



NAMA : Adani Salsab
NIM : 2341720123
KELAS : 1-G
MATERI : Sorting (Jobsheet 5)

1. Praktikum 1

```
src > P6 > ... DaftarMahasiswaBerprestasi.java
1 package P6;
2
3 public class DaftarMahasiswaBerprestasi {
4     Mahasiswa listMhs[] = new Mahasiswa[5];
5     int idx;
6
7     // method tambah()
8
9     void tambah(Mahasiswa m) {
10         if (idx < listMhs.length) {
11             listMhs[idx] = m;
12             idx++;
13         } else {
14             System.out.println(x:"Data sudah penuh!!");
15         }
16     }
17
18     // method tampil()
19
20     void tampil() {
21         for (Mahasiswa m : listMhs) {
22             m.tampil();
23             System.out.println(x:"-----");
24         }
25     }
26
27     // method bubbleSort()
28
29     void bubbleSort() {
30         for (int i = 0; i < listMhs.length - 1; i++) {
31             for (int j = 0; j < listMhs.length - i - 1; j++) {
32                 if (listMhs[j].ipk > listMhs[j + 1].ipk) {
33                     // proses swap
34                     Mahasiswa tmp = listMhs[j];
35                     listMhs[j] = listMhs[j + 1];
36                     listMhs[j + 1] = tmp;
37                 }
38             }
39         }
40     }
41 }
42
```

```
src > P6 > J Main.java > Main > main(String[])
1 package P6;
2
3 import java.util.Scanner;
4
5 public class Main {
6     Run | Debug
7     public static void main(String[] args) {
8         Scanner sc = new Scanner(System.in);
9
10         DaftarMahasiswaBerprestasi list = new DaftarMahasiswaBerprestasi();
11         Mahasiswa m1 = new Mahasiswa(n:"Nusa", t:2017, u:25, i:3);
12         Mahasiswa m2 = new Mahasiswa(n:"Rara", t:2012, u:19, i:4);
13         Mahasiswa m3 = new Mahasiswa(n:"Dompu", t:2018, u:19, i:3.5);
14         Mahasiswa m4 = new Mahasiswa(n:"Abdul", t:2017, u:23, i:2);
15         Mahasiswa m5 = new Mahasiswa(n:"Ummi", t:2019, u:21, i:3.75);
16
17         list.tambah(m1);
18         list.tambah(m2);
19         list.tambah(m3);
20         list.tambah(m4);
21         list.tambah(m5);
22
23         System.out.println(x:"Data Mahasiswa sebelum sorting:");
24         list.tampil();
25
26         System.out.println(x:"Data Mahasiswa setelah sorting desc berdasarkan ipk:");
27         list.bubbleSort();
28         list.tampil();
29     }
30 }

```



NAMA : Adani Salsab
NIM : 2341720123
KELAS : 1-G
MATERI : Sorting (Jobsheet 5)

```
PS D:\college\semester 2\Algoritma dan Struktur Data_16_01> & "C:\Program Files\Java\jdk-20\bin\java.exe" "-XX:+ShowCodeDetailsInExceptionMessages" "-cp" "D:\college\semester 2\Algoritma dan Struktur Data_16_01\bin" "P6.Main"
Data Mahasiswa sebelum sorting:
Nama = Nisa
Tahun Masuk = 2017
Umur = 25
IPK = 3,0
-----
Nama = Rara
Tahun Masuk = 2012
Umur = 19
IPK = 4,0
-----
Nama = Dampu
Tahun Masuk = 2018
Umur = 19
IPK = 3,5
-----
Nama = Abdul
Tahun Masuk = 2017
Umur = 23
IPK = 2,0
-----
Nama = Umi
Tahun Masuk = 2019
Umur = 21
IPK = 3,75
-----
Data Mahasiswa setelah sorting desc berdasarkan ipk:
Nama = Abdul
Tahun Masuk = 2017
Umur = 23
IPK = 2,0
-----
Nama = Nisa
Tahun Masuk = 2017
Umur = 25
IPK = 3,0
-----
Nama = Dampu
Tahun Masuk = 2018
Umur = 19
IPK = 3,5
-----
Nama = Umi
Tahun Masuk = 2019
Umur = 21
IPK = 3,75
-----
Nama = Rara
Tahun Masuk = 2012
Umur = 19
IPK = 4,0
-----
```

Pertanyaan!

1. Terdapat di method apakah proses bubble sort?

Jawab : Ada di method bubbleSort(). Kode :

```
// method bubbleSort()

void bubbleSort() {
    for (int i = 0; i < listMhs.length - 1; i++) {
        for (int j = 0; j < listMhs.length - i - 1; j++) {
            if (listMhs[j].ipk > listMhs[j + 1].ipk) {
                // proses swap
                Mahasiswa tmp = listMhs[j];
                listMhs[j] = listMhs[j + 1];
                listMhs[j + 1] = tmp;
            }
        }
    }
}
```

2. Di dalam method bubbleSort(), terdapat baris program seperti di bawah ini:

```
if (listMhs[j].ipk > listMhs[j + 1].ipk) {
    // proses swap
    Mahasiswa tmp = listMhs[j];
    listMhs[j] = listMhs[j + 1];
    listMhs[j + 1] = tmp;
}
```



NAMA : Adani Salsab
NIM : 2341720123
KELAS : 1-G
MATERI : Sorting (Jobsheet 5)

```
}
```

Untuk apakah proses tersebut?

Jawab : Proses tersebut adalah pertukaran atau swap antara dua elemen dalam array `listMhs` jika kondisi tertentu terpenuhi. Jika nilai `ipk` dari `listMhs[j]` lebih besar dari nilai `ipk` dari `listMhs[j + 1]` (elemen berikutnya), maka kedua elemen tersebut ditukar posisinya. Proses swap memastikan bahwa elemen dengan nilai `ipk` yang lebih kecil akan bergerak ke posisi yang lebih awal dalam array, sedangkan elemen dengan nilai `ipk` yang lebih besar akan bergerak ke posisi yang lebih belakang.

3. Perhatikan perulangan di dalam bubbleSort() di bawah ini:

```
for (int i = 0; i < listMhs.length - 1; i++) {  
    for (int j = 0; j < listMhs.length - i - 1; j++) {
```

- a. Apakah perbedaan antara kegunaan perulangan i dan perulangan j?

Jawab : Perbedaan antara penggunaan perulangan `i` dan `j` adalah bahwa perulangan `i` mengatur jumlah iterasi keseluruhan sementara perulangan `j` mengatur perbandingan dan pertukaran antara elemen-elemen dalam setiap iterasi `i`.

- b. Mengapa syarat dari perulangan i adalah $i < \text{listMhs.length} - 1$?

Jawab : Syarat dari perulangan `i` adalah $i < \text{listMhs.length} - 1$ karena algoritma Bubble Sort memerlukan `n-1` iterasi untuk mengurutkan sebuah array dengan `n` elemen. Pada setiap iterasi, elemen terakhir akan berada pada posisi yang tepat setelah dilakukan `n-1` iterasi pertama. Oleh karena itu, tidak perlu lagi melakukan iterasi setelah itu karena elemen terakhir sudah pasti sudah berada pada posisi yang benar.

- c. Mengapa syarat dari perulangan j adalah $j < \text{listMhs.length} - i - 1$?

Jawab : Syarat dari perulangan j, yaitu $j < \text{listMhs.length} - i - 1$, dipilih karena pada setiap iterasi dari loop luar (iterasi i), elemen terbesar (dengan nilai ipk terbesar) akan tergeser ke posisi terakhir, sehingga tidak perlu lagi membandingkan elemen tersebut dengan elemen-elemen lainnya yang telah diurutkan sebelumnya. Cara ini mengoptimalkan kinerja algoritma Bubble Sort dengan mengurangi jumlah perbandingan yang tidak perlu pada setiap iterasi.

- d. Jika banyak data di dalam listMhs adalah 50, maka berapakah perulangan i akan berlangsung? Dan ada berapa Tahap bubble sort yang ditempuh?

Jawab : Jika terdapat 50 data dalam array `listMhs`, maka jumlah iterasi `i` akan berlangsung sebanyak 49 kali. Hal ini karena jumlah iterasi akan terus berkurang seiring dengan berjalannya algoritma, dimana setiap iterasi `i` akan menempatkan satu elemen terbesar ke posisi akhir. Maka, jumlah tahap yang ditempuh akan menjadi 49. Sehingga, jumlah iterasi `i` yang berlangsung sebanyak 49 kali sesuai dengan jumlah tahap dalam algoritma Bubble Sort.



NAMA : Adani Salsab
NIM : 2341720123
KELAS : 1-G
MATERI : Sorting (Jobsheet 5)

2. Praktikum 2

```
src > P6 > J DaftarMahasiswaBerprestasi.java > DaftarMahasiswaBerprestasi
3 public class DaftarMahasiswaBerprestasi {
27 // method bubbleSort()
28
29 void bubbleSort() {
30     for (int i = 0; i < listMhs.length - 1; i++) {
31         for (int j = 0; j < listMhs.length - i - 1; j++) {
32             if (listMhs[j].ipk > listMhs[j + 1].ipk) {
33                 // proses swap
34                 Mahasiswa tmp = listMhs[j];
35                 listMhs[j] = listMhs[j + 1];
36                 listMhs[j + 1] = tmp;
37             }
38         }
39     }
40 }
41
42 // method selectionSort()
43 void selectionSort() {
44     for (int i = 0; i < listMhs.length - 1; i++) {
45         int idxMin = i;
46         for (int j = i + 1; j < listMhs.length; j++) {
47             if (listMhs[j].ipk < listMhs[idxMin].ipk) {
48                 idxMin = j;
49             }
50         }
51         // swap
52         Mahasiswa tmp = listMhs[idxMin];
53         listMhs[idxMin] = listMhs[i];
54         listMhs[i] = tmp;
55     }
56 }
57 }
```

```
P6 > J Main.java > Main > main(String[])
1 package P6;
2
3 import java.util.Scanner;
4
5 public class Main {
6     Run | Debug
7     public static void main(String[] args) {
8         Scanner sc = new Scanner(System.in);
9
10        DaftarMahasiswaBerprestasi list = new DaftarMahasiswaBerprestasi();
11        Mahasiswa m1 = new Mahasiswa(n:"Nusa", t:2017, u:25, i:3);
12        Mahasiswa m2 = new Mahasiswa(n:"Rara", t:2012, u:19, i:4);
13        Mahasiswa m3 = new Mahasiswa(n:"Dampu", t:2018, u:19, i:3.5);
14        Mahasiswa m4 = new Mahasiswa(n:"Abdul", t:2017, u:23, i:2);
15        Mahasiswa m5 = new Mahasiswa(n:"Umi", t:2019, u:21, i:3.75);
16
17        list.tambah(m1);
18        list.tambah(m2);
19        list.tambah(m3);
20        list.tambah(m4);
21        list.tambah(m5);
22
23        System.out.println(x:"Data Mahasiswa sebelum sorting:");
24        list.tampil();
25
26        System.out.println(x:"Data Mahasiswa setelah sorting desc berdasarkan ipk:");
27        list.bubbleSort();
28        list.tampil();
29
30        System.out.println(x:"Data Mahasiswa setelah sorting asc berdasarkan ipk:");
31        list.selectionSort();
32        list.tampil();
33    }
34 }
```

```
PS D:\college\semester 2\Algoritma dan Struktur Data_16_01> cd .\di\c
ollege\semester 2\Algoritma dan Struktur Data_16_01; & "C:\Program Files
\Java\jdk-20\bin\java.exe" "-XX:ShowCodeDetailsInExceptionMessages" "-cp
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 = Dampu
Tahun Masuk = 2018
Umur = 19
IPK = 3.5
-----
Nama = Abdul
Tahun Masuk = 2017
Umur = 23
IPK = 2.0
-----
Nama = Umi
Tahun Masuk = 2019
Umur = 21
IPK = 3.75
-----
```



NAMA : Adani Salsab
NIM : 2341720123
KELAS : 1-G
MATERI : Sorting (Jobsheet 5)

```
-----  
Data Mahasiswa setelah sorting desc berdasarkan ipk:  
Nama = Abdul  
Tahun Masuk = 2017  
Umur = 23  
IPK = 2.0  
-----  
Nama = Nisa  
Tahun Masuk = 2017  
Umur = 25  
IPK = 3.0  
-----  
Nama = Dimpu  
Tahun Masuk = 2018  
Umur = 19  
IPK = 3.5  
-----  
Nama = Umi  
Tahun Masuk = 2019  
Umur = 21  
IPK = 3.75  
-----  
Nama = Rara  
Tahun Masuk = 2012  
Umur = 19  
IPK = 4.0  
-----
```

```
-----  
Data Mahasiswa setelah sorting asc berdasarkan ipk  
Nama = Abdul  
Tahun Masuk = 2017  
Umur = 23  
IPK = 2.0  
-----  
Nama = Nisa  
Tahun Masuk = 2017  
Umur = 25  
IPK = 3.0  
-----  
Nama = Dimpu  
Tahun Masuk = 2018  
Umur = 19  
IPK = 3.5  
-----  
Nama = Umi  
Tahun Masuk = 2019  
Umur = 21  
IPK = 3.75  
-----  
Nama = Rara  
Tahun Masuk = 2012  
Umur = 19  
IPK = 4.0  
-----  
PS D:\college\semester 2\Algoritma dan Struktur Data_16_01> |
```

Pertanyaan!

1. Di dalam method selection sort, terdapat baris program seperti di bawah ini:

```
int idxMin = i;  
for (int j = i + 1; j < listMhs.length; j++) {  
    if (listMhs[j].ipk < listMhs[idxMin].ipk) {  
        idxMin = j;  
    }  
}
```

Untuk apakah proses tersebut, jelaskan!

Jawab : Proses Algoritma Selection Sort bekerja dengan cara mencari elemen terkecil dari sisa array yang belum terurut dan menukarnya dengan elemen terkecil tersebut. Proses ini terjadi dalam loop di mana variabel `j` digunakan untuk mencari elemen terkecil dari elemen yang belum terurut. Proses ini berlanjut sampai seluruh array terurut dengan elemen-elemen terurut secara bertambah dari indeks awal.



NAMA : Adani Salsab
NIM : 2341720123
KELAS : 1-G
MATERI : Sorting (Jobsheet 5)

3. Praktikum 3

```
P6 > J Pangkat.java J MainPangkat.java 1 J Sum.java J Showroom.java J Mahasiswa.java ...P6 J DaftarMahasiswaBerprestasi.java X J
> P6 > J DaftarMahasiswaBerprestasi.java > DaftarMahasiswaBerprestasi > insertionSort()
3 public class DaftarMahasiswaBerprestasi {
4
5 // method insertionSort()
6
7 void insertionSort() {
8     for (int i = 0; i < listMhs.length; i++) {
9         Mahasiswa temp = listMhs[i];
10        int j = i;
11        while (j > 0 && listMhs[j - 1].ipk > temp.ipk) {
12            listMhs[j] = listMhs[j - 1];
13            j--;
14        }
15        listMhs[j] = temp;
16    }
17 }
18 }
19 }
```

```
P6 > J Pangkat.java J MainPangkat.java 1 J Sum.java J Showroom.java J Mahasiswa.java ...P6 J DaftarMahasiswaBerprestasi.java J Main.java X J Main
P6 > P6 > J Main.java > Main > main(String[])
5 public class Main {
6     public static void main(String[] args) {
7
8         System.out.println(x:"Data Mahasiswa sebelum sorting:");
9         list.tampil();
10
11        System.out.println(x:"Data Mahasiswa setelah sorting desc berdasarkan ipk:");
12        list.bubbleSort();
13        list.tampil();
14
15        System.out.println(x:"Data Mahasiswa setelah sorting asc berdasarkan ipk:");
16        list.selectionSort();
17        list.tampil();
18
19        System.out.println(x:"Data Mahasiswa setelah sorting asc berdasarkan ipk:");
20        list.insertionSort();
21        list.tampil();
22    }
23 }
24 }
```

```
PS D:\college\semester 2\Algoritma dan Struktur Data 1G_01> d.; cd 'd:\college\semester 2\Algoritma dan Struktur Data 1G_01'; & 'C:\Program Files\Java\jdk-20\bin\java.exe' '-XX:+5
howCodeDetailsInExceptionMessages' '-cp' 'D:\college\semester 2\Algoritma dan Struktur Dat
a 1G_01\bin' 'P6.Main'
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 = Dampu
Tahun Masuk = 2018
Umur = 19
IPK = 3.5
-----
Nama = Abdul
Tahun Masuk = 2017
Umur = 23
IPK = 2.0
-----
Nama = Ummi
Tahun Masuk = 2019
Umur = 21
IPK = 3.75
-----
Data Mahasiswa setelah sorting desc berdasarkan ipk:
Nama = Abdul
Tahun Masuk = 2017
Umur = 23
IPK = 2.0
-----
Nama = Nusa
Tahun Masuk = 2017
Umur = 25
IPK = 3.0
-----
```



NAMA : Adani Salsab
NIM : 2341720123
KELAS : 1-G
MATERI : Sorting (Jobsheet 5)

```
IPK = 3.5
-----
Nama = Umi
Tahun Masuk = 2019
Umur = 21
IPK = 3.75
-----
Nama = Rara
Tahun Masuk = 2012
Umur = 19
IPK = 4.0
-----
Data Mahasiswa setelah sorting asc berdasarkan ipk
Nama = Abdul
Tahun Masuk = 2017
Umur = 23
IPK = 2.0
-----
Nama = Nurca
Tahun Masuk = 2017
Umur = 25
IPK = 3.0
-----
Nama = Dimpu
Tahun Masuk = 2018
Umur = 19
IPK = 3.5
-----
Nama = Umi
Tahun Masuk = 2019
Umur = 21
IPK = 3.75
-----
Nama = Rara
Tahun Masuk = 2012
Umur = 19
IPK = 4.0
-----
PS D:\college\semester 2\Algoritma dan Struktur Data_16_01>
```

Pertanyaan!

1. Ubahlah fungsi pada InsertionSort sehingga fungsi ini dapat melaksanakan proses sorting dengan cara descending.

Jawab :

```
src > P6 > J DaftarMahasiswaBerprestasi.java > DaftarMahasiswaBerprestasi > InsertionSort()
3 public class DaftarMahasiswaBerprestasi {
58 // method insertionSort()
59
60 void insertionSort() {
61     for (int i = 1; i < listMhs.length; i++) {
62         Mahasiswa temp = listMhs[i];
63         int j = i;
64         while (j > 0 && listMhs[j - 1].ipk < temp.ipk) {
65             listMhs[j] = listMhs[j - 1];
66             j--;
67         }
68         listMhs[j] = temp;
69     }
70 }
71
72 }
73 }
```

```
-----
Data Mahasiswa setelah sorting asc berdasarkan ipk
Nama = Rara
Tahun Masuk = 2012
Umur = 19
IPK = 4.0
-----
Nama = Umi
Tahun Masuk = 2019
Umur = 21
IPK = 3.75
-----
Nama = Dimpu
Tahun Masuk = 2018
Umur = 19
IPK = 3.5
-----
Nama = Nurca
Tahun Masuk = 2017
Umur = 25
IPK = 3.0
-----
Nama = Abdul
Tahun Masuk = 2017
Umur = 23
IPK = 2.0
-----
PS D:\college\semester 2\Algoritma dan Struktur Data_16_01>
```



NAMA : Adani Salsab
NIM : 2341720123
KELAS : 1-G
MATERI : Sorting (Jobsheet 5)

Latihan Praktikum

```
src > P6 > LatihanPraktikum > J HotelService.java > ts HotelService > selectionSortHotel[]
1 package P6.LatihanPraktikum;
2
3 public class HotelService {
4     void bubbleSort(Hotel[] hotels) {
5         int n = hotels.length;
6         for (int i = 0; i < n - 1; i++) {
7             for (int j = 0; j < n - i - 1; j++) {
8                 if (hotels[j].getHarga() > hotels[j + 1].getHarga()) {
9                     // Swap hotels
10                    Hotel temp = hotels[j];
11                    hotels[j] = hotels[j + 1];
12                    hotels[j + 1] = temp;
13                }
14            }
15        }
16    }
17
18    void selectionSort(Hotel[] hotels) {
19        int n = hotels.length;
20        for (int i = 0; i < n - 1; i++) {
21            int maxIdx = i;
22            for (int j = i + 1; j < n; j++) {
23                if (hotels[j].getBintang() > hotels[maxIdx].getBintang()) {
24                    maxIdx = j;
25                }
26            }
27            // Swap hotels
28            Hotel temp = hotels[maxIdx];
29            hotels[maxIdx] = hotels[i];
30            hotels[i] = temp;
31        }
32    }
33
34    void tambah(Hotel H, Hotel[] hotels) {
35        for (int i = 0; i < hotels.length; i++) {
36            if (hotels[i] == null) {
37                hotels[i] = H;
38                System.out.println("Hotel berhasil ditambahkan.");
39                return;
40            }
41        }
42        System.out.println("Kamar penuh, tidak bisa menambahkan hotel baru.");
43    }
44
45    void tampilAll(Hotel[] hotels) {
46        System.out.println("Daftar Hotel:");
47        for (Hotel hotel : hotels) {
48            if (hotel != null) {
49                System.out.println("Nama : " + hotel.getNama() + ", Harga: Rp." + hotel.getHarga() + ", Bintang : " + hotel.getBintang());
50            }
51        }
52    }
53 }
54 }
```

```
Selection View Go ... Algorithm dan Struktur Data IG 01
src > P6 > LatihanPraktikum > J MainHotel.java > ts MainHotel > main(String[])
1 // MainHotel.java
2 package P6.LatihanPraktikum;
3
4 public class MainHotel {
5     Run | Debug
6     public static void main(String[] args) {
7
8         Hotel[] hotels = new Hotel[5];
9         hotels[0] = new Hotel(nama: "Hotel Pramudya", harga: 300000, (byte) 4);
10        hotels[1] = new Hotel(nama: "Hotel Flameboy", harga: 800000, (byte) 3);
11        hotels[2] = new Hotel(nama: "Hotel Krisatena", harga: 320000, (byte) 5);
12        hotels[3] = new Hotel(nama: "Hotel Zuko", harga: 450000, (byte) 2);
13        hotels[4] = new Hotel(nama: "Hotel Sugoi", harga: 900000, (byte) 4);
14
15        // bubble Sort
16        HotelService hotelService = new HotelService();
17        hotelService.bubbleSort(hotels);
18        System.out.println("-----");
19        System.out.println("Data Hotel Urut Berdasarkan Harga (Terendah ke Tertinggi):");
20        printHotels(hotels);
21        System.out.println("-----");
22
23        // selection Sort
24        hotelService.selectionSort(hotels);
25        System.out.println("-----");
26        System.out.println("Data Hotel Urut Berdasarkan Rating Bintang (Tertinggi ke Terendah):");
27        printHotels(hotels);
28        System.out.println("-----");
29
30        static void printHotels(Hotel[] hotels) {
31            for (Hotel hotel : hotels) {
32                System.out.println(hotel.getNama() + " - Harga: Rp." + hotel.getHarga() + " - Rating: " + hotel.getBintang() + " Bintang ");
33            }
34        }
35    }
}
```




NAMA : Adani Salsab
NIM : 2341720123
KELAS : 1-G
MATERI : Sorting (Jobsheet 5)

```
PROBLEMS 15 OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS D:\college\semester 2\Algoritma_dan_Struktur_Data_1G_01> d;; cd 'd:\college\semester 2
\Algoritma_dan_Struktur_Data_1G_01'; & 'C:\Program Files\Java\jdk-20\bin\java.exe' '-XX:+S
howCodeDetailsInExceptionMessages' '-cp' 'D:\college\semester 2\Algoritma_dan_Struktur_Dat
a_1G_01\bin' 'P6.LatihanPraktikum.MainHotel'
-----
Data Hotel Urut Berdasarkan Harga (Terendah ke Tertinggi):
Hotel Pramudya - Harga: Rp.300000 - Rating: 4 Bintang
Hotel Krisatena - Harga: Rp.320000 - Rating: 5 Bintang
Hotel Zuko - Harga: Rp.450000 - Rating: 2 Bintang
Hotel Flamboyan - Harga: Rp.800000 - Rating: 3 Bintang
Hotel Sugoi - Harga: Rp.900000 - Rating: 4 Bintang
-----
Data Hotel Urut Berdasarkan Rating Bintang (Tertinggi ke Terendah):
Hotel Krisatena - Harga: Rp.320000 - Rating: 5 Bintang
Hotel Pramudya - Harga: Rp.300000 - Rating: 4 Bintang
Hotel Sugoi - Harga: Rp.900000 - Rating: 4 Bintang
Hotel Flamboyan - Harga: Rp.800000 - Rating: 3 Bintang
Hotel Zuko - Harga: Rp.450000 - Rating: 2 Bintang
-----
PS D:\college\semester 2\Algoritma_dan_Struktur_Data_1G_01> 
```