

Nama : Windiapriani Ginayawati

NIM : L200170157

Kelas : D

Modul:V

1. Program mengurutkan array mahasiswa berdasarkan NIM yang elemennya terbuat dari class MhsTIF.

```
print("No. 1")
class MhsTIF(object):
    def __init__(self, nama, umur, kota, nim):
        self.nama = nama
        self.umur = umur
        self.kotaTinggal = kota
        self.nim = nim

    def __str__(self):
        x = self.nim
        return x

    def nim(self):
        return self.nim

m1 = MhsTIF('Ara', 19, 'Surabaya', 'L200170023')
m2 = MhsTIF('Ari', 19, 'Madiun', 'L200170143')
m3 = MhsTIF('Ina', 20, 'Surakarta', 'L200170157')
m4 = MhsTIF('Ino', 21, 'Klaten', 'L200170111')
m5 = MhsTIF('Ana', 20, 'Salatiga', 'L200170067')
m6 = MhsTIF('Ani', 19, 'Jakarta', 'L200170004')
m7 = MhsTIF('Ika', 19, 'Bogor', 'L200170098')
m8 = MhsTIF('Ike', 21, 'Ngawi', 'L200170123')
m9 = MhsTIF('Aya', 20, 'Malang', 'L200170116')
m10 = MhsTIF('Ayu', 20, 'Surakarta', 'L200170099')
Daftar = [m1, m2, m3, m4, m5, m6, m7, m8, m9, m10]

def insertionSort(A):
    n = len(A)
    for i in range(1, n):
        nilai = A[i]
        pos = i
        while pos > 0 and nilai.nim < A[pos - 1].nim:
            A[pos] = A[pos - 1]
            pos = pos - 1
        A[pos] = nilai

def cekDaftar(z):
    for i in z:
        print(i)

insertionSort(Daftar)
cekDaftar(Daftar)
```

2. Terdapat array A dan B yang sudah urut. Buat program untuk menggabungkan kedua array tersebut menjadi suatu array C yang urut.

```
print("No.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 = [3,7,22,53,1,5,21,9,75]
B = [12,65,14,15,43,23,19]
A, B = bubblesort(A), bubblesort(B)

print("A =", (A))
print("B =", (B))
print("C =", gabung(A,B))
```

3. Bandingkan antara bubble sort, selection sort dan insertion sort, manakah yang lebih cepat.

```
print("No.3")
from time import time as detak
from random import shuffle as kocok

def swap(A,p,q):
    tmp = A[p]
    A[p]= A[q]
    A[q]= tmp

def bubbleSort(A):
    n = len(A)
    for i in range (n-1):
        for j in range(n-i-1):
            if A[j] > A[j+1]:
                swap(A,j,j+1)

def cariPosisiYangTerkecil(A, dariSini, sampaiSini):
    posisiYangTerkecil = dariSini
    for i in range(dariSini+1, sampaiSini):
        if A[i] < A[posisiYangTerkecil]:
            posisiYangTerkecil = i
    return posisiYangTerkecil

def selectionSort(A):
    n = len(A)
    for i in range(n - 1):
        indexKecil = cariPosisiYangTerkecil(A, i, n)
        if indexKecil !=i:
            swap(A, i, indexKecil)

def insertionSort(A):
    n = len(A)
    for i in range(1, n):
        nilai = A[i]
        pos = i
        while pos > 0 and nilai < A[pos - 1]:
            A[pos] = A[pos - 1]
            pos = pos - 1
        A[pos] = nilai

k=[i for i in range (1,6001)]
kocok(k)
u_bub = k[:]
u_sel = k[:]
u_ins = k[:]

aw=detak();bubbleSort(u_bub);ak=detak();print('bubble: %g detik' %(ak-aw));
aw=detak();selectionSort(u_sel);ak=detak();print('selection: %g detik' %(ak-aw));
aw=detak();insertionSort(u_ins);ak=detak();print('insertion:%g detik' %(ak-aw));
```

Hasil setelah di run

```
No. 1
L200170004
L200170023
L200170067
L200170098
L200170099
L200170111
L200170116
L200170123
L200170143
L200170157
No.2
A = [1, 3, 5, 7, 9, 21, 22, 53, 75]
B = [12, 14, 15, 19, 23, 43, 65]
C = [1, 3, 5, 7, 9, 12, 14, 15, 19, 21, 22, 23, 43, 53, 65, 75]
No.3
bubble: 9.39451 detik
selection: 3.81831 detik
insertion:4.69758 detik
^^^
```