**ALGORITMA DAN STRUKTUR DATA**

**MODUL 5**

**PENGURUTAN**



**Disusun oleh:**

Muhammad Ferizal Fadhli

L200210119

D

**TEKNIK INFORMATIKA**

**FAKULTAS KOMUNIKASI DAN INFORMATIKA**

**UNIVERSITAS MUHAMMADIYAH SURAKARTA**

**2022/2023**

Tugas

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

Berikut Kode Program :

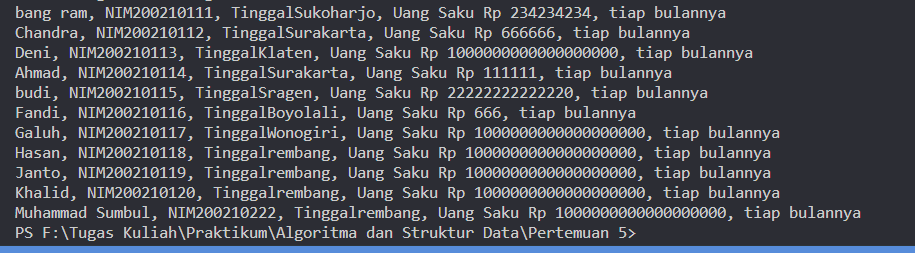
from MhsTIF import \*

baru = sorted(Daftar, *key*=lambda x: x.**NIM**)

for i in baru:

    print(i)

Berikut SS saat program dijalankan :



1. 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.

Berikut Kode Program :

def merge\_sort(arr):

    if len(arr) > 1:

        mid = len(arr) // 2

        left\_arr = arr[:mid]

        right\_arr = arr[mid:]

        merge\_sort(left\_arr)

        merge\_sort(right\_arr)

        i = j = k = 0

        while i < len(left\_arr) and j < len(right\_arr):

            if left\_arr[i] < right\_arr[j]:

                arr[k] = left\_arr[i]

                i += 1

            else:

                arr[k] = right\_arr[j]

                j += 1

            k += 1

        while i < len(left\_arr):

            arr[k] = left\_arr[i]

            i += 1

            k += 1

        while j < len(right\_arr):

            arr[k] = right\_arr[j]

            j += 1

            k += 1

def merge\_sorted\_arrays(arr1, arr2):

    result = []

    i = j = 0

    while i < len(arr1) and j < len(arr2):

        if arr1[i] < arr2[j]:

            result.append(arr1[i])

            i += 1

        else:

            result.append(arr2[j])

            j += 1

    while i < len(arr1):

        result.append(arr1[i])

        i += 1

    while j < len(arr2):

        result.append(arr2[j])

        j += 1

    return result

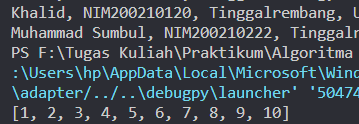
A = [1, 3, 5, 7, 9]

B = [2, 4, 6, 8, 10]

C = merge\_sorted\_arrays(A, B)

print(C)

Berikut SS saat programa dijalanakan :



1. 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?7 Untuk memulai menyelidikinya, kamu bisa membandingkan waktu yang diperlukan untuk mengurutkan sebuah array yang besar, misal sepanjang 6000 (enam ribu) elemen.

Berikut kode program :

from latihan2 import bubbleSort

from latihan3 import selectionSort

from latihan4 import insertionSort

from time import time as detak

from random import shuffle as kocok

k = 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))*;*

Berikut SS program saat dijalanakan :

