

**PRAKTIKUM**  
**ALGORITMA DAN STRUKTUR DATA**  
**MODUL 5**



**Nama : NICKY JULYATRIKA SARI**

**NIM : L200200101**

**PROGRAM STUDI**  
**INFORMATIKA**  
**FAKULTAS KOMUNIKASI DAN INFORMATIKA**  
**UNIVERSITAS MUHAMMADIYAH SURAKARTA**  
**TAHUN 2021/2022**

1. Buatlah suatu program untuk mengurutkan array mahasiswa berdasarkan NIM, yang elemennya terbuat dari class MhsTIF, yang telah kamu buat sebelumnya.

```
nomer 1.py - D:\MATKUL SMT 4\Praktikum ASD\modul 5\nomer 1.py (3.8.6)
File Edit Format Run Options Window Help

#Modul 5 Nicky Julyatrika Sari L200200101
#Nomer 1

class MhsTIF(object):
    def __init__(self,nama,nim,kota,saku):
        self.nama = nama
        self.nim = nim
        self.kota = kota
        self.saku = saku

def insertionSort(nim):
    a = len(nim)
    for i in range(1, a):
        nilai = nim[i]
        posisi = i
        while posisi > 0 and nilai < nim[posisi - 1]:
            nim[posisi] = nim[posisi - 1]
            posisi = posisi - 1
        nim[posisi] = nilai

c0 = MhsTIF('Ika',10,'Sukoharjo', 240000)
c1 = MhsTIF('Budi',19,'Sragen', 230000)
c2 = MhsTIF('Ahmad',12,'Surakarta', 250000)
c3 = MhsTIF('Chandra',17,'Surakarta', 235000)
c4 = MhsTIF('Eka',13,'Boyolali', 240000)
c5 = MhsTIF('Fandi',11,'Salatiga', 250000)
c6 = MhsTIF('Deni',16,'Klaten', 245000)
c7 = MhsTIF('Galuh',18,'Wonogiri', 245000)
c8 = MhsTIF('Janto',20,'Klaten', 245000)
c9 = MhsTIF('Hasan',14,'Karanganyar', 270000)
c10 = MhsTIF('Khalid',15,'Purwodadi', 265000)

#Lalu kita membuat daftar mahasiswa dalam bentuk list seperti ini:

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

def urutanNIM(nim):
    new = []
    for x in nim:
        new.append(x.nim)
    insertionSort(new)
    return new

print(urutanNIM(Daftar))

Python 3.8.6 Shell
File Edit Shell Debug Options Window Help
Python 3.8.6 (tags/v3.8.6:db45529, Sep 23 2020, 15:52:53) [MSC v.1927 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: D:\MATKUL SMT 4\Praktikum ASD\modul 5\nomer 1.py =====
[10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20]
>>>
```

2. Misal terdapat dua buah array yang sudah urut A dan B. Buatlah suatu program untuk menggabungkan, secara efisien, kedua array itu menjadi suatu array C yang urut.

```
*nomer2.py - D:/MATKUL SMT 4/Praktikum ASD/modul 5/nomer2.py (3.8.6)*
File Edit Format Run Options Window Help
#Modul 5 Nicky Julyatrika Sari L200200101
#Nomer 2

def insertionSort(nim):
    a = len(nim)
    for i in range(1, a):
        nilai = nim[i]
        posisi = i
        while posisi > 0 and nilai < nim[posisi - 1]:
            nim[posisi] = nim[posisi - 1]
            posisi = posisi - 1
        nim[posisi] = nilai

a = [11, 1, 50, 10, 22, 48, 34, 21]
b = [8, 43, 19, 25, 17, 13, 67]

def urutanAB(a, b):
    c = a + b
    insertionSort(c)
    return c
print(urutanAB(a,b))

Python 3.8.6 Shell
File Edit Shell Debug Options Window Help
Python 3.8.6 (tags/v3.8.6:db45529, Sep 23 2020, 15:52:53) [MSC v.1927 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: D:/MATKUL SMT 4/Praktikum ASD/modul 5/nomer2.py =====
[1, 8, 10, 11, 13, 17, 19, 21, 22, 25, 34, 43, 48, 50, 67]
>>>
```

3. Kamu mungkin sudah menduga, bubble sort lebih lambat dari selection sort dan juga insertion sort. Tapi manakah yang lebih cepat antara selection sort dan insertion sort? Untuk memulai menyelidikinya, kamu bisa membandingkan waktu yang diperlukan untuk mengurutkan sebuah array yang besar, misal sepanjang 6000 (enam ribu) elemen.

```
*nomer3.py - D:/MATKUL SMT 4/Praktikum ASD/modul 5/nomer3.py (3.8.6)*
File Edit Format Run Options Window Help
#Modul 5 Nicky Julyatrika Sari L200200101
#Nomer 3

from time import time as detik
from random import shuffle as kocok

#Latihan 1
def swap(A,p,q):
    tmp = A[p]
    A[p] = A[q]
    A[q] = tmp

#Latihan 2
def cariPosisiTerkecil(A, dariSini, sampaiSini):
    posisiTerkecil = dariSini
    for i in range(dariSini+1, sampaiSini):
        if A[i] < A[posisiTerkecil]:
            posisiTerkecil = i
    return posisiTerkecil

#Latihan 3
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)
```

```

#Latihan 4
def selectionSort(A):
    n = len(A)
    for i in range(n - 1):
        indexKecil = cariPosisiTerkecil(A, i, n)
        if indexKecil != i:
            swap(A, i, indexKecil)

#Latihan 5
def insertionSort(nim):
    a = len(nim)
    for i in range(1, a):
        nilai = nim[i]
        posisi = i
        while posisi > 0 and nilai < nim[posisi - 1]:
            nim[posisi] = nim[posisi - 1]
            posisi = posisi - 1
        nim[posisi] = nilai

k = list(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_bub);ak=detak();print('selection: %g detik' %(ak-aw));
aw=detak();insertionSort(u_bub);ak=detak();print('insertion: %g detik' %(ak-aw));

```

Ln: 2 Col: 8

Python 3.8.6 Shell

File Edit Shell Debug Options Window Help

Python 3.8.6 (tags/v3.8.6:db45529, Sep 23 2020, 15:52:53) [MSC v.1927 64 bit (AMD64)] on win32  
Type "help", "copyright", "credits" or "license()" for more information.

```

>>>
===== RESTART: D:/MATKUL SMT 4/Praktikum ASD/modul 5/nomer3.py =====
bubble: 6.88857 detik
selection: 2.31157 detik
insertion: 0 detik
>>> |

```