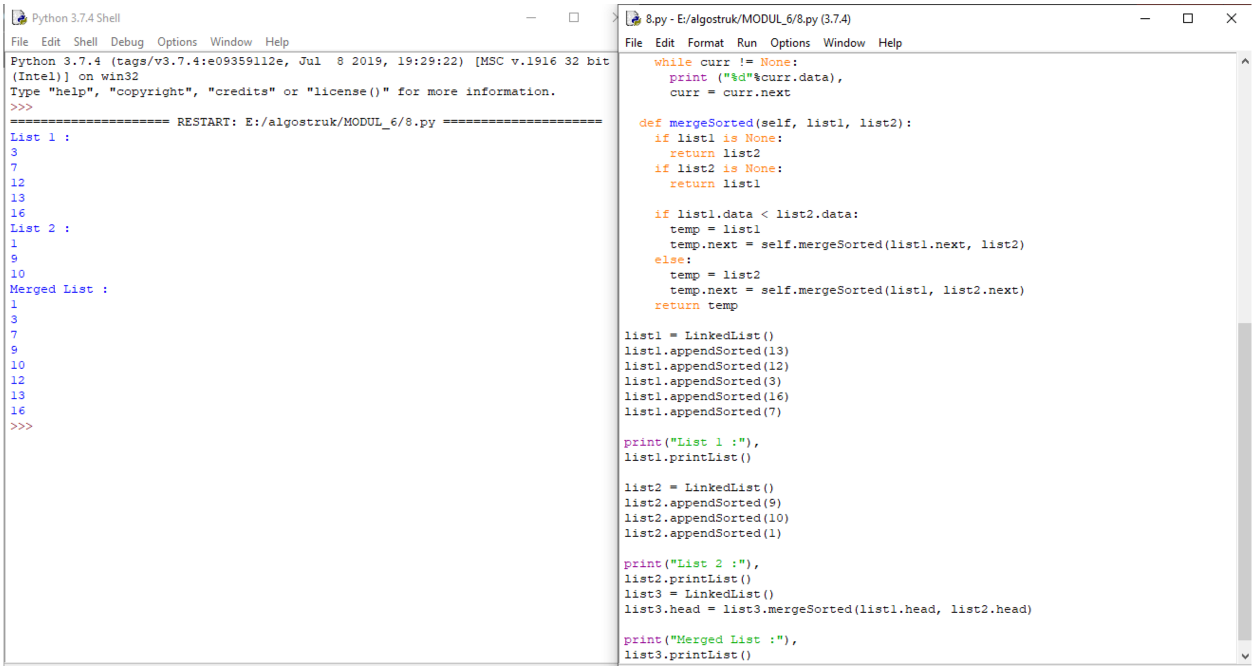
7.


Python 3.7.4 Shell
File Edit Shell Debug Options Window Help
7.py - E:/algostruk/MODUL_6/7.py (3.7.4)
File Edit Format Run Options Window Help

```
Python 3.7.4 (tags/v3.7.4:e09359112e, Jul  8 2019, 19:29:22) [MSC v.191
(Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information
>>> 
==================== RESTART: E:/algostruk/MODUL_6/7.py ==============
merge : 0.0359457 detik
quick : 0.0189481 detik
merge mod : -0.00303316 detik
quick mod : -0.0438828 detik
>>> 
```

```
#Nomor 7
from time import time as detak
from random import shuffle as kocok
import time

k = [i for i in range(1,6001)]
kocok(k)

def mergeSort(arr):
    if len(arr) >1:
        mid = len(arr)//2
        L = arr[:mid]
        R = arr[mid:]
        mergeSort(L)
        mergeSort(R)
        i = j = k = 0
        while i < len(L) and j < len(R):
            if L[i] < R[j]:
                arr[k] = L[i]
                i+=1
            else:
                arr[k] = R[j]
                j+=1
            k+=1
        while i < len(L):
            arr[k] = L[i]
            i+=1
            k+=1
        while j < len(R):
            arr[k] = R[j]
            j+=1
            k+=1

def partition(arr,low,high):
    i = ( low-1 )
    pivot = arr[high]
    for j in range(low , high):
        if   arr[j] <= pivot:
            i = i+1
            arr[i],arr[j] = arr[j],arr[i]
```

8.



Python 3.7.4 Shell
File Edit Shell Debug Options Window Help

```
Python 3.7.4 (tags/v3.7.4:e09359112e, Jul  8 2019, 19:29:22) [MSC v.1916 32 bit
(Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
==================== RESTART: E:/algostruk/MODUL_6/8.py ====================
List 1 :
3
7
12
13
16
List 2 :
1
9
10
Merged List :
1
3
7
9
10
12
13
16
>>>
```

8.py - E:/algostruk/MODUL_6/8.py (3.7.4)
File Edit Format Run Options Window Help

```python
    while curr != None:
      print ("%d"%curr.data),
      curr = curr.next

  def mergeSorted(self, list1, list2):
    if list1 is None:
      return list2
    if list2 is None:
      return list1

    if list1.data < list2.data:
      temp = list1
      temp.next = self.mergeSorted(list1.next, list2)
    else:
      temp = list2
      temp.next = self.mergeSorted(list1, list2.next)
    return temp

list1 = LinkedList()
list1.appendSorted(13)
list1.appendSorted(12)
list1.appendSorted(3)
list1.appendSorted(16)
list1.appendSorted(7)

print("List 1 :"),
list1.printList()

list2 = LinkedList()
list2.appendSorted(9)
list2.appendSorted(10)
list2.appendSorted(1)

print("List 2 :"),
list2.printList()
list3 = LinkedList()
list3.head = list3.mergeSorted(list1.head, list2.head)

print("Merged List :"),
list3.printList()
```