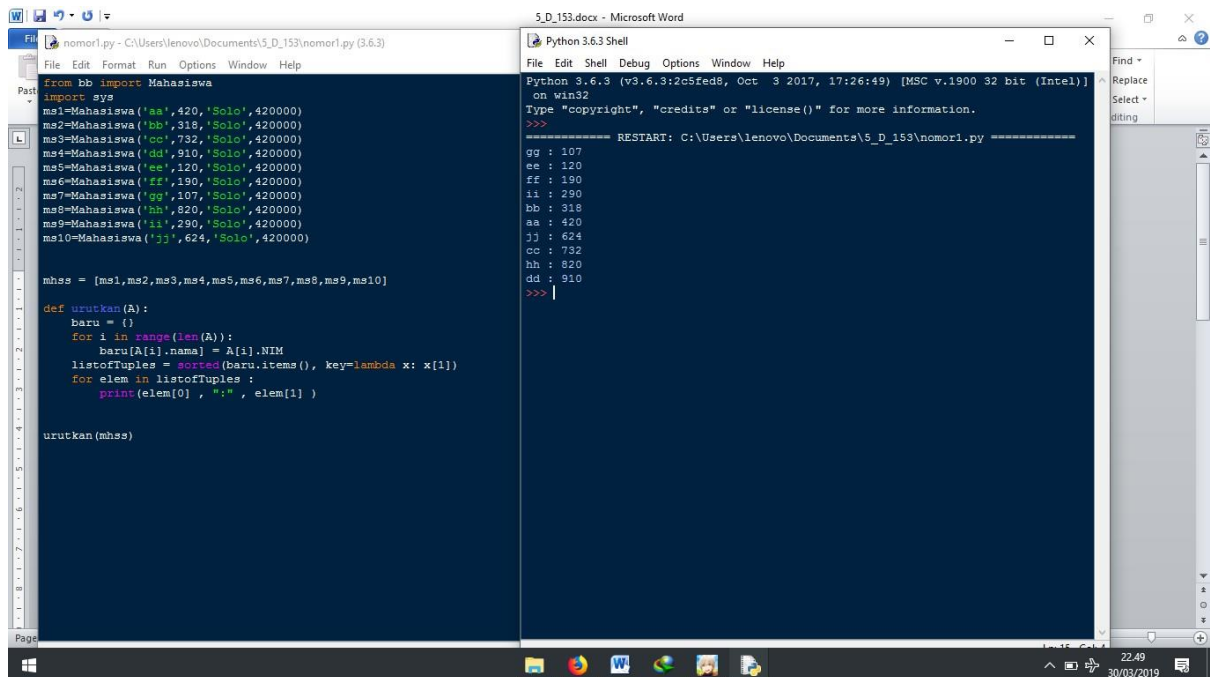


Nama : Yarin Nanditya Anggraeni  
NIM : L200170155  
Kelas : D

## Laporan Praktikum ASD

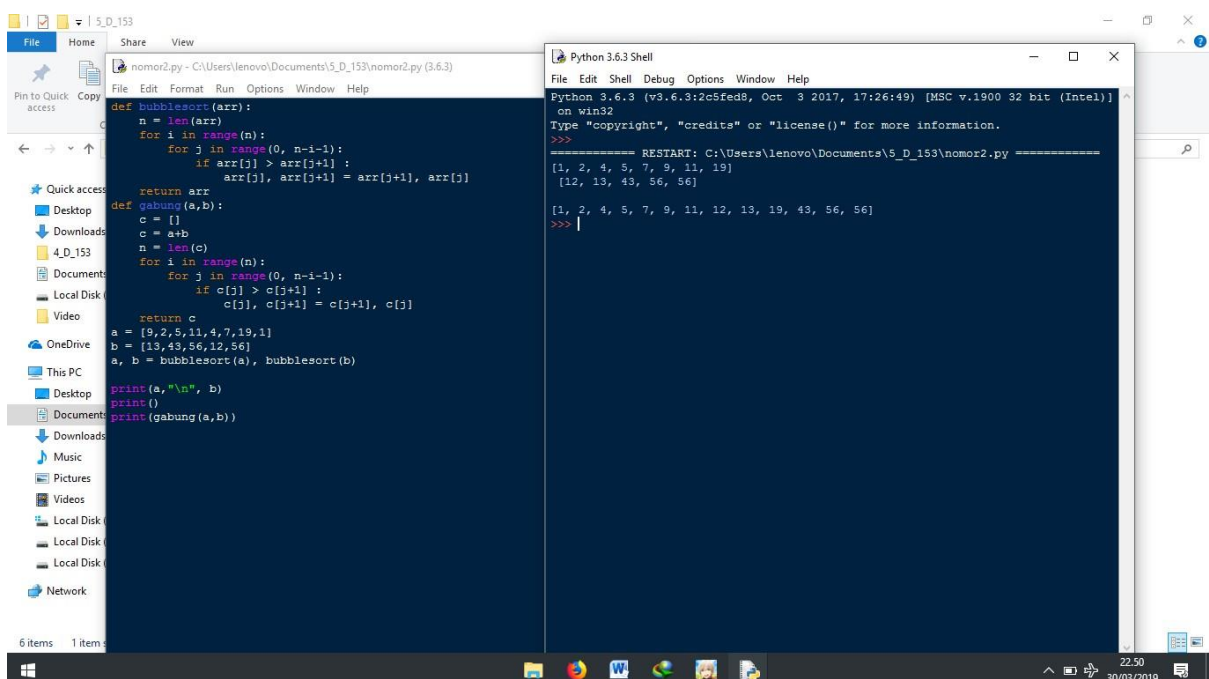
### Modul 5

#### 1.



```
File Edit Shell Debug Options Window Help
Python 3.6.3 Shell
Python 3.6.3 (v3.6.3:2c5fed8, Oct 3 2017, 17:26:49) [MSC v.1900 32 bit (Intel)]
on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:\Users\lenovo\Documents\5_D_153\nomor1.py =====
gg : 107
ee : 120
ff : 190
ii : 290
bb : 318
aa : 420
jj : 624
cc : 732
hh : 820
dd : 910
>>>
```

#### 2.



```
File Edit Shell Debug Options Window Help
Python 3.6.3 Shell
Python 3.6.3 (v3.6.3:2c5fed8, Oct 3 2017, 17:26:49) [MSC v.1900 32 bit (Intel)]
on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:\Users\lenovo\Documents\5_D_153\nomor2.py =====
[1, 2, 4, 5, 7, 9, 11, 19]
[12, 13, 43, 56, 56]
>>>
[1, 2, 4, 5, 7, 9, 11, 12, 13, 19, 43, 56, 56]
>>>
```

3.

The screenshot displays a Windows desktop environment. In the background, a Microsoft Word window titled '5\_D\_153.docx - Microsoft Word' is open, showing a document with Python code. The code defines three sorting functions: `bubb(arr)` for bubble sort, `sele(A)` for selection sort, and `inse(arr)` for insertion sort. The `bubb` function includes comments in Indonesian explaining its logic. The `sele` function finds the minimum element in the remaining unsorted array and swaps it with the first element. The `inse` function inserts each element into its correct position within the sorted sub-array.

In the foreground, a 'Python 3.6.3 Shell' window is open, showing the execution of the `bubb` function. The output indicates that the bubble sort completed in 7.93602 detik. The shell window also shows the file path `C:\Users\lenovo\Documents\5_D_153\nomor3.py` and the Python version `Python 3.6.3 (v3.6.3:2c5fed8, Oct 3 2017, 17:26:49) [MSC v.1900 32 bit (Intel)] on win32`.

```
File Edit Shell Debug Options Window Help
Python 3.6.3 (v3.6.3:2c5fed8, Oct 3 2017, 17:26:49) [MSC v.1900 32 bit (Intel)]
on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:\Users\lenovo\Documents\5_D_153\nomor3.py =====
bubble : 7.93602 detik
selection : 3.87832 detik
insertion : 3.78768 detik
>>>
```

```
File Edit Format Run Options Window Help
5_D_153.docx - Microsoft Word
C:\Users\lenovo\Documents\5_D_153\nomor3.py (3.6.3)
File Edit Format Run Options Window Help
from time import time as detik
from random import shuffle as kocok
x = [i for i in range(1,6001)]
kocok(x)

def bubb(arr):
    n = len(arr)
    # Traverse through all array elements
    for i in range(n):
        # Last i elements are already in place
        for j in range(0, n-i-1):
            # traverse the array from 0 to n-i-1
            # Swap if the element found is greater
            # than the next element
            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)):
        # Find the minimum element in remaining
        # unsorted array
        min_idx = i
        for j in range(i+1, len(A)):
            if A[min_idx] > A[j]:
                min_idx = j
        # Swap the found minimum element with
        # the first element
        A[i], A[min_idx] = A[min_idx], A[i]

def inse(arr):
    # Traverse through 1 to len(arr)
    for i in range(1, len(arr)):
```

Page: 2 of

22:51  
30/03/2019