

Laporan Praktikum

SORTING (BUBBLE, SELECTION, DAN INSERTION SORT)



Disusun oleh:

Tomi Martino Affandi

2341720245

D-IV Teknik Informatika - 1E

Jl. Soekarno Hatta No.9, Jatimulyo, Kec. Lowokwaru, Kota Malang Jawa Timur 65141

Phone : (0341)404424, 404425

E-mail : Polinema.ac.id

5.1 Tujuan Praktikum

Setelah melakukan praktikum ini diharapkan mahasiswa mampu:

- Mahasiswa mampu membuat algoritma searching bubble sort, selection sort dan insertion sort
- Mahasiswa mampu menerapkan algoritma searching bubble sort, selection sort dan insertion sort pada program

5.2 Mengurutkan Data Mahasiswa Berdasarkan IPK Menggunakan Bubble Sort

Code:

<https://github.com/TomiMartinoAffandi/Algoritma---Struktur-Data/commit/0a7f21cf5201b79c0926c283cbcdf470ec721e4c>

Output:

Data mahasiswa sebelum sorting:

```
Nama: Nusa
Tahun Masuk: 2017
Umur: 25
IPK: 3.0
-----
Nama: Rara
Tahun Masuk: 2012
Umur: 19
IPK: 4.0
-----
Nama: Dompus
Tahun Masuk: 2018
Umur: 19
IPK: 3.5
-----
Nama: nusa
Tahun Masuk: 2017
Umur: 23
IPK: 2.0
-----
Nama: nusa
Tahun Masuk: 2019
Umur: 21
IPK: 3.75
-----
```

Data mahasiswa setelah sorting desc berdasarkan ipk:

```
Nama: Rara
Tahun Masuk: 2012
Umur: 19
IPK: 4.0
-----
Nama: nusa
Tahun Masuk: 2019
Umur: 21
IPK: 3.75
-----
Nama: Dompus
Tahun Masuk: 2018
Umur: 19
IPK: 3.5
-----
Nama: Nusa
Tahun Masuk: 2017
Umur: 25
IPK: 3.0
-----
Nama: nusa
Tahun Masuk: 2017
Umur: 23
IPK: 2.0
-----
```

Pertanyaan

1. Terdapat di method bubbleSort(); didalam kelas daftarMhsBerprestasi{ }
2. Untuk membandingkan apakah value index j lebih besar daripada index j-1 jika benar maka akan dilakukan swap. value index j menjadi index j-1 dan value index j-1 berubah menjadi value index j
3.
 - a. -Perulangan i digunakan untuk mengontrol jumlah iterasi pada seluruh array.
-Perulangan j digunakan untuk membandingkan elemen array j dengan array j-1 dalam array dan melakukan pertukaran jika diperlukan. Perulangan j bergantung pada iterasi i karena setiap iterasi i mengurangi jumlah elemen yang perlu dibandingkan pada setiap iterasi j
 - b. karena pada iterasi terakhir, hanya satu elemen yang perlu diposisikan, sehingga tidak perlu lagi melakukan perbandingan
 - c. karena setiap iterasi i akan memposisikan setidaknya satu elemen ke posisi akhir yang sesuai. Oleh karena itu, setelah setiap iterasi i, elemen terakhir yang telah diurutkan tidak perlu lagi dibandingkan.
 - d. Jika banyak data dalam listMhs adalah 50, maka perulangan i akan berlangsung sebanyak 49 kali. Ini karena setiap iterasi i memposisikan setidaknya satu elemen ke posisi akhir yang sesuai, dan setelah iterasi terakhir, array akan terurut secara keseluruhan. Jumlah tahap (iterasi i) yang diperlukan untuk menyelesaikan bubble sort pada data dengan 50 elemen adalah 49.

5.3 Mengurutkan Data Mahasiswa Berdasarkan IPK Menggunakan Selection Sort

Code:

<https://github.com/TomiMartinoAffandi/Algoritma---Struktur-Data/commit/f1173c8bad0c00ac363cd22a8dfe2ea05156da8e>

Output:

Data mahasiswa setelah sorting asc berdasarkan ipk:

Nama: nusa

Tahun Masuk: 2017

Umur: 23

IPK: 2.0

Nama: Nusa

Tahun Masuk: 2017

Umur: 25

IPK: 3.0

Nama: Dompu

Tahun Masuk: 2018

Umur: 19

IPK: 3.5

Nama: nusa

Tahun Masuk: 2019

Umur: 21

IPK: 3.75

Nama: Rara

Tahun Masuk: 2012

Umur: 19

IPK: 4.0

Pertanyaan

1. Untuk mencari nilai terkecil dalam suatu array, yaitu pertama tama set value idxMin = i, dalam kasus ini i = 0, lalu membandingkan value index array j[1] apakah lebih kecil dari index array[idxMin], jika benar akan dilakukan update yaitu idxMin = j.

5.4 Mengurutkan Data Mahasiswa Berdasarkan IPK Menggunakan Insertion Sort

Code:

<https://github.com/TomiMartinoAffandi/Algoritma---Struktur-Data/commit/20e279a4116103e720d4113c09230ea41b5fcd12>

Output:

Data mahasiswa setelah sorting asc berdasarkan ipk:

Nama: nusa

Tahun Masuk: 2017

Umur: 23

IPK: 2.0

Nama: Nusa

Tahun Masuk: 2017

Umur: 25

IPK: 3.0

Nama: Dompus

Tahun Masuk: 2018

Umur: 19

IPK: 3.5

Nama: nusa

Tahun Masuk: 2019

Umur: 21

IPK: 3.75

Nama: Rara

Tahun Masuk: 2012

Umur: 19

IPK: 4.0

Pertanyaan

1. Code:

<https://github.com/TomiMartinoAffandi/Algoritma---Struktur-Data/commit/ee34d289fe88bee8620bea01e0320c39e70df8fb>

Latihan Praktikum

1. Code:

<https://github.com/TomiMartinoAffandi/Algoritma---Struktur-Data/commit/1018943fb1e197af63463a8c3c0023b6e837d9e2>

Output:

Data Hotel sebelum sorting:

Nama: Swiss-Belinn

Kota: Malang

Harga: 550000

Bintang: 5

Nama: Everyday Smart

Kota: jakarta

Harga: 260000

Bintang: 4

Nama: Oyo

Kota: Yogyakarta

Harga: 70000

Bintang: 1

Nama: Hotel Syariah 99

Kota: Bandung

Harga: 230000

Bintang: 3

Nama: Embun Pagi

Kota: Blitar

Harga: 160000

Bintang: 2

Data Hotel setelah sorting asc berdasarkan Harga:

Nama: Oyo

Kota: Yogyakarta

Harga: 70000

Bintang: 1

Nama: Embun Pagi

Kota: Blitar

Harga: 160000

Bintang: 2

Nama: Hotel Syariah 99

Kota: Bandung

Harga: 230000

Bintang: 3

Nama: Everyday Smart

Kota: jakarta

Harga: 260000

Bintang: 4

Nama: Swiss-Belinn

Kota: Malang

Harga: 550000

Bintang: 5

Data Hotel setelah sorting desc berdasarkan Bintang:

Nama: Swiss-Belinn

Kota: Malang

Harga: 550000

Bintang: 5

Nama: Everyday Smart

Kota: jakarta

Harga: 260000

Bintang: 4

Nama: Hotel Syariah 99

Kota: Bandung

Harga: 230000

Bintang: 3

Nama: Embun Pagi

Kota: Blitar

Harga: 160000

Bintang: 2

Nama: Oyo

Kota: Yogyakarta

Harga: 70000

Bintang: 1
