

**LAPORAN HASIL PRAKTIKUM ALGORITMA DAN
STRUKTUR DATA
JOBSHEET 7**



**NAMA : MOHAMAT FAUZI ROHMAN
NIM : 244107020067
KELAS : 1E**

**PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNOLOGI INFORMASI
POLITEKNIK NEGERI MALANG
2025**

JOBSHEET 7

7. Praktikum

7.1 Praktikum 1 - Searching / Pencarian Menggunakan Algoritma Sequential Search

7.1.1 Langkah-langkah Percobaan

1. Pada pertemuan Jobsheet 7 ini akan menggunakan class Mahasiswa, MahasiswaBerprestasi, dan MahasiswaDemo pada pertemuan Jobsheet 6 sebelumnya
2. Buat folder baru bernama Jobsheet7 di dalam repository Praktikum ASD, kemudian buka ketiga class dari Jobsheet 6 tersebut dan copy ke folder Jobsheet 7
3. Tambahkan method sequentialSearching bertipe integer dengan parameter cari bertipe double pada class MahasiswaBerprestasi. Kemudian Deklarasikan isi method sequentialSearching dengan algoritma pencarian data menggunakan teknik sequential searching.

```
int sequentialSearching(double cari){
    int posisi = -1;
    for (int j = 0; j < listMhs.length; j++){
        if (listMhs[j].ipk == cari) {
            posisi = j;
        }
    }
    return posisi;
}
```

4. Buatlah method tampilPoisisi bertipe void dan Deklarasikan isi dari method tampilPoisisi pada class MahasiswaBerprestasi

```
void tampilPoisisi(double x, int pos){
    if (pos != -1) {
        System.out.println("Data mahasiswa dengan IPK : " + x +
" ditemukan pada indeks " + pos);
    } else {
        System.out.println("Data " + x + " tidak ditemukan");
    }
}
```

5. Pada class MahasiswaBerprestasi, buatlah method tampilDataSearch bertipe void dan Deklarasikan isi dari method tampilDataSearch

```
void tampilDataSearch(double x, int pos){
    if (pos != -1) {
        System.out.println("nim\t : " + listMhs[pos].nim);
        System.out.println("nama\t : " + listMhs[pos].nama);
        System.out.println("kelas\t : " + listMhs[pos].kelas);
        System.out.println("ipk\t : " + x);
    } else {
        System.out.println("Data mahasiswa dengan IPK " + x +
" tidak ditemukan");
    }
}
```

6. Pada class MahasiswaDemo , tambahkan kode program berikut ini untuk melakukan pencarian data dengan algoritma sequ ential searching

```
import java.util.Scanner;
public class MahasiswaDemo16 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int jumMhs = 5;
        MahasiswaBerprestasi16 list = new MahasiswaBerprestasi16();

        for (int i = 0; i < jumMhs; 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();

            Mahasiswa16 mhs = new Mahasiswa16(nim, nama, kelas, ipk);
            list.tambah(mhs);
        }
        list.tampil();

        System.out.println("-----");
        System.out.println("Pencarian data");
        System.out.println("-----");
        System.out.println("Masukkan IPK mahasiswa yang dicari: ");
        System.out.print("IPK: ");
        double cari = sc.nextDouble();

        System.out.println("Menggunakan Sequential Searching");
        double posisi = list.sequentialSearching(cari);
        int pss = (int)posisi;
        list.tampilPosisi(cari, pss);
        list.tampilDataSearch(cari, pss);
    }
}
```

7. Jalankan dan amati hasilnya

7.1.2 Verifikasi Hasil Percobaan

```
Masukkan Data Mahasiswa ke-1
NIM : 111
Nama : adi
Kelas : 2
IPK : 3.6
-----
Masukkan Data Mahasiswa ke-2
NIM : 222
Nama : tio
Kelas : 2
IPK : 3.8
-----
Masukkan Data Mahasiswa ke-3
NIM : 333
Nama : ila
Kelas : 2
IPK : 3.0
-----
Masukkan Data Mahasiswa ke-4
NIM : 444
Nama : lia
Kelas : 2
IPK : 3.5
-----
Masukkan Data Mahasiswa ke-5
NIM : 555
Nama : fia
Kelas : 2
IPK : 3.3
-----
Nama : adi
NIM : 111
Kelas: 2
IPK : 3.6
-----
Nama : tio
NIM : 222
Kelas: 2
IPK : 3.8
-----
Nama : ila
NIM : 333
Kelas: 2
IPK : 3.0
-----
Nama : lia
NIM : 444
Kelas: 2
IPK : 3.5
-----
Nama : fia
NIM : 555
Kelas: 2
IPK : 3.3
-----
Pencarian data
-----
IPK: 3.5
Menggunakan Sequential Searching
Data mahasiswa dengan IPK :3.5 ditemukan pada indeks 3
nim : 444
nama : lia
kelas : 2
ipk : 3.5
```

7.1.3 Pertanyaan

1. Jelaskan perbedaan metod tampilDataSearch dan tampilPosisi pada class MahasiswaBerprestasi!
2. Jelaskan fungsi break pada kode program dibawah ini!

```
if (listMhs[j].ipk==cari){
    posisi=j;
    break;
}
```

Jawab :

1. tampilDataSearch digunakan untuk menampilkan data mahasiswa berupa nim, nama, kelas dan ipk yang sebelumnya sudah diisi
Sedangkan tampilPosisi digunakan untuk menampilkan posisi/indeks data mahasiswa yang dicari
2. Fungsi kode tersebut untuk mencari IPK didalam listMhs. Jika ditemukan, kode ini menyimpan indeks mahasiswa tersebut dalam variabel posisi dan pencarian dihentikan dengan break

7.2 Praktikum 2- (Sorting Menggunakan Array of Object)

7.2.1 Langkah-langkah Percobaan

1. Pada percobaan 7.1.1 (sequential search) tambahkan method `findBinarySearch` bertipe integer pada class `MahasiswaBerprestasi`. Kemudian Deklarasikan isi method `findBinarySearch` dengan algoritma pencarian data menggunakan teknik binary searching

```
int findBinarySearch(double cari, int left, int right){
    int mid;
    if (right >= left) {
        mid = (left+right)/2;
        if (cari == listMhs[mid].ipk) {
            return (mid);
        }
        else if (listMhs[mid].ipk > cari) {
            return findBinarySearch(cari, left, mid-1);
        }
        else {
            return findBinarySearch(cari, mid+1, right);
        }
    }
    return -1;
}
```

2. Panggil method `findBinarySearch` terdapat pada class `MahasiswaBerprestasi` di kelas `MahasiswaDemo`. Kemudian panggil method `tampilPosisi` dan `tampilDataSearch`

```
list.tampil();

System.out.println("-----");
System.out.println("Pencarian data");
System.out.println("-----");
System.out.println("Masukkan IPK mahasiswa yang dicari: ");
System.out.print("IPK: ");
double cari = sc.nextDouble();

System.out.println("-----");
System.out.println("Menggunakan Binary Search");
System.out.println("-----");
double posisi2 = list.findBinarySearch(cari, 0, jumMhs-1);
int pss2 = (int)posisi2;
list.tampilPosisi(cari, pss2);
list.tampilDataSearch(cari, pss2);
```

3. Jalankan dan amati hasilnya (inputkan data IPK secara terurut -ASC seperti verifikasi hasil percobaan dibawah ini).

7.2.2 Verifikasi Hasil Percobaan

```
Masukkan Data Mahasiswa ke-1
NIM : 111
Nama : adi
Kelas : 2
IPK : 3.1
-----
Masukkan Data Mahasiswa ke-2
NIM : 222
Nama : ila
Kelas : 2
IPK : 3.2
-----
Masukkan Data Mahasiswa ke-3
NIM : 333
Nama : lia
Kelas : 2
IPK : 3.3
-----
Masukkan Data Mahasiswa ke-4
NIM : 444
Nama : susi
Kelas : 2
IPK : 3.5
-----
Masukkan Data Mahasiswa ke-5
NIM : 555
Nama : anita
Kelas : 2
IPK : 3.7
-----
```

```
Nama : adi
NIM : 111
Kelas: 2
IPK : 3.1
-----
```

```
Nama : ila
NIM : 222
Kelas: 2
IPK : 3.2
-----
```

```
Nama : lia
NIM : 333
Kelas: 2
IPK : 3.3
-----
```

```
Nama : susi
NIM : 444
Kelas: 2
IPK : 3.5
-----
```

```
Nama : anita
NIM : 555
Kelas: 2
IPK : 3.7
-----
```

```
-----
Pencarian data
-----
```

```
Masukkan IPK mahasiswa yang dicari:
```

```
IPK: 3.7
```

```
Menggunakan Sequential Searching
```

```
Data mahasiswa dengan IPK :3.7 ditemukan pada indeks 4
```

```
nim : 555
```

```
nama : anita
```

```
kelas : 2
```

```
ipk : 3.7
-----
```

```
Menggunakan Binary Search
```

```
-----
Data mahasiswa dengan IPK :3.7 ditemukan pada indeks 4
```

```
nim : 555
```

```
nama : anita
```

```
kelas : 2
```

```
ipk : 3.7
```

7.2.3 Pertanyaan

1. Tunjukkan pada kode program yang mana proses divide dijalankan!
2. Tunjukkan pada kode program yang mana proses conquer dijalankan!
3. Jika data IPK yang dimasukkan tidak urut. Apakah program masih dapat berjalan? Mengapa demikian!
4. Jika IPK yang dimasukkan dari IPK terbesar ke terkecil (misal : 3.8, 3.7, 3.5, 3.4, 3.2) dan elemen yang dicari adalah 3.2. Bagaimana hasil dari binary search? Apakah sesuai? Jika tidak sesuai maka ubahlah kode program binary search agar hasilnya sesuai
5. Modifikasilah program diatas yang mana jumlah mahasiswa yang di inputkan sesuai dengan masukan dari keyboard.

Jawab :

1. Proses divide terdapat pada kode program bagian :

```
mid = (left + right) / 2;
```

2. Terdapat 2 pemanggilan fungsi `findBinarySearch` yang digunakan untuk mencari di bagian kiri atau kanan dari array. :

```
return findBinarySearch(cari, left, mid - 1);
```

```
return findBinarySearch(cari, mid+1, right);
```

3. Berjalan tetapi program tidak berfungsi, karena data yang dicari harus urut. Jika data tidak urut, maka pencarian tidak dapat menemukan elemen yang dicari meskipun elemen tersebut ada dalam array.
4. Hasil dari binary search tidak sesuai, karena binary search yang digunakan ialah data dengan pengurutan ascending (dari terkecil ke terbesar)

Agar sesuai, ubah kode program pada bagian tanda lebih besar menjadi lebih kecil seperti berikut

```
else if (listMhs[mid].ipk < cari)
```

```
Nama : adi
NIM : 111
Kelas: 2
IPK : 3.8
-----
Nama : lia
NIM : 222
Kelas: 2
IPK : 3.7
-----
Nama : ila
NIM : 333
Kelas: 2
IPK : 3.5
-----
Nama : bagus
NIM : 444
Kelas: 2
IPK : 3.4
-----
Nama : roy
NIM : 555
Kelas: 2
IPK : 3.2
-----
```

```
Menggunakan Binary Search
-----
Data mahasiswa dengan IPK :3.2 ditemukan pada indeks 4
nim      : 555
nama     : roy
kelas    : 2
ipk      : 3.2
```

5. MahasiswaDemo16

```
import java.util.Scanner;
public class MahasiswaDemo16 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        System.out.print("Masukkan Jumlah Mahasiswa : ");
        int jumMhs = sc.nextInt();
        sc.nextLine();

        MahasiswaBerprestasi16 list = new MahasiswaBerprestasi16(jumMhs);

        for (int i = 0; i < jumMhs; 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();

            Mahasiswa16 mhs = new Mahasiswa16(nim, nama, kelas, ipk);
            list.tambah(mhs);
        }
        list.tampil();

        System.out.println("-----");
        System.out.println("Pencarian data");
        System.out.println("-----");
        System.out.print("Masukkan IPK mahasiswa yang dicari: ");
        System.out.print("IPK: ");
        double cari = sc.nextDouble();

        System.out.println("-----");
        System.out.println("Menggunakan Binary Search");
        System.out.println("-----");
        double posisi2 = list.findBinarySearch(cari, 0, jumMhs-1);
        int pss2 = (int)posisi2;
        list.tampilPosisi(cari, pss2);
        list.tampilDataSearch(cari, pss2);
    }
}
```


Hasil

```
Masukkan Jumlah Mahasiswa : 2
Masukkan Data Mahasiswa ke-1
NIM : 111
Nama : fauzi
Kelas : 2
IPK : 3.2
-----
Masukkan Data Mahasiswa ke-2
NIM : 222
Nama : agus
Kelas : 2
IPK : 3.5
-----
Nama : fauzi
NIM : 111
Kelas : 2
IPK : 3.2
-----
Nama : agus
NIM : 222
Kelas : 2
IPK : 3.5
-----
Pencarian data
-----
Masukkan IPK mahasiswa yang dicari:
IPK: 3.5
-----
Menggunakan Binary Search
-----
Data mahasiswa dengan IPK :3.5 ditemukan pada indeks 1
nim : 222
nama : agus
kelas : 2
ipk : 3.5
```

Latihan Praktikum

1. Pada Latihan praktikum pertemuan sebelumnya pada Jobsheet 6 yang terdapat 3 class yaitu Dosen, DataDosen , dan DosenDemo, tambahkan method:

a. PencarianDataSequential : digunakan untuk mencari data dosen berdasarkan nama dengan algoritma sequential search.

b. PencarianDataBinary : digunakan untuk mencari data dosen berdasarkan usia dengan algoritma Binary Search.

c. Buat aturan untuk mendeteksi hasil pencarian lebih dari 1 hasil dalam bentuk kalimat peringatan! Pastikan algoritma yang diterapkan sesuai dengan kasus yang diberikan!

Dosen16

```
public class Dosen16 {
    String kode;
    String nama;
    boolean jenisKelamin;
    int usia;

    Dosen16(){

    }

    Dosen16(String kd, String name, boolean jk, int age) {
        kode = kd;
        nama = name;
        jenisKelamin = jk;
        usia = age;
    }

    public void tampil() {
        System.out.println("Kode Dosen    : " + kode);
        System.out.println("Nama Dosen    : " + nama);
        System.out.println("Jenis Kelamin: " + (jenisKelamin ? "Laki-laki" :
"Perempuan"));
        System.out.println("Usia          : " + usia + " tahun");
    }

    int getUsia() {
        return usia;
    }
}
```

DataDosen16

```
public class DataDosen16 {
    Dosen16 [] dataDosen = new Dosen16[10];
    int idx;

    void tambah (Dosen16 dsn){
        if (idx < dataDosen.length) {
            dataDosen[idx] = dsn;
            idx++;
        }else {
            System.out.println("Data sudah penuh");
        }
    }

    void tampil(){
        for (int i = 0; i < idx; i++){
            dataDosen[i].tampil();
            System.out.println("-----");
        }
    }

    void sortingASC(){
        for (int i = 0; i < idx-1; i++){
            for (int j = 0; j < idx-1-i; j++ ){
                if (dataDosen[j].usia > dataDosen[j+1].usia) {
                    Dosen16 temp = dataDosen[j];
                    dataDosen[j] = dataDosen[j+1];
                    dataDosen[j+1] = temp;
                }
            }
        }
    }

    void SortingDSC(){
        for (int i = 0; i < idx-1; i++){
            int idxMax = i;
            for (int j = i+1; j < idx; j++){
                if (dataDosen[j].usia > dataDosen[idxMax].usia) {
                    idxMax = j;
                }
            }
            Dosen16 temp = dataDosen[idxMax];
            dataDosen[idxMax] = dataDosen[i];
            dataDosen[i] = temp;
        }
    }

    int PencarianDataSequential16(String cari){
        int posisi = -1;
        for (int j = 0; j < idx; j++){
            if (dataDosen[j].nama.equals(cari)) {
                posisi = j;
                break;
            }
        }
        return posisi;
    }
}
```

```

void tampilPosisi(String x, int pos){
    if (pos != -1) {
        System.out.println("Data Dosen dengan nama " + x + " ditemukan pada indeks " +
pos);
    } else {
        System.out.println("Nama " + x + " tidak ditemukan");
    }
}

void tampilDataSearch(String x, int pos){
    if (pos != -1) {
        System.out.println("kode           : " + dataDosen[pos].kode);
        System.out.println("nama           : " + dataDosen[pos].nama);
        System.out.println("jenis kelamin : " + (dataDosen[pos].jenisKelamin ? "Laki-
laki" : "Perempuan"));
        System.out.println("usia           : " + dataDosen[pos].usia);
    } else {
        System.out.println("Data dosen dengan nama " + x + " tidak ditemukan");
    }
}

int PencarianDataBinary16(int cariUsia, int left, int right){
    int mid;
    if (right >= left) {
        mid = (left+right)/2;
        if (cariUsia == dataDosen[mid].usia) {
            return (mid);
        }
        else if (dataDosen[mid].usia > cariUsia) {
            return PencarianDataBinary16(cariUsia, left, mid-1);
        }
        else {
            return PencarianDataBinary16(cariUsia, mid+1, right);
        }
    }
    return -1;
}

void posisiUsia(int cariUsia, int posisi2) {
    if (posisi2 != -1) {
        System.out.println("Usia ditemukan pada indeks " + posisi2);
    } else {
        System.out.println("Data dengan usia " + cariUsia + " tidak ditemukan.");
    }
}

void tampilUsia(int cariUsia, int posisi2) {
    if (posisi2 != -1) {
        System.out.println("kode           : " + dataDosen[posisi2].kode);
        System.out.println("nama           : " + dataDosen[posisi2].nama);
        System.out.println("jenis kelamin : " + (dataDosen[posisi2].jenisKelamin ?
"Laki-laki" : "Perempuan"));
        System.out.println("usia           : " + dataDosen[posisi2].usia);
    } else {
        System.out.println("Data dosen dengan usia " + cariUsia + " tidak ditemukan");
    }
}
}

```

DosenDemo16

```

import java.util.Scanner;
public class DosenDemol6 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        DataDosen16 data = new DataDosen16();
        int pilihan;

        do {
            System.out.println("Menu:");
            System.out.println("1. Tambah Data Dosen");
            System.out.println("2. Tampil Data Dosen");
            System.out.println("3. Sorting ASC (dari termuda ke tertua)");
            System.out.println("4. Sorting DSC (dari tertua ke termuda)");
            System.out.println("5. Cari Nama Dosen");
            System.out.println("6. Cari Usia");
            System.out.print("Pilih menu: ");
            pilihan = sc.nextInt();
            sc.nextLine();

            switch (pilihan) {
                case 1:
                    System.out.print("Kode          : ");
                    String kode = sc.nextLine();
                    System.out.print("Nama          : ");
                    String nama = sc.nextLine();
                    System.out.print("Jenis Kelamin: ");
                    boolean jenisKelamin = sc.nextBoolean();
                    System.out.print("Usia          : ");
                    int usia = sc.nextInt();
                    sc.nextLine();

                    Dosen16 dosen = new Dosen16(kode, nama, jenisKelamin, usia);
                    data.tambah(dosen);
                    break;

                case 2:
                    data.tampil();
                    break;

                case 3:
                    data.sortingASC();
                    System.out.println("Sorting dari termuda ke tertua");
                    data.tampil();
                    break;

                case 4:
                    data.SortingDSC();
                    System.out.println("Sorting dari tertua ke termuda");
                    data.tampil();
                    break;
            }
        } while (pilihan != 0);
    }
}

```

```

case 5:
    System.out.print("Nama : ");
    String cari = sc.nextLine();

    System.out.println("Menggunakan Sequential Searching");
    int posisi = data.PencarianDataSequential16(cari);
    data.tampilPosisi(cari, posisi);
    data.tampilDataSearch(cari, posisi);
    break;
case 6:
    System.out.print("Usia : ");
    int cariUsia = sc.nextInt();

    System.out.println("Menggunakan Binary Search");
    int posisi2 = data.PencarianDataBinary16(cariUsia, 0, 1);
    data.posisiUsia(cariUsia, posisi2);
    data.tampilUsia(cariUsia, posisi2);
    break;

    }
    System.out.println();
} while (pilihan != 7);
}
}

```

Hasil

```

Menu:
1. Tambah Data Dosen
2. Tampil Data Dosen
3. Sorting ASC (dari termuda ke tertua)
4. Sorting DSC (dari tertua ke termuda)
5. Cari Nama Dosen
6. Cari Usia
Pilih menu: 1
Kode      : 111
Nama      : agus
Jenis Kelamin: true
Usia      : 25

Menu:
1. Tambah Data Dosen
2. Tampil Data Dosen
3. Sorting ASC (dari termuda ke tertua)
4. Sorting DSC (dari tertua ke termuda)
5. Cari Nama Dosen
6. Cari Usia
Pilih menu: 1
Kode      : 222
Nama      : roy
Jenis Kelamin: true
Usia      : 30

Menu:
1. Tambah Data Dosen
2. Tampil Data Dosen
3. Sorting ASC (dari termuda ke tertua)
4. Sorting DSC (dari tertua ke termuda)
5. Cari Nama Dosen
6. Cari Usia
Pilih menu: 2
Kode Dosen : 111
Nama Dosen : agus
Jenis Kelamin: Laki-laki
Usia       : 25 tahun
-----
Kode Dosen : 222
Nama Dosen : roy
Jenis Kelamin: Laki-laki
Usia       : 30 tahun
-----

```

```
Menu:
1. Tambah Data Dosen
2. Tampil Data Dosen
3. Sorting ASC (dari termuda ke tertua)
4. Sorting DSC (dari tertua ke termuda)
5. Cari Nama Dosen
6. Cari Usia
Pilih menu: 5
Nama : agus
Menggunakan Sequential Searching
Data Dosen dengan nama agus ditemukan pada indeks 0
kode      : 111
nama      : agus
jenis kelamin : laki-laki
usia      : 25
```

```
Menu:
1. Tambah Data Dosen
2. Tampil Data Dosen
3. Sorting ASC (dari termuda ke tertua)
4. Sorting DSC (dari tertua ke termuda)
5. Cari Nama Dosen
6. Cari Usia
Pilih menu: 6
Usia : 30
Menggunakan Binary Search
Usia ditemukan pada indeks 1
kode      : 222
nama      : roy
jenis kelamin : laki-laki
usia      : 30
```