

LAPORAN HASIL PRAKTIKUM

ALGORITMA DAN STRUKTUR DATA

JOBSHEET 7



NAMA :MAULIDYAAFRIANI

NIM: 2441070200559

KELAS : 1E

PROGRAM STUDI TEKNIK INFORMATIKA

JURUSAN TEKNOLOGI INFORMASI

POLITEKNIK NEGERI MALANG 2025

1.1 PERCOBAAN 1

1. Pada pertemuan Jobsheet 7 ini akan menggunakan class **Mahasiswa**<no Presensi>, **MahasiswaBerprestasi**<no Presensi>, dan **MahasiswaDemo**<no presensi> 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**<no presensi>. Kemudian Deklarasikan isi method **sequentialSearching** dengan algoritma pencarian data menggunakan teknik sequential searching.

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

4. Buatlah method **tampilPoisisi** bertipe void dan Deklarasikan isi dari method **tampilPoisisi** pada class **MahasiswaBerprestasi**<no presensi>.

```
void tampilPosisi(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**<no presensi>, 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**<noPresensi> , tambahkan kode program berikut ini untuk melakukan pencarian data dengan algoritma sequential searching.

```

import java.util.Scanner;
public class MahasiswaDemo15 {
    public static void main(String[] args) {
        MahasiswaBerprestasi15 list = new MahasiswaBerprestasi15();
        Scanner sc = new Scanner(System.in);
        int jumMhs=5;

        for(int i=0; i<jumMhs; i++){
            System.out.println("Masukkan data mahasiswa ke-" + (i + 1));
            System.out.println("NIM: ");
            String nim = sc.nextLine();
            System.out.println("Nama: ");
            String nama = sc.nextLine();
            System.out.println("Kelas: ");
            String kelas = sc.nextLine();
            System.out.println("IPK: ");
            String ip =sc.nextLine();
            Double ipk = Double.parseDouble(ip);
            System.out.println("-----");
            list.tambah(new Mahasiswa15(nim, nama, kelas, ipk));
        }

        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.sequentialSerching(cari);
        int pss = (int)posisi;
        list.tampilPosisi(cari, pss);
        list.tampilDataSearch(cari, pss);
    }
}

```

Hasil run

```

Masukkan data mahasiswa ke-1
NIM: 98
Nama: litya
Kelas: 12
IPK: 98
-----
Masukkan data mahasiswa ke-2
NIM: 9808
Nama: jiuw
Kelas: 6
IPK: 67
-----
Masukkan data mahasiswa ke-3
NIM: 9801
Nama: kia
Kelas: 3
IPK: 78
-----
Masukkan data mahasiswa ke-4
NIM: 91801
Nama: popo
Kelas: 7
IPK: 99
-----
Masukkan data mahasiswa ke-5
NIM: 0101
Nama: ais
Kelas: 7
IPK: 98
-----
Nama : litya
NIM : 98
Kelas : 12
ipk : 98.0
-----
Nama : jiuw
NIM : 9808
Kelas : 6
ipk : 67.0
-----
Nama : kia
NIM : 9801
Kelas : 3
ipk : 78.0
-----
Nama : popo
NIM : 91801
Kelas : 7
ipk : 99.0
-----
Pencarian data
Pencarian data
-----
Masukkan ipk mahasiswa yang dicari:
IPK: 98
Menggunakan sequential searching
Data mahasiswa dengan IPK: 98.0 ditemukan pada indeks 0
NIM : 98
Nama : litya
Kelas : 12
IPK : 98.0

```

PERTANYAAN

- Jelaskan perbedaan metod tampilDataSearch dan tampilPosisi pada class MahasiswaBerprestasi!
 - tampilDataSearch: Menampilkan detail lengkap mahasiswa berdasarkan IPK.
 - tampilPosisi: Menampilkan posisi (indeks) mahasiswa dalam array berdasarkan IPK.
- Jelaskan fungsi break pada kode program di bawah ini!

```

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

```

- Menghentikan perulangan lebih awal saat kondisi terpenuhi agar pencarian tidak dilanjutkan ke elemen berikutnya.

1.2 PERCOBAAN 2

- Pada percobaan 6.2.1 (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**

```

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).

```
Masukkan data mahasiswa ke-1
NIM: 0394
Nama: lilya
Kelas: 3g
IPK: 78
-----
Masukkan data mahasiswa ke-2
NIM: 0530
Nama: popo
Kelas: 8
IPK: 90
-----
Masukkan data mahasiswa ke-3
NIM: 959160
Nama: ais
Kelas: 3
IPK: 45
-----
Masukkan data mahasiswa ke-4
NIM: 3030
Nama: kia
Kelas: 2
IPK: 99
-----
Masukkan data mahasiswa ke-5
NIM: 90090
Nama: iki
Kelas: 8
IPK: 9
-----
Nama : lilya
NIM : 0394
Kelas : 3g
ipk : 78.0
-----
Nama : popo
NIM : 0530
Kelas : 8
ipk : 90.0
-----
Nama : ais
NIM : 959160
Kelas : 3
ipk : 45.0
-----
Nama : kia
NIM : 3030
Kelas : 2
ipk : 99.0
-----
Nama : iki
NIM : 90090
Kelas : 8
ipk : 9.0
-----
Pencarian data
-----
Masukkan ipk mahasiswa yang dicari:
IPK: 9
-----
Menggunakan binary search
-----
Data 9.0 tidak ditemukan
Data mahasiswa dengan IPK 9.0 tidak ditemukan
```

PERTANYAAN

1. Tunjukkan pada kode program yang mana proses divide dijalankan!
⇒ Program Divide terjadi ketika nilai mid dari rentang pencarian dihitung dengan rumus :

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

2. Tunjukkan pada kode program yang mana proses conquer dijalankan!
⇒ Proses conquer terjadi dalam pemanggilan rekursif berdasarkan perbandingan

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

3. Jika data IPK yang dimasukkan tidak urut. Apakah program masih dapat berjalan? Mengapa demikian!
⇒ Program masih tetap bisa berjalan tetapi tidak menemukan data yang di cari walaupun data tersebut ada
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
⇒ Mengganti if (listMhs[mid].ipk > cari) menjadi if (listMhs[mid].ipk < cari)

```
} else if (listMhs[mid].ipk > cari) {
```

Menjadi

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

5. Modifikasilah program diatas yang mana jumlah mahasiswa yang di inputkan sesuai dengan masukan dari keyboard.
⇒ Modifikasi

```
System.out.print(s:"Masukkan jumlah mahasiswa: ");
int jumMhs = Integer.parseInt(sc.nextLine());
```

Kode Program Modifikasi

```
import java.util.Scanner;
public class MahasiswaDemo15
{
    public static void main(String[] args) {
        MahasiswaBerprestasi15 list = new MahasiswaBerprestasi15();
        Scanner sc = new Scanner(System.in);

        System.out.print("Masukkan jumlah mahasiswa: ");
        int jumMhs = Integer.parseInt(sc.nextLine());

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

            System.out.println("Masukkan data mahasiswa ke-" + (i + 1));
            System.out.println("NIM: ");
            String nim = sc.nextLine();
            System.out.println("Nama: ");
            String nama = sc.nextLine();
            System.out.println("Kelas: ");
            String kelas = sc.nextLine();
            System.out.println("IPK: ");
            String ip =sc.nextLine();
            Double ipk = Double.parseDouble(ip);
            System.out.println("-----");
            list.tambah(new Mahasiswa15(nim, nama, kelas, ipk));
        }

        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("Menggunakan sequential searching");
        double posisi = list.sequentialSerching(cari);
        int pss = (int)posisi;
        list.tampilPosisi(cari, pss);
        list.tampilDataSearch(cari, pss);
    }
}
```

LATIHAN PRAKTIKUM

KODE PROGRAM

Kode Dosen15.java

```
public class Dosen15 {
    String kode;
    String nama;
    boolean jenisKelamin;
    int usia;
```

```

Dosen15(String kd, String name, boolean jk, int age) {
    kode = kd;
    nama = name;
    jenisKelamin = jk;
    usia = age;
}

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

```

Kode DataDosen15.java

```

public class DataDosen15 {
    Dosen15[] dataDosen = new Dosen15[10];
    int idx;

    void tambah(Dosen15 dsn) {
        if (idx < dataDosen.length) {
            dataDosen[idx++] = dsn;
        } else {
            System.out.println("Data Dosen Sudah Penuh!");
        }
    }

    void tampil() {
        if (idx == 0) {
            System.out.println("Tidak Ada Data Dosen.");
            return;
        }
        for (int i = 0; i < idx; i++) {
            dataDosen[i].tampil();
        }
    }

    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) {
                    Dosen15 temp = dataDosen[j];
                    dataDosen[j] = dataDosen[j + 1];
                    dataDosen[j + 1] = temp;
                }
            }
        }
    }
}

```



```

void sortingDSC() {
    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) {
                Dosen15 temp = dataDosen[j];
                dataDosen[j] = dataDosen[j + 1];
                dataDosen[j + 1] = temp;
            }
        }
    }
}

void pencarianDataSequential(String nama) {
    boolean ditemukan = false;
    int hitung = 0;
    for (int i = 0; i < idx; i++) {
        if (dataDosen[i].nama.equalsIgnoreCase(nama)) {
            dataDosen[i].tampil();
            ditemukan = true;
            hitung++;
        }
    }
    if (!ditemukan) {
        System.out.println("Data dengan nama '" + nama + "' tidak ditemukan.");
    }
}

void pencarianDataBinary(int usiaCari) {
    sortingASC();
    int left = 0, right = idx - 1;
    boolean ditemukan = false;

    while (left <= right) {
        int mid = (left + right) / 2;
        if (dataDosen[mid].usia == usiaCari) {
            int awal = mid, akhir = mid;

            while (awal - 1 >= 0 && dataDosen[awal - 1].usia == usiaCari) {
                awal--;
            }
            while (akhir + 1 < idx && dataDosen[akhir + 1].usia ==
usiaCari) {
                akhir++;
            }

            int jumlah = akhir - awal + 1;
            System.out.println("Ditemukan " + jumlah + " data dengan usia
" + usiaCari + ":");
            for (int i = awal; i <= akhir; i++) {
                dataDosen[i].tampil();
            }
        }
    }
}

```

```

        ditemukan = true;
        break;
    } else if (usiaCari < dataDosen[mid].usia) {
        right = mid - 1;
    } else {
        left = mid + 1;
    }
}

if (!ditemukan) {
    System.out.println("Data dengan usia " + usiaCari + " tidak
ditemukan.");
}
}
}

```

Kode DosenMain15.java

```

import java.util.Scanner;

public class DosenMain15 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        DataDosen15 datadsn = new DataDosen15();
        int pilihan;

        do {
            System.out.println("\n===== MENU
=====");
            System.out.println("1. Tambah Data Dosen");
            System.out.println("2. Tampilkan Data Dosen");
            System.out.println("3. Cari Data Dosen (Nama - Sequential
Search)");
            System.out.println("4. Cari Data Dosen (Usia - Binary Search)");
            System.out.println("5. Keluar");
            System.out.print("Pilih Menu: ");
            pilihan = sc.nextInt();
            sc.nextLine();

            switch (pilihan) {
                case 1:
                    if (datadsn.idx >= 10) {
                        System.out.println("Kapasitas data sudah penuh.");
                        break;
                    }
                    System.out.print("Masukkan jumlah data yang ingin
ditambahkan: ");
                    int jumlah = sc.nextInt();
                    sc.nextLine();

                    for (int i = 0; i < jumlah; i++) {
                        if (datadsn.idx >= 10) {

```

```

                System.out.println("Kapasitas data sudah
penuh.");
                break;
            }
            System.out.println("Data Dosen ke-" + (datadsn.idx +
1));
            System.out.print("Kode Dosen      : ");
            String kode = sc.nextLine();
            System.out.print("Nama Dosen      : ");
            String nama = sc.nextLine();
            System.out.print("Jenis Kelamin (L/P): ");
            char jk = sc.next().toUpperCase().charAt(0);
            boolean jenisKelamin = jk == 'L';
            System.out.print("Usia          : ");
            int usia = sc.nextInt();
            sc.nextLine();

            Dosen15 dsn = new Dosen15(kode, nama, jenisKelamin,
usia);

            datadsn.tambah(dsn);
            System.out.println("-----
---");
        }
        break;

    case 2:
        datadsn.tampil();
        break;

    case 3:
        System.out.print("Masukkan nama dosen yang dicari: ");
        String namaCari = sc.nextLine();
        datadsn.pencarianDataSequential(namaCari);
        break;

    case 4:
        System.out.print("Masukkan usia dosen yang dicari: ");
        int usiaCari = sc.nextInt();
        sc.nextLine();
        datadsn.pencarianDataBinary(usiaCari);
        break;

    case 5:
        System.out.println("Keluar dari program.");
        break;
    default:
        System.out.println("Pilihan tidak valid. Coba lagi.");
    }

    } while (pilihan != 5);

    sc.close();

```

Hasil kode

```
===== MENU =====
1. Tambah Data Dosen
2. Tampilkan Data Dosen
3. Cari Data Dosen (Nama - Sequential Search)
4. Cari Data Dosen (Usia - Binary Search)
5. Keluar
Pilih Menu: 1
Masukkan jumlah data yang ingin ditambahkan: 2
Data Dosen ke-1
Kode Dosen      : 99
Nama Dosen      : kaia
Jenis Kelamin (L/P): p
Usia            : 54
-----
Data Dosen ke-2
Kode Dosen      : 920
Nama Dosen      : putra
Jenis Kelamin (L/P): L
Usia            : 43
-----
```

```
===== MENU =====
1. Tambah Data Dosen
2. Tampilkan Data Dosen
3. Cari Data Dosen (Nama - Sequential Search)
4. Cari Data Dosen (Usia - Binary Search)
5. Keluar
Pilih Menu: 2
Kode            : 99
Nama            : kaia
Jenis Kelamin   : Perempuan
Usia            : 54
-----
Kode            : 920
Nama            : putra
Jenis Kelamin   : Laki-laki
Usia            : 43
-----
===== MENU =====
1. Tambah Data Dosen
2. Tampilkan Data Dosen
3. Cari Data Dosen (Nama - Sequential Search)
4. Cari Data Dosen (Usia - Binary Search)
5. Keluar
Pilih Menu: 3
Masukkan nama dosen yang dicari: kaia
Kode            : 99
Nama            : kaia
Jenis Kelamin   : Perempuan
Usia            : 54
-----
```

```
===== MENU =====
1. Tambah Data Dosen
2. Tampilkan Data Dosen
3. Cari Data Dosen (Nama - Sequential Search)
4. Cari Data Dosen (Usia - Binary Search)
5. Keluar
Pilih Menu: 4
Masukkan usia dosen yang dicari: 54
Ditemukan 1 data dengan usia 54:
Kode           : 99
Nama           : kaia
Jenis Kelamin  : Perempuan
Usia           : 54
-----

===== MENU =====
1. Tambah Data Dosen
2. Tampilkan Data Dosen
3. Cari Data Dosen (Nama - Sequential Search)
4. Cari Data Dosen (Usia - Binary Search)
5. Keluar
Pilih Menu: 5
Keluar dari program
```