

**LAPORAN PRAKTIKUM
PEMROGRAMAN 1
MODUL 7**



Oleh:

Fauzan Wahyu Mubarak

2211104027

SE06A

**PRODI S1 REKAYASA PERANGKAT LUNAK
FAKULTAS INFORMATIKA
INSTITUT TEKNOLOGI TELKOM PURWOKERTO
2023**

I. DASAR TEORI

Algoritma Sorting adalah algoritma untuk meletakkan kumpulan elemen data ke dalam urutan tertentu, berdasarkan satu atau beberapa kunci ke dalam tiap-tiap elemen. Berdasarkan data terurutnya, algoritma sorting dibagi menjadi dua jenis, yaitu:

- Ascending, pengurutan dari nilai terkecil hingga terbesar.

Contoh : a, b, c, d, e.

- Descending, pengurutan dari nilai terbesar hingga terkecil.

Contoh : e, d, c, b, a.

1. PEMBAHASAN

A. Insertion sort

Insertion sort adalah sebuah algoritma pengurutan yang mengurutkannya dicek satu persatu mulai dari yang kedua sampai dengan yang terakhir. Apabila ditemukan data yang lebih kecil dari data sebelumnya, maka data tersebut disisipkan pada posisi yang sesuai. Contoh Insertion Sort:

- **Ascending insertion sort**

Buatlah fungsi dengan nama “insertion_sort” Lalu, jalankan program dengan menuliskan perintah berikut:

```
def insertion_sort(array):  
    for i in range(1, len(array)):  
        item = array[i]  
        j = i - 1  
        while j >= 0 and array[j] > item:  
            array[j + 1] = array[j]  
            j -= 1  
        array[j + 1] = item  
    return array
```

Lalu, jalankan program dengan menuliskan perintah berikut:

```
data_array = [32, 53, 11, 43, 91, 100]
print(insertion_sort(data_array))
```

Output dari sort ascending di atas adalah:

```
[11, 32, 43, 53, 91, 100]
```

- Descending insertion sort

Pada descending sort kita hanya perlu mengubah pembanding “>” menjadi “<” pada `while`.

```
while j >= 0 and array[j] > item:
```

diubah menjadi:

```
while j >= 0 and array[j] < item:
```

Output dari sort descending ketika dijalankan adalah :

```
[100, 91, 53, 43, 32, 11]
```

B. Bubble sort

Bubble Sort adalah algoritma sorting dengan cara membandingkan elemen yang sekarang dengan elemen yang berikutnya. Jika elemen sekarang > elemen berikutnya, maka elemen akan tukar. Jika perbandingan sudah sampai index akhir, maka akan diulang dari index awal. Jika di perbandingan selanjutnya tidak ada perubahan maka sorting selesai.

Proses pengurutan pada algoritma ini dengan membandingkan masing – masing elemen secara berpasangan lalu menukarnya dalam kondisi tertentu. Proses ini akan terus diulang sampai elemen terakhir atau sampai tidak ada lagi elemen yang dapat ditukar. Inilah kenapa algoritma ini diberi nama “Bubble”, dimana gelembung yang terbesar akan naik ke atas.

Contoh Bubble Sort:

- Ascending Bubble Sort

Buatlah fuction dengan nama `bubble_sort`:

```
def bubble_sort(array):
    n = len(array)
    for i in range(n):
        for j in range(n - i - 1):
            if array[j] > array[j + 1]:
                array[j], array[j + 1] = array[j + 1], array[j]
    return array
```

jalankan program dengan menuliskan perintah berikut:

```
data_array = [5, 8, 40, 3, 100, 45]
print(bubble_sort(data_array))
```

output dari sort ascending di atas adalah:

```
[3, 5, 8, 40, 45, 100]
```

- Descending Bubble Sort

Pada descending sort kita hanya perlu mengubah pembandingan ">" menjadi "<" pada if.

```
if array[j] > array[j + 1]:
```

diubah menjadi:

```
if array[j] < array[j + 1]:
```

output dari sort descending ketika dijalankan adalah :

```
[100, 45, 40, 8, 5, 3]
```

C. Selection Sort

Selection Sort adalah algoritma sorting yang mengurutkan data dengan cara mencari elemen paling kecil dari list, lalu menukar elemen tersebut ke urutan paling awal. Selection Sort adalah algoritma sorting dengan cara mengurutkan dengan membandingkan elemen sekarang (ditandai) dengan elemen yang berikutnya sampai terakhir. Jika ditemukan elemen paling kecil, kemudian ditukar dengan elemen sekarang.

Contoh selection sort:

- Ascending Selection Sort

Buatlah function dengan nama selection_sort:

```
def selection_sort(arr):  
    n = len(arr)  
    for i in range(n):  
        min_idx = i  
        for j in range(i+1, n):  
            if arr[min_idx] > arr[j]:  
                min_idx = j  
        arr[i], arr[min_idx] = arr[min_idx], arr[i]  
    return arr
```

jalankan program dengan menuliskan perintah berikut:

```
data_array = [90, 34, 57, 32, 4, 1]  
print(selection_sort(data_array))
```

output dari sort ascending di atas adalah:

```
[1, 4, 32, 57, 34, 90]
```

- Descending Selection Sort

Pada descending short kita hanya perlu mengubah pembanding ">" menjadi "<" pada if.

```
if arr[min_idx] > arr[j]:
```

diubah menjadi:

```
if arr[min_idx] < arr[j]:
```

output dari sort descending ketika dijalankan adalah :

```
[90, 57, 34, 32, 4, 1]
```

II. GUIDED

- A. Buatlah sebuah program penerimaan mahasiswa. Fitur dari program tersebut dapat menambah data mahasiswa, menghapus

data mahasiswa, urutkan data berdasarkan nim, dan cetak seluruh data.

Contoh Output :

1. Tambah data Mahasiswa

```
===== Menu Data Mahasiswa =====  
1. Tambah Data Mahasiswa  
2. Hapus Data Mahasiswa  
3. Urutkan Data Mahasiswa  
4. Lihat Data Mahasiswa  
5. Keluar  
Pilih: 1  
Jumlah Mahasiswa: 3  
Nama Mahasiswa: Kurnia  
Nama Mahasiswa: Tika  
Nama Mahasiswa: Jati
```

2. Hapus data Mahasiswa

```
===== Menu Data Mahasiswa =====  
1. Tambah Data Mahasiswa  
2. Hapus Data Mahasiswa  
3. Urutkan Data Mahasiswa  
4. Lihat Data Mahasiswa  
5. Keluar  
Pilih: 2  
Data Mahasiswa ['Kurnia', 'Tika', 'Jati']  
Hapus Mahasiswa: Tika  
Data Mahasiswa: ['Kurnia', 'Jati']
```

3. Urutkan data Mahasiswa

```
===== Menu Data Mahasiswa =====  
1. Tambah Data Mahasiswa  
2. Hapus Data Mahasiswa  
3. Urutkan Data Mahasiswa  
4. Lihat Data Mahasiswa  
5. Keluar  
Pilih: 3  
['Jati', 'Kurnia']
```

4. Cetak data Mahasiswa

```
===== Menu Data Mahasiswa =====  
1. Tambah Data Mahasiswa  
2. Hapus Data Mahasiswa  
3. Urutkan Data Mahasiswa  
4. Lihat Data Mahasiswa  
5. Keluar  
Pilih: 4  
Nama Mahasiswa: Jati  
Nama Mahasiswa: Kurnia
```

5. Tutup

```
===== Menu Data Mahasiswa =====  
1. Tambah Data Mahasiswa  
2. Hapus Data Mahasiswa  
3. Urutkan Data Mahasiswa  
4. Lihat Data Mahasiswa  
5. Keluar  
Pilih: 5  
Anda Keluar!  
PS D:\Tugas ITTP\Semester 2\Praktikum Pemograman 1\Praktikum
```

6. Code Program:

```
man.py > panggil
def panggil(mhs):
    print("\n===== Menu Data Mahasiswa =====")
    print("1. Tambah Data Mahasiswa ")
    print("2. Hapus Data Mahasiswa ")
    print("3. Urutkan Data Mahasiswa ")
    print("4. Lihat Data Mahasiswa ")
    print("5. Keluar ")

    while True:
        pilih = input("Pilih: ")
        if pilih == '1':
            addmhs(mhs)
            break
        elif pilih == '2':
            removemhs(mhs)
            break
        elif pilih == '3':
            ascmhs(mhs)
            break
        elif pilih == '4':
            viewmhs(mhs)
            break
        elif pilih == '5':
            print("Anda Keluar!")
            return
        else:
            print("Masukkan pilihan yang valid")
            break

#Tambah
def addmhs(mhs):
    while True:
        jumlah = input("Jumlah Mahasiswa: ")
        if not jumlah.isdigit():
            print("Jumlah mahasiswa harus berupa angka")
            continue

        jumlah = int(jumlah)
        if jumlah <= 0:
            print("Jumlah mahasiswa harus lebih besar dari 0")
            continue

        while(jumlah > 0):
            nama = input("Nama Mahasiswa: ")
            mhs.append(nama)
            jumlah = jumlah - 1

        panggil(mhs)
        break
```



```

Latihan.py > panggil
48     break
49
50     #hapus "hapus": Unknown word.
51     def removemhs(mhs): "removemhs": Unknown word.
52         print("Data Mahasiswa %s" %mhs)
53         remove_mhs = input("hapus Mahasiswa: ") "hapus": Unknown word.
54         while True:
55             if remove_mhs in mhs:
56                 mhs.remove(remove_mhs)
57                 print("Data Mahasiswa: %s"%mhs)
58                 panggil(mhs) "panggil": Unknown word.
59                 break
60             else:
61                 print("Nama tersebut tidak ada dalam data")
62                 removemhs(mhs) "removemhs": Unknown word.
63                 break
64
65     #urut "urut": Unknown word.
66     def ascmhs(mhs): "ascmhs": Unknown word.
67         mhs.sort()
68         print(mhs)
69         panggil(mhs) "panggil": Unknown word.
70
71     #lihat "lihat": Unknown word.
72     def viewmhs(mhs): "viewmhs": Unknown word.
73         if not mhs :
74             print("Belum ada mahasiswa") "Belum": Unknown word.
75         else:
76             for x in mhs:
77                 print("Nama Mahasiswa: %s" %x)
78                 panggil(mhs) "panggil": Unknown word.
79
80     mhs = []
81     panggil(mhs)

```

III. UNGUIDED

Tugas 1

Pada suatu kelas terdapat 5 mahasiswa. Pada akhir semester mereka menerima lembar Indeks Prestasi Semester (IPS), masing-masing mahasiswa tersebut memiliki IPS sebagai berikut: {3.8, 2.9, 3.3, 4.0, 2.7}. Buatlah program untuk mengurutkan IPS mahasiswa tersebut dari yang terbesar hingga terkecil dengan menggunakan algoritma **Bubble sort** !

- Source Code :

```
Tugas1.py > ...
1 print("\nIndeks Prestasi Semester (IPS)")      "Indeks": Unknown word.
2
3 def bubble_sort(array):
4     for i in range(len(array)):
5         for j in range(len(array)-i-1):
6             if array[j] < array[j + 1]:
7                 array[j], array[j + 1] = array[j + 1], array[j]
8     return array
9
10
11 list_1 = [3.8, 2.9, 3.3, 4.0, 2.7]
12 print(f"List Sebelum Diurutkan: {list_1}")      "Sebelum": Unknown word.
13 bubble_sort(list_1)
14 print(f"List Setelah Diurutkan: {list_1}")      "Setelah": Unknown word.
```

- Output :

```
Indeks Prestasi Semester (IPS)
List Sebelum Diurutkan: [3.8, 2.9, 3.3, 4.0, 2.7]
List Setelah Diurutkan: [4.0, 3.8, 3.3, 2.9, 2.7]
PS D:\Tugas IITP\Semester 2\Praktikum Pemograman 1\Praktikum 7> |
```

Tugas B

Pada suatu organisasi memiliki 10 anggota dengan nama masing-masing: *Zhafira, Nirmala, Aksara, Nalendra, Cakra, Sastra, Agni, Bagas, Jerome, Kiara*. Supaya mudah dalam melakukan pencarian, Ketua organisasi akan mengurutkan nama – nama tersebut sesuai dengan alfabet. Buatlah program untuk membantu Pain dengan menggunakan algoritma **Selection Sort**!

- Source Code :

```
Tugas2.py > ...
1 print("\nNama 10 Anggota Organisasi") "Anggota": Unknown word.
2
3 def selection_sort(array):
4     for i in range(len(array)):
5         min_index = i
6         for j in range(i + 1, len(array)):
7             if array[min_index] > array[j]:
8                 min_index = j
9         array[i],array[min_index] = array[min_index], array[i]
10    return array
11
12
13 list_1 = ["Zhafira", "Nirmala", "Aksara", "Nalendra", "Cakra", "Sastra", "Agni", "Bagas", "Jerome", "Kiara"] "Zhafira": Unknown word
14 print(f"before : {list_1}")
15 selection_sort(list_1)
16 print(f"After : {list_1}")
17
```

- Output :

```
PS D:\Tugas IITP\Semester 2\Praktikum Pemograman 1\Praktikum 7> & C:/Users/Hp/AppData/Local/Microsoft/WindowsApps/python
Nama 10 Anggota Organisasi
before : ['Zhafira', 'Nirmala', 'Aksara', 'Nalendra', 'Cakra', 'Sastra', 'Agni', 'Bagas', 'Jerome', 'Kiara']
After : ['Agni', 'Aksara', 'Bagas', 'Cakra', 'Jerome', 'Kiara', 'Nalendra', 'Nirmala', 'Sastra', 'Zhafira']
PS D:\Tugas IITP\Semester 2\Praktikum Pemograman 1\Praktikum 7>
```

Tugas C

Buatlah program untuk menginput nama buku lalu muncul pilihan jenis sorting (**dengan Insertion Sort**).

- Source Code :

```
Tugas3.py > ...
1 def insertionasc_sort(array):    "insertionasc": Unknown word.
2     for i in range (1, len(array)):
3         item = array [i]
4         j = i - 1
5
6         while j >= 0 and array[j] > item:
7             array[j + 1] = array[j]
8             j -= 1
9
10            array[j + 1] = item
11
12
13 def insertiondsc_sort(array):    "insertiondsc": Unknown word.
14     for i in range (1, len(array)):
15         item = array [i]
16         j = i - 1
17
18         while j >= 0 and array[j] < item:
19             array[j + 1] = array[j]
20             j -= 1
21
22            array[j + 1] = item
23
24
25 def tambah_buku(array):    "tambah": Unknown word.
26     total_buku = int(input("Masukkan Total Buku : "))    "buku": Unknown word.
27     n = 1
28     while total_buku > 0:    "buku": Unknown word.
29         nama_buku = input(f"Masukkan judul buku ke- {n} : ")    "buku": Unknown word.
30         array.append(nama_buku)
31         n += 1
32         total_buku -= 1
33
34 def urut_sort(buku):    "urut": Unknown word.
35     print("\n===== Urutkan?=====")    "Urutkan": Unknown word.
36     print("1. Sorting Ascending")
37     print("2. Sorting Descending")
38
39     pilihan = input("Masukkan pilihan: ")    "pilihan": Unknown word.
40
41     if pilihan == '1':    "pilihan": Unknown word.
42         insertionasc_sort(buku)    "insertionasc": Unknown word.
43         print("\nDaftar buku setelah diurutkan secara ascending:")    "Daftar": Unknown word.
44
45     elif pilihan == '2':    "pilihan": Unknown word.
46         insertiondsc_sort(buku)    "insertiondsc": Unknown word.
47         print("\nDaftar buku setelah diurutkan secara descending:")    "Daftar": Unknown word.
48
49     else:
50         print("Pilihan tidak valid")    "Pilihan": Unknown word.
51
52 def cetak_hasil(array):    "cetak": Unknown word.
53     print(" ")
54     for i, n in enumerate(array):
55         print(f"Judul buku ke- {i+1} : {n}")    "Judul": Unknown word.
56
57 data_buku = []
58 tambah_buku(data_buku)    "tambah": Unknown word.
59 urut_sort(data_buku)    "urut": Unknown word.
60 cetak_hasil(data_buku)    "cetak": Unknown word.
61
62
```

- Output secara Ascending :

```
PS D:\Tugas ITTP\Semester 2\Praktikum Pemograman 1\Praktikum 7> & C
Masukkan Total Buku : 5
Masukkan judul buku ke- 1 : Laut Bercerita
Masukkan judul buku ke- 2 : Bumi Manusia
Masukkan judul buku ke- 3 : Geez & Ann
Masukkan judul buku ke- 4 : Hujan
Masukkan judul buku ke- 5 : Garis Waktu

===== Urutkan?=====
1. Sorting Ascending
2. Sorting Descending
Masukkan pilihan: 1

Daftar buku setelah diurutkan secara ascending:

Judul buku ke- 1 : Bumi Manusia
Judul buku ke- 2 : Garis Waktu
Judul buku ke- 3 : Geez & Ann
Judul buku ke- 4 : Hujan
Judul buku ke- 5 : Laut Bercerita
PS D:\Tugas ITTP\Semester 2\Praktikum Pemograman 1\Praktikum 7> |
```

- Output secara Descending :

```
PS D:\Tugas ITTP\Semester 2\Praktikum Pemograman 1\Praktikum 7> & C
Masukkan Total Buku : 5
Masukkan judul buku ke- 1 : Laut Bercerita
Masukkan judul buku ke- 2 : Bumi Manusia
Masukkan judul buku ke- 3 : Geez & Ann
Masukkan judul buku ke- 4 : Hujan
Masukkan judul buku ke- 5 : Garis Waktu

===== Urutkan?=====
1. Sorting Ascending
2. Sorting Descending
Masukkan pilihan: 2

Daftar buku setelah diurutkan secara descending:

Judul buku ke- 1 : Laut Bercerita
Judul buku ke- 2 : Hujan
Judul buku ke- 3 : Geez & Ann
Judul buku ke- 4 : Garis Waktu
Judul buku ke- 5 : Bumi Manusia
PS D:\Tugas ITTP\Semester 2\Praktikum Pemograman 1\Praktikum 7> |
```