

LAPORAN HASIL PRAKTIKUM
Algoritma Struktur Data
JOBSHEET 6



Muhammad Fitra Adhim Nurrochman
2441007020089

TI 1E

PROGRAM STUDI D-IV TEKNIK INFORMATIKA
JURUSAN TEKNOLOGI INFORMASI
POLITEKNIK NEGERI MALANG

PERCOBAAN 1

a. SORTING-BUBBLE SORT

Kode program sorting

```
public class Sorting19 {  
  
    int[] data;  
    int jumData;  
  
    Sorting19(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();  
    }  
}
```

Kode Program Main Sorting

```
public class SortingMain19 {  
  
    public static void main(String[] args) {  
        int a[] = {20, 10, 2, 7, 12};  
        Sorting19 dataurut1 = new Sorting19(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();  
    }  
}
```

Hasil Run Kode Program

```
) { java SortingMain19 }  
Data Awal 1  
20 10 2 7 12  
Data sudah diurutkan dengan BUBBLE SORT (ASC)  
2 7 10 12 20
```

b. SORTING-SELECTION SORT

Kode program tambahan

```
void SelectionSort() {
    for (int i = 0; i < jumData - 1; i++) {
        int min = i;
        for (int j = i; j < jumData; j++) {
            if (data[j] < data[min]) {
                min = j;
            }
        }
        int temp = data[i];
        data[i] = data[min];
        data[min] = temp;
    }
}
```

```
int b[] = {30, 20, 2, 8, 14};
Sorting19 dataurut2 = new Sorting19(b, b.length);
System.out.println("Data Awal 2");
dataurut2.tampil();
dataurut2.SelectionSort();
System.out.println("Data sudah diurutkan dengan SELECTION SORT (ASC)");
dataurut2.tampil();
```

Hasil Run Kode Program

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

c. SORTING-INSERTION SORT

Kode Program tambahan

```
void InstertionSort() {
    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();
}
```

```
int c[] = {40, 10, 4, 9, 3};
Sorting19 dataurut3 = new Sorting19(c, c.length);
System.out.println("Data Awal 3");
dataurut3.tampil();
dataurut3.InsertionSort();
System.out.println("Data sudah diurutkan dengan INSERTION SORT (ASC)");
dataurut3.tampil();
```

Hasil Run Kode Program

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

PERTANYAAN 1

1. Baris pertama adalah kondisi yang memeriksa apakah elemen array pada j-1 lebih besar dari elemen pada posisi j, jika kondisi iya maka masuk ke baris ke 2. Baris kedua adalah menyimpan nilai elemen data[j] ke variabel sementara yaitu temp, agar nilai asli tidak hilang saat diganti nilainya. Baris ketiga berfungsi untuk memindahkan nilai dari data[j-1] ke data[j]. Baris ke 4 yaitu nilai yang disimpan dalam variabel temp dipindahkan ke data[j-1]
2. for (int j = i + 1; j < jumData; j++) {
if (data[j] < data[min]) {
min = j;
3. inti dari maksud perulangan tersebut adalah untuk membandingkan nilai elemen array pada posisi i dengan nilai yang disimpan di variabel temp, untuk menggeser elemen elemen yang lebih besar dari temp ke kanan untuk mencari posisi yang tepat untuk nilai temp. perulangan ini akan terus dilakukan sampai nilai temp mendapatkan posisi yang tepat
4. bertujuan untuk menggeser elemen array ke kanan untuk membuat ruang bagi elemen yang akan diletakkan

PERCOBAAN 2

1. Kode Program Mahasiswa19

```

public class Mahasiswa19 {
    String nim;
    String nama;
    String kelas;
    double ipk;

    //konstruktor default
    Mahasiswa19() {

    }
    //konstruktor berparameter (dibuat ada yang nama vaar parameter
    inputnya sama ada yang tidak)
    Mahasiswa19(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);
    }
}

```

2. Kode Program MahasiswaBerprestasi

```

public class MahasiswaBerprestasi {
    Mahasiswa19 [] listMhs= new Mahasiswa19 [5];
    int idx;

    void tambah(Mahasiswa19 m) {
        if (idx<listMhs.length) {
            listMhs[idx]=m;
            idx++;
        } else {
            System.out.println("data sudah penuh");
        }
    }
    void tampil(){
        for (Mahasiswa19 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){
                    Mahasiswa19 tmp = listMhs[j];
                    listMhs[j]=listMhs[j-1];
                    listMhs[j-1]=tmp;
                }
            }
        }
    }
}

```

3. Kode Program MahasiswaDemo19

```
public class MahasiswaDemo19 {
    public static void main(String[] args) {
        MahasiswaBerprestasi list = new MahasiswaBerprestasi();
        Mahasiswa19 m1 = new Mahasiswa19("123","Zidan","2A",3.2);
        Mahasiswa19 m2 = new Mahasiswa19("124","Ayu","2A",3.5);
        Mahasiswa19 m3 = new Mahasiswa19("125","Sofi","2A",3.1);
        Mahasiswa19 m4 = new Mahasiswa19("126","Sita","2A",3.9);
        Mahasiswa19 m5 = new Mahasiswa19("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();
    }
}
```

4. Hasil Run Kode Program

```
-ASD\Jobsheet 6\ ; if ($?) { javac MahasiswaDemo19.java }
```

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

PS C:\Users\noval\OneDrive\Documents\1. KULIAH\Semester 2\Pr

PERTANYAAN 2

1. Jawaban nomer 1
 - a. Agar tahap pengurutan terkontrol sesuai dengan yang diperlukan, elemen terakhir otomatis berada diposisi yang tepat setelah semua elemen lain ditempatkan pada posisinya
 - b. Pada tahap l, sebanyak i elemen terbesar sudah berada diposisi paling kanan, syarat tersebut digunakan supaya tidak mengecek elemen yang sudah diposisi yang benar
 - c. Perulangan i sebanyak 49 kali, i = 0 sampai l = 48, 49 tahap bubble sort yang dilakukan
2. Hasil modifikasi

```
void input(){
    for (int i = 0; i < listMhs.length; i++) {
        System.out.println("Masukkan Data Mahasiswa ke-" + (i+1));
        listMhs[i] = new Mahasiswa19();
        System.out.print("NIM: ");
        listMhs[i].nim = sc.next();
        System.out.print("Nama: ");
        listMhs[i].nama = sc.next();
        System.out.print("kelas: ");
        listMhs[i].kelas = sc.next();
        System.out.print("IPK: ");
        listMhs[i].ipk = sc.nextDouble();
    }
}
```

Hasil run kode program


```
Masukkan Data Mahasiswa ke-1
NIM      : 123
Nama     : adhim
kelas    : 1E
IPK      : 2.3
Masukkan Data Mahasiswa ke-2
NIM      : 124
Nama     : fitra
kelas    : 1E
IPK      : 3.5
Masukkan Data Mahasiswa ke-3
NIM      : 897
Nama     : ahmad
kelas    : 1E
IPK      : 2.9
Masukkan Data Mahasiswa ke-4
NIM      : 652
Nama     : muhammad
kelas    : 1E
IPK      : 3.2
Masukkan Data Mahasiswa ke-5
NIM      : 871
Nama     : cahya
kelas    : 1E
IPK      : 3.6
Data mahasiswa setelah sorting berdasarkan IPK (DESC : )
Nama: cahya
NIM: 871
Kelas: 1E
IPK: 3.6
-----
Nama: fitra
NIM: 124
Kelas: 1E
IPK: 3.5
-----
Nama: muhammad
NIM: 652
Kelas: 1E
IPK: 3.2
-----
Nama: ahmad
NIM: 897
Kelas: 1E
IPK: 2.9
-----
Nama: adhim
NIM: 123
Kelas: 1E
IPK: 2.3
-----
```

D5-G:\Users\nova\OneDrive\Documents\1-KULIAH\Semester 2\Pr

PERCOBAAN 3

1. Penambahan method

```
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;
            }
        }
        Mahasiswa tmp = listMhs[idxMin];
        listMhs[idxMin]=listMhs[i];
        listMhs[i]=tmp;
    }
}
```

2. Hasil Run Kode Program

```
data yang sudah terurut menggunakan SELECTION SORT (ASC)
Nama: FTRA
NIM: 123
Kelas: 1E
IPK: 2.1
-----
Nama: ADHIM
NIM: 234
Kelas: 1E
IPK: 3.3
-----
Nama: QIRA
NIM: 121
Kelas: 1E
IPK: 3.4
-----
Nama: HIBNA
NIM: 832
Kelas: 1E
IPK: 3.6
-----
Nama: AHMAD
NIM: 932
Kelas: 1E
IPK: 4.0
-----
```

PERTANYAAN

1. Proses perulangan trsbut berfungsi untuk mencari nilai ipk terkecil dan mengurutkan nilai ipk dari yang terkecil hingga yang terbesar

PERCOBAAN 4

1. Penambahan method

```
void insertionSort(){
    for(int i=1; i<listMhs.length;i++){
        Mahasiswa19 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 run kode program

```
data yang sudah terurut menggunakan INSERTION SORT (ASC)
Nama: NUKMA
NIM: 899
Kelas: 1E
IPK: 2.7
-----
Nama: FITRA
NIM: 435
Kelas: 1E
IPK: 3.1
-----
Nama: WIRMAN
NIM: 882
Kelas: 1E
IPK: 3.4
-----
Nama: ABI
NIM: 111
Kelas: 1E
IPK: 3.9
-----
Nama: ADHIM
NIM: 543
Kelas: 1E
IPK: 4.0
-----
PS C:\Users\nova\OneDrive\Documents\1. KULTAH\Semester 2\Pr
```

PERTANYAAN

1. Hanya perlu (mengubah lebih besar dari) menjadi (lebih kecil dari)

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

Mengubah yang
semulanya ipk>temp.ipk
menjadi ipk<temp.ipk

Hasilnya

data yang sudah terurut menggunakan INSERTION SORT (DESC)

Nama: fwrtg

NIM: 34521

Kelas: 1e

IPK: 4.0

Nama: egb

NIM: 3456

Kelas: 1e

IPK: 3.8

Nama: qwe

NIM: 123

Kelas: 1e

IPK: 2.3

Nama: wwsxc

NIM: 231

Kelas: 1e

IPK: 1.7

Nama: ewfw

NIM: 234

Kelas: 1e

IPK: 1.2

D5: C:\Users\neval\OneDrive\Documents\1 - KULTAH\Semester 2\Bank

TUGAS

1. Kode program

```
public class DataDosen19 {
    Dosen19 [] DataDosen19 = new Dosen19[10];
    int idx;

    void tambah(Dosen19 dsn){
        if (idx < DataDosen19.length) {
            DataDosen19[idx] = dsn;
            idx++;
        } else {
            System.out.println("Data sudah penuh");
        }
    }

    void tampil() {
        for (int i = 0; i < idx; i++) {
            DataDosen19[i].tampil();
        }
    }

    void SortingASC() {
        for (int i = 0; i < idx - 1; i++) {
            for (int j = 0; j < idx - 1 - i; j++) {
                if (DataDosen19[j].umur > DataDosen19[j + 1].umur) {
                    Dosen19 temp = DataDosen19[j];
                    DataDosen19[j] = DataDosen19[j + 1];
                    DataDosen19[j + 1] = temp;
                }
            }
        }
    }

    void SortingDSC() {
        for (int i = 1; i < idx; i++) {
            Dosen19 temp = DataDosen19[i];
            int j = i;
            while (j > 0 && DataDosen19[j - 1].umur < temp.umur) {
                DataDosen19[j] = DataDosen19[j - 1];
                j--;
            }
            DataDosen19[j] = temp;
        }
    }
}
```



```

import java.util.Scanner;

public class DosenMain19 {
    public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);
        DataDosen19 dosen = new DataDosen19();

        while (true) {
            System.out.println("\nMenu:");
            System.out.println("1. Tambah Data Dosen");
            System.out.println("2. Tampil Data Dosen");
            System.out.println("3. Sorting ASC (Usia Termuda ke Tertua)");
            System.out.println("4. Sorting DSC (Usia Tertua ke Termuda)");
            System.out.println("5. Keluar");
            System.out.print("Pilih menu: ");
            int pilihan = scanner.nextInt();
            scanner.nextLine();

            switch (pilihan) {
                case 1:
                    System.out.print("Kode: ");
                    String kode = scanner.nextLine();
                    System.out.print("Nama: ");
                    String nama = scanner.nextLine();
                    System.out.print("Jenis Kelamin (Laki-laki = true, Perempuan = false): ");
                    boolean jk = scanner.nextBoolean();
                    System.out.print("Umur: ");
                    int umur = scanner.nextInt();
                    scanner.nextLine();
                    dosen.tambah(new Dosen19(kode, nama, jk, umur));
                    break;
                case 2:
                    dosen.tampil();
                    break;
                case 3:
                    dosen.SortingASC();
                    System.out.println("Data setelah sorting ASC:");
                    dosen.tampil();
                    break;
                case 4:
                    dosen.SortingDSC();
                    System.out.println("Data setelah sorting DSC:");
                    dosen.tampil();
                    break;
                case 5:
                    System.out.println("Keluar dari program.");
                    scanner.close();
                    return;
                default:
                    System.out.println("Pilihan tidak valid!");
            }
        }
    }
}

```

Hasil Run


```
Menu:
1. Tambah Data Dosen
2. Tampil Data Dosen
3. Sorting ASC (Usia Termuda ke Tertua)
4. Sorting DSC (Usia Tertua ke Termuda)
5. Keluar
Pilih menu: 1
Kode: 123
Nama: Adhim
Jenis Kelamin (Laki-laki = true, Perempuan = false): true
Umur: 35

Menu:
1. Tambah Data Dosen
2. Tampil Data Dosen
3. Sorting ASC (Usia Termuda ke Tertua)
4. Sorting DSC (Usia Tertua ke Termuda)
5. Keluar
Pilih menu: 1
Kode: 332
Nama: Iyul
Jenis Kelamin (Laki-laki = true, Perempuan = false): false
Umur: 21

Menu:
1. Tambah Data Dosen
2. Tampil Data Dosen
3. Sorting ASC (Usia Termuda ke Tertua)
4. Sorting DSC (Usia Tertua ke Termuda)
5. Keluar
Pilih menu: 1
Kode: 990
Nama: nimas
Jenis Kelamin (Laki-laki = true, Perempuan = false): false
Umur: 28

Menu:
1. Tambah Data Dosen
2. Tampil Data Dosen
3. Sorting ASC (Usia Termuda ke Tertua)
4. Sorting DSC (Usia Tertua ke Termuda)
5. Keluar
Pilih menu: 2
kode dosen: 123
nama dosen: Adhim
jenis kelamin dosen: Perempuan
umur dosen: 35
kode dosen: 332
nama dosen: Iyul
jenis kelamin dosen: Laki-laki
umur dosen: 21
kode dosen: 990
nama dosen: nimas
jenis kelamin dosen: Laki-laki
umur dosen: 28

Menu:
1. Tambah Data Dosen
2. Tampil Data Dosen
3. Sorting ASC (Usia Termuda ke Tertua)
4. Sorting DSC (Usia Tertua ke Termuda)
5. Keluar
Pilih menu: 3
Data setelah sorting ASC:
kode dosen: 332
nama dosen: Iyul
jenis kelamin dosen: Laki-laki
umur dosen: 21
kode dosen: 990
nama dosen: nimas
jenis kelamin dosen: Laki-laki
umur dosen: 28
```

```
kode dosen: 123
nama dosen: Adhim
jenis kelamin dosen: Perempuan
umur dosen: 35

Menu:
1. Tambah Data Dosen
2. Tampil Data Dosen
3. Sorting ASC (Usia Termuda ke Tertua)
4. Sorting DSC (Usia Tertua ke Termuda)
5. Keluar
Pilih menu: 4
Data setelah sorting DSC:
kode dosen: 123
nama dosen: Adhim
jenis kelamin dosen: Perempuan
umur dosen: 35
kode dosen: 990
nama dosen: nimas
jenis kelamin dosen: Laki-laki
umur dosen: 28
kode dosen: 332
nama dosen: Iyul
jenis kelamin dosen: Laki-laki
umur dosen: 21

Menu:
1. Tambah Data Dosen
2. Tampil Data Dosen
3. Sorting ASC (Usia Termuda ke Tertua)
4. Sorting DSC (Usia Tertua ke Termuda)
5. Keluar
Pilih menu: 
```