

Percobaan 1: Mengurutkan Data Mahasiswa Berdasarkan IPK Menggunakan Bubble Sort

```
package P5;

public class Mahasiswa {
    String nama;
    int thnMasuk;
    int umur;
    double ipk;

Mahasiswa(String n, int t, int u, double i) {
    nama = n;
    thnMasuk = t;
    umur = u;
    ipk = i;

void tampil() {
    System.out.println("Nama = " + nama);
    System.out.println("Tahun Masuk = " + thnMasuk);
    System.out.println("Umur = " + umur);
    System.out.println("IPK = " + ipk);
}

system.out.println("IPK = " + ipk);
}

and int the double i) {
    int umur;
    int undule i) {
        System.out.println("Nama = " + nama);
        System.out.println("Tahun Masuk = " + thnMasuk);
        System.out.println("IPK = " + ipk);
}
```

```
public class Main {
    public static void main(String[] args) {
        DaftarMahasiswaBerprestasi list = new DaftarMahasiswaBerprestasi();
        Mahasiswa m1 = new Mahasiswa("Nusa", 2017, 5, 3);
        Mahasiswa m2 = new Mahasiswa("Rara", 2012, 19, 4);
        Mahasiswa m3 = new Mahasiswa("Dompu", 2018, 19, 3.5);
        Mahasiswa m3 = new Mahasiswa("Dompu", 2018, 19, 3.5);
        Mahasiswa m5 = new Mahasiswa("Ummi", 2019, 21, 3.75);

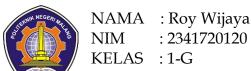
        list.tambah(m1);
        list.tambah(m2);
        list.tambah(m3);
        list.tambah(m3);
        list.tambah(m3);
        list.tambah(m5);

        System.out.println("Data Mahasiswa sebelum sorting=");
        list.tampil();
        System.out.println();
        System.out.println("Data Mahasiswa setelah disorting desc berdasarkan ipk ");
        list.bubbleSort();
        list.tampil();
    }
}
```



KELAS : 1-G

```
Data Mahasiswa sebelum sorting=
Nama = Nusa
Tahun Masuk = 2017
Umur = 5
IPK = 3.0
Nama = Rara
Tahun Masuk = 2012
Umur = 19
IPK = 4.0
Nama = Dompu
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
```



Pertanyaan

Terdapat di method apakah proses bubble sort?
 Jawab: proses bubble sort terdapat pada method void bubbleSort();

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

```
if(listMhs[j].ipk > listMhs[j-1].ipk){
    //di bawah ini proses swap atau penukaran
    Mahasiswa tmp = listMhs[j];
    listMhs[j] = listMhs[j-1];
    listMhs[j-1] = tmp;
}
```

Untuk apakah proses tersebut?

Jawab:

Proses tersebut bertujuan untuk menukar posisi dua objek "Mahasiswa" berdasarkan IPK. Jika IPK mahasiswa di posisi j lebih tinggi daripada IPK mahasiswa di posisi j-1, kedua objek tersebut ditukar untuk memastikan bahwa array terurut dari IPK tertinggi ke terendah.

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

a. Apakah perbedaan antara kegunaan perulangan i dan perulangan j?
 Jawab:

Perulangan i digunakan untuk menentukan berapa kali proses pengurutan harus dilakukan pada seluruh array,. Sedangkan, perulangan j digunakan untuk melakukan pembandingan dan penukaran antara elemen-elemen yang berdekatan dalam array.

b. Mengapa syarat dari perulangan i adalah ilistMhs.length-1?Jawab:

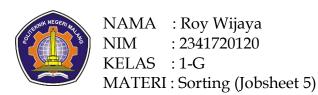
Jika I < listMhs.length tidak dikurangi 1 maka pada iterasi terakhir perulangan dalam tidak akan terpenuhi karena syarat perulangan dalam adalah j < listMhs.length – i.

c. Mengapa syarat dari perulangan j adalah j < listMhs.length-i? Jawab:

Jika listMhs.length tidak dikurangi i maka program akan memeriksa atribut/ipk yang telah diurutkan sehingga code program tidak efisien.

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

Jika banyak data di dalam listMhs adalah 50, maka perulangan i akan berlangsung sebanyak 49 kali. Karena syarat dari perulangan i adalah i < listMhs.length - 1, yang berarti i akan berjalan dari 0 hingga 48 (untuk listMhs.length sama dengan 50, 50 - 1 = 49)



Percobaan 2: Mengurutkan Data Mahasiswa Berdasarkan IPK Menggunakan Selection Sort

```
Data Mahasiswa sebelum sorting=
Nama = Nusa
Tahun Masuk = 2017
Umur = 5
IPK = 3.0
Nama = Rara
Tahun Masuk = 2012
Umur = 19
IPK = 4.0
Nama = Dompu
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 asc berdasarkan ipk
Nama = Abdul
Tahun Masuk = 2017
Umur = 23
IPK = 2.0
Nama = Nusa
Tahun Masuk = 2017
Umur = 5
IPK = 3.0
Nama = Dompu
Tahun Masuk = 2018
Umur = 19
IPK = 3.5
Nama = Ummi
Tahun Masuk = 2019
Umur = 21
IPK = 3.75
Nama = Rara
Tahun Masuk = 2012
Umur = 19
IPK = 4.0
```



KELAS: 1-G

```
package P5;
   public class DaftarMahasiswaBerprestasi {
      Mahasiswa listMhs[] = new Mahasiswa[5];
       int idx;
       void tambah(Mahasiswa m) {
          if (idx < listMhs.length) {</pre>
               listMhs[idx] = m;
               idx++;
          } else {
               System.out.println("Data sudah penuh!!");
       void tampil() {
          for (Mahasiswa m : listMhs) {
               m.tampil();
               System.out.println("----");
       void bubbleSort() {
           for (int i = 0; i < listMhs.length - 1; i++) {</pre>
               for (int j = 1; j < listMhs.length - i; <math>j++) {
                   if (listMhs[j].ipk > listMhs[j - 1].ipk) {
                      Mahasiswa tmp = listMhs[j];
                       listMhs[j] = listMhs[j - 1];
                       listMhs[j - 1] = tmp;
       void selectionSort() {
         for (int i = 0; i < listMhs.length - 1; i++) {
               int idxMin = i;
               for (int j = i + 1; j < listMhs.length; j++) {
                   if (listMhs[j].ipk < listMhs[idxMin].ipk) {</pre>
                       idxMin = j;
               Mahasiswa tmp = listMhs[idxMin];
               listMhs[idxMin] = listMhs[i];
               listMhs[i] = tmp;
```



KELAS : 1-G

```
package P5;

public class Main {
    public static void main(String[] args) {
        DaftarMahasiswaBerprestasi list = new DaftarMahasiswaBerprestasi();
        Mahasiswa m1 = new Mahasiswa("Nusa", 2017, 5, 3);
        Mahasiswa m2 = new Mahasiswa("Rara", 2012, 19, 4);
        Mahasiswa m3 = new Mahasiswa("Dompu", 2018, 19, 3.5);
        Mahasiswa m4 = new Mahasiswa("Dompu", 2018, 19, 3.5);
        Mahasiswa m5 = new Mahasiswa("Ummi", 2019, 21, 3.75);

list.tambah(m1);
        list.tambah(m2);
        list.tambah(m3);
        list.tambah(m4);
        list.tambah(m4);
        list.tambah(m5);

        System.out.println("Data Mahasiswa sebelum sorting=");
        list.tampil();

        System.out.println();
        System.out.println();
        System.out.println();
        System.out.println();
        System.out.println();
        System.out.println();
        System.out.println();
        System.out.println();
        System.out.println();
        list.tampil();

list.tampil();
```



Pertanyaan

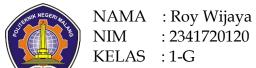
Di dalam method selection sort, terdapat baris program seperti di bawah ini:

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

Untuk apakah proses tersebut, jelaskan!

Jawab:

Pada deklarasi dan inisialisasi variabel idxMin = i digunakan untuk menentukan posisi nilai awal minimal. Kemudian terdapat perulangan menggunakan for yang di dalamnya terdapat pengecekan ipk, apakah ipk mahasiswa dengan indeks ke- j kurang dari ipk mahasiswa indeks ke- idxmin dimana idxmin telah ditentukan nilainya yaitu i, lalu jika pemilihan tersebut memenuhi maha idxMin diinisialisasi dengan j.



KELAS: 1-G
MATERI: Sorting (Jobsheet 5)



Percobaan 3: Mengurutkan Data Mahasiswa Berdasarkan IPK Menggunakan Insertion Sort

```
.
 1 package <u>P</u>5;
              Mahasiswa listMhs[] = new Mahasiswa[5];
               int idx;
                   if (idx < listMhs.length) {
    listMhs[idx] = m;</pre>
                              System.out.println("Data sudah penuh!!");
              void tampil() {
   for (Mahasiswa m : listMhs) {
                              System.out.println("----");
               void bubbleSort() {
   for (int i = 0; i < listMhs.length - 1; i++) {
      for (int j = 1; j < listMhs.length - i; j++) {
        if (listMhs[j].ipk > listMhs[j - 1].ipk) {
            Mahasiswa tmp = listMhs[j];
            listMhs[j] = listMhs[j - 1];
            listMhs[j - 1] = tmp;
      }
}
              void selectionSort() {
   for (int i = 0; i < listMhs.length - 1; i++) {</pre>
                          for (int j = i + 1; j < listMhs.length; j++) {
    if (listMhs[j].ipk < listMhs[idxMin].ipk) {
        idxMin = j;</pre>
                            ,
Mahasiswa tmp = listMhs[idxMin];
listMhs[idxMin] = listMhs[i];
listMhs[i] = tmp;
              void insertionSort() {
   for (int i = 1; i < listMhs.length; i++) {
      Mahasiswa temp = listMhs[i];</pre>
                         int j = i;

while (j > 0 && listMhs[j - 1].ipk > temp.ipk) {

    listMhs[j] = listMhs[j - 1];

    i - .
                               listMhs[j] = temp;
```

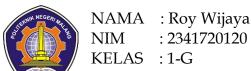


KELAS : 1-G

```
1 package <u>P</u>5;
    public class Main {
         public static void main(String[] args) {
            DaftarMahasiswaBerprestasi list = new DaftarMahasiswaBerprestasi();
            Mahasiswa m1 = new Mahasiswa("Nusa", 2017, 5, 3);
           Mahasiswa m2 = new Mahasiswa("Rara", 2012, 19, 4);

Mahasiswa m3 = new Mahasiswa("Dompu", 2018, 19, 3.5);

Mahasiswa m4 = new Mahasiswa("Abdul", 2017, 23, 2);
           Mahasiswa m5 = new Mahasiswa("Ummi", 2019, 21, 3.75);
          list.tambah(m1);
list.tambah(m2);
list.tambah(m3);
              list.tambah(m4);
             list.tambah(m5);
              System.out.println("Data Mahasiswa sebelum sorting=");
              list.tampil();
              System.out.println();
              System.out.println("Data mahasiswa setelah sorting asc berdasarkan ipk");
              list.insertionSort();
              list.tampil();
```



```
Data mahasiswa setelah sorting asc berdasarkan ipk
Nama = Abdul
Tahun Masuk = 2017
Umur = 23
IPK = 2.0
Nama = Nusa
Tahun Masuk = 2017
Umur = 5
IPK = 3.0
Nama = Dompu
Tahun Masuk = 2018
Umur = 19
IPK = 3.5
Nama = Ummi
Tahun Masuk = 2019
Umur = 21
IPK = 3.75
Nama = Rara
Tahun Masuk = 2012
Umur = 19
IPK = 4.0
```

```
Data Mahasiswa sebelum sorting=
Nama = Nusa
Tahun Masuk = 2017
Umur = 5
IPK = 3.0
Nama = Rara
Tahun Masuk = 2012
Umur = 19
IPK = 4.0
Nama = Dompu
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
```

Pertanyaan

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

Jawab:

```
System.out.println();
System.out.println();

System.out.println(x:"Data mahasiswa setelah sorting asc berdasarkan ipk");
list.insertionSort();
list.tampil();
```



KELAS : 1-G

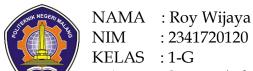
```
Data mahasiswa setelah sorting asc berdasarkan ipk
Nama = Rara
Tahun Masuk = 2012
Umur = 19
IPK = 4.0
Nama = Ummi
Tahun Masuk = 2019
Umur = 21
IPK = 3.75
Nama = Dompu
Tahun Masuk = 2018
Umur = 19
IPK = 3.5
Nama = Nusa
Tahun Masuk = 2017
Umur = 25
IPK = 3.0
Nama = Abdul
Tahun Masuk = 2017
Umur = 23
IPK = 2.0
```



Latihan Praktikum

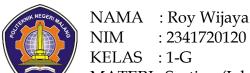
Main.java

```
package <u>P</u>6;
    public class MainHotel {
        public static void main(String[] args) {
              HotelService ht = new HotelService();
             Hotel h1 = new Hotel("Surya Hotel", "Kediri", 300000, Byte.valueOf("3"));
Hotel h2 = new Hotel("Victory Hotel", "Batu", 200000, Byte.valueOf("2"));
Hotel h3 = new Hotel("Nirwana Hotel", "Malang", 150000, Byte.valueOf("4"));
Hotel h4 = new Hotel("Ciputra Hotel", "Surabaya", 400000, Byte.valueOf("2"));
              Hotel h5 = new Hotel("Fave Hotel", "Denpasar", 500000, Byte.valueOf("5"));
              ht.tambah(h1);
              ht.tambah(h2);
              ht.tambah(h3);
              ht.tambah(h4);
              ht.tambah(h5);
              System.out.println("Sebelum di sorting");
              ht.tampilAll();
              System.out.println();
              System.out.println();
              ht.bubbleSortHarga();
              System.out.println("Setelah di sorting menggunakan bubble sort");
              ht.tampilAll();
              ht.selectionSortHarga();
              System.out.println("Setelah di sorting menggunakan selection sort");
              ht.tampilAll();
              ht.bubbleSortBintang();
              System.out.println("Setelah di sorting menggunakan bubble sort");
              ht.tampilAll();
              ht.selectionSortBintang();
              System.out.println("Setelah di sorting menggunakan ");
              ht.tampilAll();
```



Hotel.java

```
package P6;
  public class Hotel {
        String nama;
        String kota;
        int harga;
        Byte bintang;
        Hotel(String n, String k, int h, Byte b) {
            this.nama = n;
            this.kota = k;
11
12
            this.harga = h;
            this.bintang = b;
13
       }
   }
15
```



HotelService.java

```
public class HotelService {
   Hotel rooms[] = new Hotel[5];
   int idx;
          void tambah(Hotel H) {
  if (idx < rooms.length) {
    rooms[idx] = H;
    idx++;</pre>
        idx++;
} else {
    System.out.println("Data sudah penuh!!");
}
}
          void tampilAll() {
  for (int i = 0; i < rooms.length; i++) {
    System.out.println("Nama Hotel: " + rooms[i].nama);
    System.out.println("Kota: " + rooms[i].kota);
    System.out.println("Harga: " - rooms[i].harga);
    System.out.println("Bintang: " + rooms[i].bintang);
}</pre>
         void bubbleSortHarga() {
    for (int i = 0; i < rooms.length - 1; i++) {
        for (int j = 1; j < rooms.length - i; j++) {
            if (rooms[j], harga < rooms[j - 1].harga) {
                Hotel turp = rooms[j];
                 rooms[j] = rooms[j - 1];
                 rooms[j - 1] = tmp;
}</pre>
          void selectionSortHarga() {
   for (int i = 0; i < rooms.length - 1; i++) {</pre>
                                int idxMin = i;
for (int j = i + 1; j < rooms.length; j++) {
   if (rooms[j].harga < rooms[idxMin].harga) {
      idxMin = j;
   }</pre>
     void bubbleSortBintang() {
    for (int i = 0; i < rooms.length - 1; i++) {
        for (int j = 1; j < rooms.length - i; j++) {
            if (rooms[j].bintang < rooms[j - 1].bintang) {
                Hotel tmp = rooms[j];
                rooms[j] = rooms[j - 1];
                rooms[j - 1] = tmp;
            }
        }
    }
}</pre>
          void selectionSortBintang() {
   for (int i = 0; i < rooms.length - 1; i++) {</pre>
                            int idxMin = i;
for (int j = i + 1; j < rooms.length; j++) {
   if (rooms[j].bintang < rooms[idxMin].bintang) {
      idxMin = j;
}</pre>

}
Hotel tmp = rooms[idxMin];
rooms[idxMin] = rooms[i];
rooms[i] = tmp;
}
}
```



KELAS : 1-G

MATERI : Sorting (Jobsheet 5)

Sebelum di sorting Nama Hotel: Surya Hotel

Kota: Kediri Harga: 300000 Bintang: 3

Nama Hotel: Victory Hotel

Kota: Batu Harga: 200000 Bintang: 2

Nama Hotel: Nirwana Hotel

Kota: Malang Harga: 150000 Bintang: 4

Nama Hotel: Ciputra Hotel

Kota: Surabaya Harga: 400000 Bintang: 2

Nama Hotel: Fave Hotel

Kota: Denpasar Harga: 500000 Bintang: 5

Setelah di sorting menggunakan bubble sort berdasarkan harga

Nama Hotel: Nirwana Hotel

Kota: Malang Harga: 150000 Bintang: 4

Nama Hotel: Victory Hotel

Kota: Batu Harga: 200000 Bintang: 2

Nama Hotel: Surya Hotel

Kota: Kediri Harga: 300000 Bintang: 3

Nama Hotel: Ciputra Hotel

Kota: Surabaya Harga: 400000 Bintang: 2

Nama Hotel: Fave Hotel

Kota: Denpasar Harga: 500000 Bintang: 5



NAMA: Roy Wijaya : 2341720120 NIM

KELAS: 1-G

MATERI : Sorting (Jobsheet 5)

Setelah di sorting menggunakan selection sort berdasarkan harga

Nama Hotel: Nirwana Hotel

Kota: Malang Harga: 150000 Bintang: 4

Nama Hotel: Victory Hotel

Kota: Batu Harga: 200000 Bintang: 2

Nama Hotel: Surya Hotel

Kota: Kediri Harga: 300000 Bintang: 3

Nama Hotel: Ciputra Hotel

Kota: Surabaya Harga: 400000 Bintang: 2

Nama Hotel: Fave Hotel

Kota: Denpasar Harga: 500000 Bintang: 5

Setelah di sorting menggunakan bubble sort berdasarkan bintang

Nama Hotel: Victory Hotel

Kota: Batu Harga: 200000 Bintang: 2 Nama Hotel: Ciputra Hotel

Kota: Surabaya Harga: 400000 Bintang: 2

Nama Hotel: Surya Hotel

Kota: Kediri Harga: 300000 Bintang: 3

Nama Hotel: Nirwana Hotel

Kota: Malang Harga: 150000 Bintang: 4

Nama Hotel: Fave Hotel Kota: Denpasar

Harga: 500000 Bintang: 5



KELAS : 1-G

MATERI : Sorting (Jobsheet 5)

Setelah di sorting menggunakan berdasarkan bintang

Nama Hotel: Victory Hotel

Kota: Batu Harga: 200000 Bintang: 2

Nama Hotel: Ciputra Hotel

Kota: Surabaya Harga: 400000 Bintang: 2

Nama Hotel: Surya Hotel

Kota: Kediri Harga: 300000 Bintang: 3

Nama Hotel: Nirwana Hotel

Kota: Malang Harga: 150000 Bintang: 4

Nama Hotel: Fave Hotel

Kota: Denpasar Harga: 500000 Bintang: 5

LINK GITHUB:

 $https://github.com/RoyW12/AlgoritmaStrukturData_1G_28/tree/main/src/P6$