

**LAPORAN HASIL PRAKTIKUM ALGORITMA DAN STRUKTUR
DASAR**

Jobsheet 5



Adi Luhung

244107020088

Kelas 1E

Program Studi Teknik Informatika

Jurusan Teknologi Informasi

Politeknik Negeri Malang

2025

1. Pratikum

1.1 Percobaan 1

a. Sorting-Bubble Sort

1. Membuat kode program untuk sorting sekumpulan bilangan, dengan menggunakan metode bubble sort

```
package Praktikum5;
public class Sorting01 {
    int [] data;
    int jumData;
    Sorting01 (int Data[], int jmlDat) {
        jumData = jmlDat;
        data = new int [jmlDat];
        for (int i = 0; i < jumData; i++) {
            data[i] = Data[i];
        }
    }
    void bubblesort(){
        int temp = 0;
        for (int i = 0; i < jumData-1; i++) {
            for (int j = 1; j < jumData-i; j++) {
                if (data[j-1] > data[j]){
                    temp = data[j];
                    data[j] = data [j-1];
                    data[j-1] = temp;
                }
            }
        }
    }
    void tampil(){
        for (int i = 0; i < jumData; i++) {
            System.out.print(data[i] + " ");
        }
        System.out.println();
    }
}
```

```
package Praktikum5;
public class SortingMain01 {
    public static void main(String[] args) {
        int a [] = {20, 10, 2, 7, 12};
        Sorting01 dataurut1 = new Sorting01 (a, a.length);
        System.out.println("Data awal 1");
        dataurut1.tampil();
        dataurut1.bubblesort();
        System.out.println("Data sudah diurutkan dengan BUBBLE SORT (ASC)");
        dataurut1.tampil();
    }
}
```

2. Hasil kode program

```
Data awal 1
20 10 2 7 12
Data sudah diurutkan dengan BUBBLE SORT (ASC)
2 7 10 12 20
```

b. Sorting-Selection Sort

1. Memodifikasi kode program agar dapat melakukan sorting menggunakan metode selection sort

```
package Praktikum5;
public class Sorting01 {
    int [] data;
    int jumData;
    Sorting01 (int Data[], int jmlDat) {
        jumData = jmlDat;
        data = new int [jmlDat];
        for (int i = 0; i < jumData; i++) {
            data[i] = Data[i];
        }
    }
    void bubbleSort (){
        int temp = 0;
        for (int i = 0; i < jumData-1; i++) {
            for ( int j = 1; j < jumData-i; j++) {
                if (data[j-1] > data[j]){
                    temp = data[j];
                    data[j] = data [j-1];
                    data[j-1] = temp;
                }
            }
        }
    }
    void SelectionSort (){
        for (int i = 0; i < jumData-1; i++) {
            int min = i;
            for (int j = i+1; j < jumData; j++) {
                if (data [j] < data [min]){
                    min = j;
                }
            }
            int temp = data [i];
            data [i] = data [min];
            data [min] = temp;
        }
    }
    void tampil(){
        for (int i = 0; i < jumData; i++) {
            System.out.print(data[i] + " ");
        }
        System.out.println();
    }
}
```

```

package Praktikum5;
public class SortingMain01 {
    public static void main(String[] args) {
        int a [] = {20, 10, 2, 7, 12};
        Sorting01 dataurut1 = new Sorting01 (a, a.length);
        System.out.println("Data awal 1");
        dataurut1.tampil();
        dataurut1.bubbleSort();
        System.out.println("Data sudah diurutkan dengan BUBBLE SORT (ASC)");
        dataurut1.tampil();

        System.out.println(" ");
        int b [] = {30, 20, 2, 8, 14};
        Sorting01 dataurut2 = new Sorting01 (b, b.length);
        System.out.println("Data awal 2");
        dataurut2.tampil();
        dataurut2.bubbleSort();
        System.out.println("Data sudah diurutkan dengan SELECTION SORT (ASC)");
        dataurut2.tampil();
    }
}

```

2. Hasil kode program

```

Data awal 1
20 10 2 7 12
Data sudah diurutkan dengan BUBBLE SORT (ASC)
2 7 10 12 20

Data awal 2
30 20 2 8 14
Data sudah diurutkan dengan SELECTION SORT (ASC)
2 8 14 20 30

```

c. Sorting-Inserttion Sort

1. Memodifikasi kode program agar dapat melakukan sorting menggunakan metode insertion sort

```
package Praktikum5;
public class Sorting01 {
    int [] data;
    int jumData;
    Sorting01 (int Data[], int jmlDat) {
        jumData = jmlDat;
        data = new int [jmlDat];
        for (int i = 0; i < jumData; i++) {
            data[i] = Data[i];
        }
    }
    void bubbleSort (){
        int temp = 0;
        for (int i = 0; i < jumData-1; i++) {
            for (int j = 1; j < jumData-i; j++) {
                if (data[j-1] > data[j]){
                    temp = data[j];
                    data[j] = data [j-1];
                    data[j-1] = temp;
                }
            }
        }
    }
    void SelectionSort (){
        for (int i = 0; i < jumData-1; i++) {
            int min = i;
            for (int j = i+1; j < jumData; j++) {
                if (data [j] < data [min]){
                    min = j;
                }
            }
            int temp = data [i];
            data [i] = data [min];
            data [min] = temp;
        }
    }
    void InsertionSort (){
        for (int i = 1; i <= data.length-1; i++) {
            int temp = data[i];
            int j = i-1;
            while (j >= 0 && data[j] > temp) {
                data [j+1] = data [j];
                j--;
            }
            data [j+1] = temp;
        }
    }
    void tampil(){
        for (int i = 0; i < jumData; i++) {
            System.out.print(data[i] + " ");
        }
        System.out.println();
    }
}
```

```

package Praktikum5;
public class SortingMain01 {
    public static void main(String[] args) {
        int a [] = {20, 10, 2, 7, 12};
        Sorting01 dataaurut1 = new Sorting01 (a, a.length);
        System.out.println("Data awal 1");
        dataaurut1.tampil();
        dataaurut1.bubbleSort();
        System.out.println("Data sudah diurutkan dengan BUBBLE SORT (ASC)");
        dataaurut1.tampil();

        System.out.println(" ");
        int b [] = {30, 20, 2, 8, 14};
        Sorting01 dataaurut2 = new Sorting01 (b, b.length);
        System.out.println("Data awal 2");
        dataaurut2.tampil();
        dataaurut2.bubbleSort();
        System.out.println("Data sudah diurutkan dengan SELECTION SORT (ASC)");
        dataaurut2.tampil();

        System.out.println(" ");
        int c [] = {40, 10, 4, 9, 3};
        Sorting01 dataaurut3 = new Sorting01 (c, c.length);
        System.out.println("Data awal 3");
        dataaurut3.tampil();
        dataaurut3.InsertionSort();
        System.out.println("Data sudah diurutkan dengan INSERTION SORT (ASC)");
        dataaurut3.tampil();
    }
}

```

2. Hasil kode program

```

Data awal 1
20 10 2 7 12
Data sudah diurutkan dengan BUBBLE SORT (ASC)
2 7 10 12 20

Data awal 2
30 20 2 8 14
Data sudah diurutkan dengan SELECTION SORT (ASC)
2 8 14 20 30

Data awal 3
40 10 4 9 3
Data sudah diurutkan dengan INSERTION SORT (ASC)
3 4 9 10 40

```

1.1.1 Pertanyaan

1. Fungsi kode program tersebut adalah untuk menukar nilai yang terdapat dalam data [j-1] dan data [j], dengan menggunakan variabel temp sebagai tempat sementara untuk menyimpan nilai data [j].

2. Yang merupakan algoritma pencarian nilai minimum pada selection sort adalah

```
for (int j = i+1; j < jumData; j++) {  
    if (data [j] < data [min]){  
        min = j;  
    }  
}
```

3. Kode program tersebut berfungsi sebagai kondisi, yaitu program akan menjalankan suatu perintah selama j >= 0 && data[j] > temp.
4. Tujuan perintah tersebut adalah untuk menyalin nilai didalam data[j] ke data[j+1]. Sehingga nilai didalam variabel temp bisa dipindah ke data[j+1] setelah j--.

1.2 Percobaan 2 - Sorting Menggunakan Array of Object

1.2.1 Mengurutkan data mahasiswa berdasarkan IPK (Bubble Sort)

1. Membuat kode program

```
package Praktikum5;  
public class Mahasiswa01 {  
    String nim;  
    String nama;  
    String kelas;  
    double ipk;  
  
    Mahasiswa01() {  
    }  
    Mahasiswa01(String nm, String name, String kls, double ip) {  
        nim = nm;  
        nama = name;  
        ipk = ip;  
        kelas = kls;  
    }  
    void tampilInformasi() {  
        System.out.println("Nama: " + nama);  
        System.out.println("NIM: " + nim);  
        System.out.println("Kelas: " + kelas);  
        System.out.println("IPK: " + ipk);  
    }  
}
```

```

package Praktikum5;
public class MahasiswaBerprestasi01 {
    Mahasiswa01 [] listMhs = new Mahasiswa01 [5];
    int idx;

    void tambah (Mahasiswa01 m) {
        if (idx < listMhs.length) {
            listMhs[idx] = m;
            idx++;
        } else {
            System.out.println("Data sudah penuh");
        }
    }
    void tampil () {
        for (Mahasiswa01 m : listMhs) {
            m.tampilInformasi();
            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) {
                    Mahasiswa01 tmp = listMhs[j];
                    listMhs[j] = listMhs[j-1];
                    listMhs[j-1] = tmp;
                }
            }
        }
    }
}

```

```

package Praktikum5;
public class MahasiswaDemo01 {
    public static void main(String[] args) {
        MahasiswaBerprestasi01 list = new MahasiswaBerprestasi01();
        Mahasiswa01 m1 = new Mahasiswa01("123", "Zidan", "2A", 3.2);
        Mahasiswa01 m2 = new Mahasiswa01("124", "Ayu", "2A", 3.5);
        Mahasiswa01 m3 = new Mahasiswa01("125", "Sodi", "2A", 3.1);
        Mahasiswa01 m4 = new Mahasiswa01("126", "Sita", "2A", 3.9);
        Mahasiswa01 m5 = new Mahasiswa01("127", "Miki", "2A", 3.7);

        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("Data mahasiswa setelah sorting berdasarkan IPK (DESC): ");
        list.bubbleSort();
        list.tampil();
    }
}

```


2. Hasil kode program

Data mahasiswa sebelum sorting:

Nama: Zidan

NIM: 123

Kelas: 2A

IPK: 3.2

Nama: Ayu

NIM: 124

Kelas: 2A

IPK: 3.5

Nama: Sodi

NIM: 125

Kelas: 2A

IPK: 3.1

Nama: Sita

NIM: 126

Kelas: 2A

IPK: 3.9

Nama: Miki

NIM: 127

Kelas: 2A

IPK: 3.7

Data mahasiswa setelah sorting berdasarkan IPK (DESC):

Nama: Sita

NIM: 126

Kelas: 2A

IPK: 3.9

Nama: Miki

NIM: 127

Kelas: 2A

IPK: 3.7

Nama: Ayu

NIM: 124

Kelas: 2A

IPK: 3.5

Nama: Zidan

NIM: 123

Kelas: 2A

IPK: 3.2

Nama: Sodi

NIM: 125

Kelas: 2A

IPK: 3.1

1.2.1.1 Pertanyaan

1. Perulangan pada `bubbleSort()` :
 - a. Kondisi `i < listMhs.length-1` berfungsi sebagai *outer loop* yang menentukan jumlah iterasi untuk memastikan seluruh elemen sudah dalam urutan yang benar.
 - b. Kondisi `j < listMhs.length-i` berfungsi sebagai *inner loop* yang digunakan untuk membandingkan dan menukar elemen yang berdekatan. Dan elemen terbesar yang sudah dalam posisi yang benar tidak akan dibandingkan lagi.
 - c. Jika banyak data didalam `listMhs` adalah 50, maka perulangan `i` akan terjadi sebanyak 49 kali, karena berjalan dari 0 hingga 48. Maka Bubble Sort akan menempuh 49 tahap.
2. Memodifikasi kode program agar dapat menerima input dari keyboard

```
package Praktikum5;
public class Mahasiswa01 {
    String nim;
    String nama;
    String kelas;
    double ipk;

    Mahasiswa01() {
    }
    Mahasiswa01(String nm, String name, String kls, double ip) {
        nim = nm;
        nama = name;
        ipk = ip;
        kelas = kls;
    }
    void tampilInformasi() {
        System.out.println("Nama: " + nama);
        System.out.println("NIM: " + nim);
        System.out.println("Kelas: " + kelas);
        System.out.println("IPK: " + ipk);
    }
}
```

```

package Praktikum5;
public class MahasiswaBerprestasi01 {
    Mahasiswa01[] listMhs;
    int idx;

    MahasiswaBerprestasi01(int jumlahMahasiswa) {
        listMhs = new Mahasiswa01[jumlahMahasiswa];
        idx = 0;
    }
    void tambah(Mahasiswa01 m) {
        if (idx < listMhs.length) {
            listMhs[idx] = m;
            idx++;
        } else {
            System.out.println("Data sudah penuh");
        }
    }
    void tampil() {
        for (Mahasiswa01 m : listMhs) {
            if (m != null) {
                m.tampilInformasi();
                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) {
                    Mahasiswa01 tmp = listMhs[j];
                    listMhs[j] = listMhs[j - 1];
                    listMhs[j - 1] = tmp;
                }
            }
        }
    }
}

```

```

package Praktikum5;
import java.util.Scanner;
public class MahasiswaDemo01 {
    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);

        System.out.print("Masukkan jumlah mahasiswa: ");
        int jumlahMahasiswa = input.nextInt();
        input.nextLine();

        MahasiswaBerprestasi01 list = new MahasiswaBerprestasi01(jumlahMahasiswa);

        for (int i = 0; i < jumlahMahasiswa; i++) {
            System.out.println("Masukkan data mahasiswa ke-" + (i + 1) + ":");
            System.out.print("NIM      : ");
            String nim = input.nextLine();
            System.out.print("Nama      : ");
            String nama = input.nextLine();
            System.out.print("Kelas    : ");
            String kelas = input.nextLine();
            System.out.print("IPK      : ");
            double ipk = input.nextDouble();
            input.nextLine();

            Mahasiswa01 m = new Mahasiswa01(nim, nama, kelas, ipk);
            list.tambah(m);
        }

        System.out.println("\nData mahasiswa sebelum sorting: ");
        list.tampil();

        System.out.println("\nData mahasiswa setelah sorting berdasarkan IPK (DESC): ");
        list.bubbleSort();
        list.tampil();

        input.close();
    }
}

```

Hasil kode program

```
Masukkan jumlah mahasiswa: 5
Masukkan data mahasiswa ke-1:
NIM      : 123
Nama     : Zidan
Kelas   : 2A
IPK      : 3,2
```

```
-----
Masukkan data mahasiswa ke-2:
NIM      : 124
Nama     : Ayu
Kelas   : 2A
IPK      : 3,5
```

```
-----
Masukkan data mahasiswa ke-3:
NIM      : 125
Nama     : Sofi
Kelas   : 2A
IPK      : 3,1
```

```
-----
Masukkan data mahasiswa ke-4:
NIM      : 126
Nama     : Sita
Kelas   : 2A
IPK      : 3,9
```

```
-----
Masukkan data mahasiswa ke-5:
NIM      : 127
Nama     : Miki
Kelas   : 2A
IPK      : 3,7
-----
```

Data mahasiswa sebelum sorting:

```
Nama     : Zidan
NIM      : 123
Kelas   : 2A
IPK      : 3.2
```

```
-----
Nama     : Ayu
NIM      : 124
Kelas   : 2A
IPK      : 3.5
```

```
-----
Nama     : Sofi
NIM      : 125
Kelas   : 2A
IPK      : 3.1
```

```
-----
Nama     : Sita
NIM      : 126
Kelas   : 2A
IPK      : 3.9
```

```
-----
Nama     : Miki
NIM      : 127
Kelas   : 2A
IPK      : 3.7
-----
```

Data mahasiswa setelah sorting berdasarkan IPK (DESC):

```
Nama     : Sita
NIM      : 126
Kelas   : 2A
IPK      : 3.9
```

```
-----
Nama     : Miki
NIM      : 127
Kelas   : 2A
IPK      : 3.7
```

```
-----
Nama     : Ayu
NIM      : 124
Kelas   : 2A
IPK      : 3.5
```

```
-----
Nama     : Zidan
NIM      : 123
Kelas   : 2A
IPK      : 3.2
```

```
-----
Nama     : Sofi
NIM      : 125
Kelas   : 2A
IPK      : 3.1
-----
```

1.2.2 Mengurutkan data mahasiswa berdasarkan IPK (Selection Sort)

1. Memodifikasi kode program dengan menambahkan method selectionSort pada class MahasiswaBerprestasi01

```
package Praktikum5;
public class MahasiswaBerprestasi01 {
    Mahasiswa01[] listMhs;
    int idx;
    MahasiswaBerprestasi01(int jumlahMahasiswa) {
        listMhs = new Mahasiswa01[jumlahMahasiswa];
        idx = 0;
    }
    void tambah(Mahasiswa01 m) {
        if (idx < listMhs.length) {
            listMhs[idx] = m;
            idx++;
        } else {
            System.out.println("Data sudah penuh");
        }
    }
    void tampil() {
        for (Mahasiswa01 m : listMhs) {
            if (m != null) {
                m.tampilInformasi();
                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) {
                    Mahasiswa01 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) {
                    idxMin = j;
                }
            }
            Mahasiswa01 tmp = listMhs[idxMin];
            listMhs[idxMin] = listMhs[i];
            listMhs[i] = tmp;
        }
    }
}
```

```

package Praktikum5;
import java.util.Scanner;
public class MahasiswaDemo01 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        System.out.print("Masukkan jumlah mahasiswa: ");
        int jumlahMahasiswa = sc.nextInt();
        sc.nextLine();

        MahasiswaBerprestasi01 list = new MahasiswaBerprestasi01(jumlahMahasiswa);

        for (int i = 0; i < jumlahMahasiswa; i++) {
            System.out.println("Masukkan data mahasiswa ke-" + (i + 1) + ":");
            System.out.print("NIM      : ");
            String nim = sc.nextLine();
            System.out.print("Nama      : ");
            String nama = sc.nextLine();
            System.out.print("Kelas    : ");
            String kelas = sc.nextLine();
            System.out.print("IPK       : ");
            double ipk = sc.nextDouble();
            System.out.println("-----");
            sc.nextLine();

            Mahasiswa01 m = new Mahasiswa01(nim, nama, kelas, ipk);
            list.tambah(m);
        }

        System.out.println("\nData mahasiswa sebelum sorting: ");
        list.tampil();

        System.out.println("\nData mahasiswa setelah sorting berdasarkan IPK (DESC): ");
        list.bubbleSort();
        list.tampil();

        System.out.println("\nData yang sudah terurut menggunakan SELECTION SORT (ASD): ");
        list.selectionSort();
        list.tampil();
        sc.close();
    }
}

```

2. Hasil kode program

```
Masukkan jumlah mahasiswa: 5
Masukkan data mahasiswa ke-1:
NIM      : 123
Nama     : Ali
Kelas   : 2B
IPK      : 3,9
-----
Masukkan data mahasiswa ke-2:
NIM      : 124
Nama     : ila
Kelas   : 2b
IPK      : 125
-----
Masukkan data mahasiswa ke-3:
NIM      : 125
Nama     : agus
Kelas   : 2B
IPK      : 3,6
-----
Masukkan data mahasiswa ke-4:
NIM      : 126
Nama     : tika
Kelas   : 2B
IPK      : 3,3
-----
Masukkan data mahasiswa ke-5:
NIM      : 127
Nama     : udin
Kelas   : 2B
IPK      : 3,2
-----
```

```
Data yang sudah terurut menggunakan SELECTION SORT (ASD):
Nama     : udin
NIM      : 127
Kelas   : 2B
IPK      : 3.2
-----
Nama     : tika
NIM      : 126
Kelas   : 2B
IPK      : 3.3
-----
Nama     : agus
NIM      : 125
Kelas   : 2B
IPK      : 3.6
-----
Nama     : Ali
NIM      : 123
Kelas   : 2B
IPK      : 3.9
-----
Nama     : ila
NIM      : 124
Kelas   : 2b
IPK      : 125.0
-----
```

1.2.2.1 Pertanyaan

1. Kode program tersebut berfungsi untuk mencari nilai IPK terkecil yang terdapat pada array `listMhs`. Kemudian jika nilai minimum telah ditemukan maka indeks dari nilai minimum tersebut akan dipindah ke variabel `idxMin`.

1.2.3 Mengurutkan data mahasiswa berdasarkan IPK menggunakan Insertion Sort

1. Memodifikasi kode program dengan menambahkan method insertionSort pada class MahasiswaBerprestasi.

```
package Praktikum5;
public class MahasiswaBerprestasi01 {
    Mahasiswa01[] listMhs;
    int idx;
    MahasiswaBerprestasi01(int jumlahMahasiswa) {
        listMhs = new Mahasiswa01[jumlahMahasiswa];
        idx = 0;
    }
    void tambah(Mahasiswa01 m) {
        if (idx < listMhs.length) {
            listMhs[idx] = m;
            idx++;
        } else {
            System.out.println("Data sudah penuh");
        }
    }
    void tampil() {
        for (Mahasiswa01 m : listMhs) {
            if (m != null) {
                m.tampilInformasi();
                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) {
                    Mahasiswa01 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) {
                    idxMin = j;
                }
            }
            Mahasiswa01 tmp = listMhs[idxMin];
            listMhs[idxMin] = listMhs[i];
            listMhs[i] = tmp;
        }
    }
    void insertionSort() {
        for (int i = 1; i < listMhs.length; i++) {
            Mahasiswa01 temp = listMhs[i];
            int j = i;
            while (j > 0 && listMhs[j-1].ipk > temp.ipk) {
                listMhs[j] = listMhs[j-1];
                j--;
            }
            listMhs[j] = temp;
        }
    }
}
```

2. Hasil kode program

```
Masukkan jumlah mahasiswa: 5
Masukkan data mahasiswa ke-1:
NIM      : 111
Nama     : ayu
Kelas   : 2c
IPK      : 3,7
-----
Masukkan data mahasiswa ke-2:
NIM      : 222
Nama     : dika
Kelas   : 2c
IPK      : 3,0
-----
Masukkan data mahasiswa ke-3:
NIM      : 333
Nama     : ila
Kelas   : 2c
IPK      : 3,8
-----
Masukkan data mahasiswa ke-4:
NIM      : 444
Nama     : susi
Kelas   : 2c
IPK      : 3,1
-----
Masukkan data mahasiswa ke-5:
NIM      : 555
Nama     : yayuk
Kelas   : 2c
IPK      : 3,4
-----
```

```
Data yang sudah terurut menggunakan INSERTION SORT (ASC):
Nama     : dika
NIM      : 222
Kelas   : 2c
IPK      : 3.0
-----
Nama     : susi
NIM      : 444
Kelas   : 2c
IPK      : 3.1
-----
Nama     : yayuk
NIM      : 555
Kelas   : 2c
IPK      : 3.4
-----
Nama     : ayu
NIM      : 111
Kelas   : 2c
IPK      : 3.7
-----
Nama     : ila
NIM      : 333
Kelas   : 2c
IPK      : 3.8
-----
```

1.2.3.1 Pertanyaan

1. Memodifikasi method insertionSort agar melaksanakan proses sorting secara *descending*

```
void insertionSort() {
    for (int i = 1; i < listMhs.length; i++) {
        Mahasiswa01 temp = listMhs[i];
        int j = i;
        while (j > 0 && listMhs[j-1].ipk < temp.ipk) {
            listMhs[j] = listMhs[j-1];
            j--;
        }
        listMhs[j] = temp;
    }
}
```

2. Hasil kode program

```
Masukkan jumlah mahasiswa: 5
Masukkan data mahasiswa ke-1:
NIM      : 111
Nama     : ayu
Kelas   : 2c
IPK      : 3,7
-----
Masukkan data mahasiswa ke-2:
NIM      : 222
Nama     : dika
Kelas   : 2c
IPK      : 3,0
-----
Masukkan data mahasiswa ke-3:
NIM      : 333
Nama     : ila
Kelas   : 2c
IPK      : 3,8
-----
Masukkan data mahasiswa ke-4:
NIM      : 444
Nama     : susi
Kelas   : 2c
IPK      : 3,1
-----
Masukkan data mahasiswa ke-5:
NIM      : 555
Nama     : yayuk
Kelas   : 2c
IPK      : 3,4
-----
```

```
Data yang sudah terurut menggunakan INSERTION SORT (ASC):
Nama     : ila
NIM      : 333
Kelas   : 2c
IPK      : 3.8
-----
Nama     : ayu
NIM      : 111
Kelas   : 2c
IPK      : 3.7
-----
Nama     : yayuk
NIM      : 555
Kelas   : 2c
IPK      : 3.4
-----
Nama     : susi
NIM      : 444
Kelas   : 2c
IPK      : 3.1
-----
Nama     : dika
NIM      : 222
Kelas   : 2c
IPK      : 3.0
-----
```

2. Tugas

1. Membuat kode program untuk menyimpan data dosen serta menyortirnya, dengan cara *ascending* dan *descending*.

```
package Praktikum5;
public class DataDosen {
    Dosen[] listDosen;
    int idx;
    DataDosen (int n) {
        listDosen = new Dosen [n];
        idx = 0;
    }
    void tambah(Dosen dsn) {
        if (idx < listDosen.length) {
            listDosen[idx] = dsn;
            idx++;
        } else {
            System.out.println("Data sudah penuh");
        }
    }
    void tampil() {
        for (Dosen dsn : listDosen) {
            if (dsn != null) {
                dsn.tampil();
                System.out.println("-----");
            }
        }
    }
    void sortingASC() {
        for (int i = 0; i < listDosen.length - 1; i++) {
            for (int j = 1; j < listDosen.length - i; j++) {
                if (listDosen[j].usia < listDosen[j - 1].usia) {
                    Dosen tmp = listDosen[j];
                    listDosen[j] = listDosen[j - 1];
                    listDosen[j - 1] = tmp;
                }
            }
        }
    }
    void sortingDSC() {
        for (int i = 1; i < listDosen.length; i++) {
            Dosen temp = listDosen[i];
            int j = i;
            while (j > 0 && listDosen[j-1].usia < temp.usia) {
                listDosen[j] = listDosen[j-1];
                j--;
            }
            listDosen[j] = temp;
        }
    }
}
```

```

package Praktikum5;
import java.util.Scanner;
public class DosenMain {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        DataDosen list = new DataDosen(10);

        for (int i = 0; i < list.listDosen.length ; i++) {
            System.out.println("Masukkan data dosen ke-" + (i + 1) + ":");
            System.out.print("Kode          : ");
            String kode = sc.nextLine();
            System.out.print("Nama          : ");
            String nama = sc.nextLine();
            System.out.print("Jenis Kelamin : ");
            String jk = sc.nextLine();
            System.out.print("Usia          : ");
            int usia = sc.nextInt();
            sc.nextLine();
            System.out.println("-----");

            Boolean jenisKelamin = false;
            if (jk.equalsIgnoreCase("pria")) {
                jenisKelamin = true;
            }

            Dosen dsn = new Dosen(kode, nama, jenisKelamin, usia);
            list.tambah(dsn);
        }
        System.out.println("\nData dosen sebelum sorting: ");
        list.tampil();

        System.out.println("\nData dosen setelah sorting berdasarkan Usia (ASC): ");
        list.sortingASC();
        list.tampil();

        System.out.println("\nData dosen setelah sorting berdasarkan Usia (DSC):");
        list.sortingDSC();
        list.tampil();
        sc.close();
    }
}

```

```

package Praktikum5;
public class Dosen {
    String kode;
    String nama;
    Boolean jenisKelamin;
    int usia;

    Dosen () {
    }
    Dosen (String kd, String name, Boolean jk, int age) {
        kode = kd;
        nama = name;
        jenisKelamin = jk;
        usia = age;
    }
    void tampil() {
        String jkl = "";
        if (jenisKelamin == true) {
            jkl = "Pria";
        } else {
            jkl = "Wanita";
        }
        System.out.println("Kode           : " + kode);
        System.out.println("Nama           : " + nama);
        System.out.println("Jenis Kelamin  : " + jkl);
        System.out.println("Usia          : " + usia);
    }
}

```

2. Hasil kode program

Masukkan data dosen ke-1:

```

Kode       : 111
Nama        : Budi
Jenis Kelamin : Pria
Usia        : 65

```

Masukkan data dosen ke-2:

```

Kode       : 123
Nama        : Mas
Jenis Kelamin : Pria
Usia        : 34

```

Masukkan data dosen ke-3:

```

Kode       : 190
Nama        : Bunga
Jenis Kelamin : Wanita
Usia        : 27

```

Masukkan data dosen ke-4:

```

Kode       : 911
Nama        : Xeno
Jenis Kelamin : Pria
Usia        : 54

```

Masukkan data dosen ke-5:

```

Kode       : 121
Nama        : Shiva
Jenis Kelamin : Wanita
Usia        : 25

```

Masukkan data dosen ke-6:

```

Kode       : 292
Nama        : Nur
Jenis Kelamin : Wanita
Usia        : 49

```

Masukkan data dosen ke-7:

```

Kode       : 459
Nama        : Moss
Jenis Kelamin : Pria
Usia        : 20

```

Masukkan data dosen ke-8:

```

Kode       : 100
Nama        : Vusia
Jenis Kelamin : Wanita
Usia        : 57

```

Masukkan data dosen ke-9:

```

Kode       : 711
Nama        : Ursus
Jenis Kelamin : Pria
Usia        : 29

```

Masukkan data dosen ke-10:

```

Kode       : 001
Nama        : Jokowi
Jenis Kelamin : Pria
Usia        : 67

```

Data dosen sebelum sorting:

Kode : 111
Nama : Budi
Jenis Kelamin : Pria
Usia : 65

Kode : 123
Nama : Mas
Jenis Kelamin : Pria
Usia : 34

Kode : 190
Nama : Bunga
Jenis Kelamin : Wanita
Usia : 27

Kode : 911
Nama : Xeno
Jenis Kelamin : Pria
Usia : 54

Kode : 121
Nama : Shiva
Jenis Kelamin : Wanita
Usia : 25

Kode : 292
Nama : Nur
Jenis Kelamin : Wanita
Usia : 49

Kode : 459
Nama : Moss
Jenis Kelamin : Pria
Usia : 20

Kode : 100
Nama : Vusia
Jenis Kelamin : Wanita
Usia : 57

Kode : 711
Nama : Ursus
Jenis Kelamin : Pria
Usia : 29

Kode : 001
Nama : Jokowi
Jenis Kelamin : Pria
Usia : 67

Data dosen setelah sorting berdasarkan Usia (ASC):

Kode : 459
Nama : Moss
Jenis Kelamin : Pria
Usia : 20

Kode : 121
Nama : Shiva
Jenis Kelamin : Wanita
Usia : 25

Kode : 190
Nama : Bunga
Jenis Kelamin : Wanita
Usia : 27

Kode : 711
Nama : Ursus
Jenis Kelamin : Pria
Usia : 29

Kode : 123
Nama : Mas
Jenis Kelamin : Pria
Usia : 34

Kode : 292
Nama : Nur
Jenis Kelamin : Wanita
Usia : 49

Kode : 911
Nama : Xeno
Jenis Kelamin : Pria
Usia : 54

Kode : 100
Nama : Vusia
Jenis Kelamin : Wanita
Usia : 57

Kode : 111
Nama : Budi
Jenis Kelamin : Pria
Usia : 65

Kode : 001
Nama : Jokowi
Jenis Kelamin : Pria
Usia : 67

```

Data dosen setelah sorting berdasarkan Usia (DSC):
Kode      : 001
Nama      : Jokowi
Jenis Kelamin : Pria
Usia      : 67
-----
Kode      : 111
Nama      : Budi
Jenis Kelamin : Pria
Usia      : 65
-----
Kode      : 100
Nama      : Vusia
Jenis Kelamin : Wanita
Usia      : 57
-----
Kode      : 911
Nama      : Xeno
Jenis Kelamin : Pria
Usia      : 54
-----
Kode      : 292
Nama      : Nur
Jenis Kelamin : Wanita
Usia      : 49
-----

```

```

Kode      : 123
Nama      : Mas
Jenis Kelamin : Pria
Usia      : 34
-----
Kode      : 711
Nama      : Ursus
Jenis Kelamin : Pria
Usia      : 29
-----
Kode      : 190
Nama      : Bunga
Jenis Kelamin : Wanita
Usia      : 27
-----
Kode      : 121
Nama      : Shiva
Jenis Kelamin : Wanita
Usia      : 25
-----
Kode      : 459
Nama      : Moss
Jenis Kelamin : Pria
Usia      : 20
-----

```

3. Push kode program ke github

Kumaaan Jobsheet	
Name	Last commit message
..	
DataDosen.java	Jobsheet
Dosen.java	Jobsheet
DosenMain.java	Jobsheet
Mahasiswa01.java	Jobsheet
MahasiswaBerprestasi01.java	Jobsheet
MahasiswaDemo01.java	Jobsheet
Sorting01.java	Jobsheet
SortingMain01.java	Jobsheet