

MODUL 4 LAPORAN

PRAKTIKUM STRUKTUR DATA



Disusun oleh :

Nama : Fidelia Ping

NIM : 245410012

Kelas : Informatika 1

PROGRAM STUDI INFORMATIKA
PROGRAM SARJANA
FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS TEKNOLOGI DIGITAL INDONESIA
YOGYAKARTA

2025

MODUL 4

PENGELOLAAN DATA PADA ARRAY/LARIK: PENAMBAHAN DAN PENGHAPUSAN DATA

A. TUJUAN PRAKTIKUM

Mahasiswa dapat melakukan pencarian terhadap suatu data yang terdapat di dalam larik

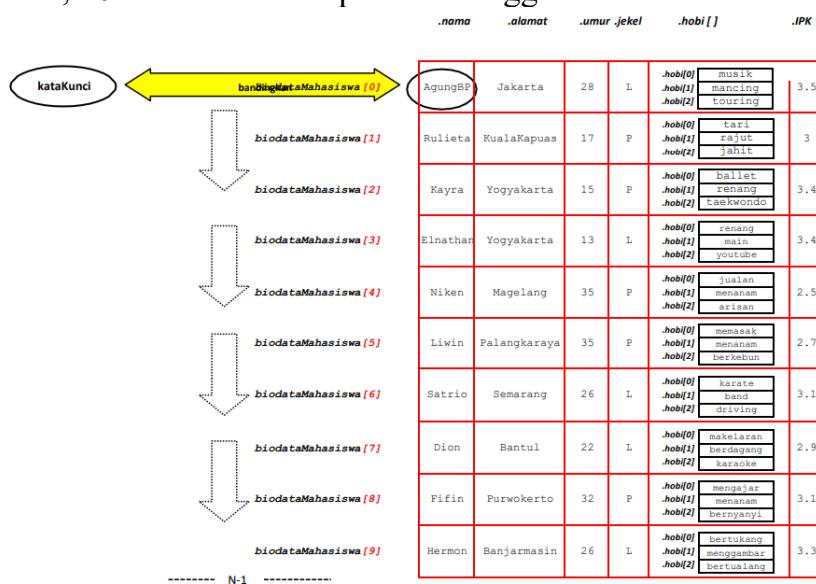
B. DASAR TEORI

Searching (Pencarian) adalah suatu metode pencarian sebuah data dalam sekumpulan besar data. Jika data yang dicari ditemukan maka program harus dapat memberikan informasi bahwa data yang dicari ditemukan, namun jika tidak berhasil ditemukan, program harus memberikan informasi bahwa data tersebut tidak ada didalam kumpulan data yang bersangkutan.

Ada beberapa metode pencarian data yaitu :

a. Linear Search (Pencarian Linear) atau Sequential Search (Pencarian Sekuensial)

Linier Search atau Sequential Search adalah metode pencarian yang membandingkan kata kunci (data yang dicari) dengan setiap data yang ada dalam larik, 40 mulai dari data pertama hingga data terakhir



Gambar 4.1 Skema Pencarian Sekuensial

b. Binary Search (Pencarian Biner)

Binary Search adalah pencarian terhadap data yang sudah terurut. Data kunci dibandingkan dengan target yaitu data yang berada di tengah larik (data pivot). Jika kata kunci sama dengan target maka data ditemukan dan pencarian dihentikan.

	.nama	.alamat	.umur	.jekel	.hobi[]	.IPK
biodataMahasiswa [0]	AgungBP	Jakarta	28	L	mancing mancing touring	3,5
biodataMahasiswa [1]	Rulieta	KualaKapuas	17	P	lari lari jantung	3
biodataMahasiswa [2]	Kayra	Yogyakarta	15	P	berlatih berlatih taekwondo	3,4
biodataMahasiswa [3]	Elnathar	Yogyakarta	13	L	olahraga olahraga volleyball	3,4
biodataMahasiswa [4]	Niken	Surabaya	35	P	hendingkan anak hendingkan anak bersepeda	3,5
biodataMahasiswa [5]	Liwin	Palangkaraya	35	P	menari menari bersepeda	2,7
biodataMahasiswa [6]	Satrio	Semarang	26	L	bermain bermain driving	3,1
biodataMahasiswa [7]	Dion	Bantul	22	L	nakalitan nakalitan karaoke	2,9
biodataMahasiswa [8]	Fifin	Purwokerto	32	P	menari menari bernyanyi	3,1
biodataMahasiswa [9]	Hermon	Banjarmasin	26	L	berrukung berrukung berolahraga	3,3

C. PEMBAHASAN LISTING

PRAKTIKUM

Praktik 1

```

import java.util.Scanner;

class formatBiodata {
    String nama;
    String alamat;
    int umur;
    char jekel;
    String hobi[] = new String[3];
    float ipk;
}

public class Program4_1 {
    public static int N = 0;

    // --- Fungsi Menambah Data di Belakang ---
    public static void tambahDataDiBelakang(formatBiodata biodataMahasiswa[]) {
        formatBiodata biodataMahasiswaBaru = new formatBiodata();
        Scanner masukan = new Scanner(System.in);
        int bacaTombol = 0;

        System.out.print("Silakan masukkan nama anda : ");
        biodataMahasiswaBaru.nama = masukan.next();
        System.out.print("Silakan masukkan alamat anda : ");
        biodataMahasiswaBaru.alamat = masukan.next();
        System.out.print("Silakan masukkan umur anda : ");
        biodataMahasiswaBaru.umur = masukan.nextInt();
        System.out.print("Silakan masukkan Jenis Kelamin anda : ");
        try {
            bacaTombol = System.in.read();
        } catch (java.io.IOException e) {}
        biodataMahasiswaBaru.jekel = (char) bacaTombol;
        System.out.println("Silakan masukkan hobi (maks 3) : ");
        System.out.print("hobi ke-0 : ");
    }
}

```

```

biodataMahasiswaBaru.hobi[0] = masukan.next();
System.out.print("hobi ke-1 : ");
biodataMahasiswaBaru.hobi[1] = masukan.next();
System.out.print("hobi ke-2 : ");
biodataMahasiswaBaru.hobi[2] = masukan.next();
System.out.print("Silakan masukkan IPK anda : ");
biodataMahasiswaBaru.ipk = masukan.nextFloat();

biodataMahasiswa[N] = biodataMahasiswaBaru;
N++;
}

// --- Fungsi Menampilkan Data ---
public static void tampilanData(formatBiodata biodataMahasiswa[]) {

System.out.println("\nNAMA\tALAMAT\tUMUR\tJEKEL\tHobi[0]\tHobi[1]\tHobi[2]\tIPK");
for (int i = 0; i < N; i++) {
    System.out.print(i + ". ");
    System.out.print(biodataMahasiswa[i].nama + "\t");
    System.out.print(biodataMahasiswa[i].alamat + "\t");
    System.out.print(biodataMahasiswa[i].umur + "\t");
    System.out.print(biodataMahasiswa[i].jekel + "\t");
    System.out.print(biodataMahasiswa[i].hobi[0] + "\t");
    System.out.print(biodataMahasiswa[i].hobi[1] + "\t");
    System.out.print(biodataMahasiswa[i].hobi[2] + "\t");
    System.out.println(biodataMahasiswa[i].ipk);
}
}

// --- Fungsi Pencarian Linear ---
public static void cariDataLinear(formatBiodata biodataMahasiswa[]) {
    Scanner masukan = new Scanner(System.in);
    System.out.print("\nSilakan masukkan nama yang ingin dicari: ");
    String kataKunci = masukan.next();
    boolean statusKetemu = false;
    int lokasiKetemu = -1;

    int i = 0;
    while ((i <= N - 1) && (statusKetemu == false)) {
        if (kataKunci.equals(biodataMahasiswa[i].nama)) {
            statusKetemu = true;
            lokasiKetemu = i;
        }
        i++;
    }

    if (statusKetemu) {
        System.out.println("Data dengan nama '" + kataKunci + "' ditemukan di posisi
ke- " + lokasiKetemu);
    } else {
        System.out.println("Data dengan nama '" + kataKunci + "' TIDAK ditemukan
dalam daftar.");
    }
}

```

```
}

// --- Program Utama ---
public static void main(String[] args) {
    formatBiodata biodataMahasiswa[] = new formatBiodata[10];
    for (int i = 0; i < biodataMahasiswa.length; i++) {
        biodataMahasiswa[i] = new formatBiodata();
    }

    Scanner masukan = new Scanner(System.in);

    // Isi awal 2 data
    for (int i = 0; i < 2; i++) {
        System.out.println("\nMasukkan data awal ke-" + (i + 1));
        System.out.print("Nama : ");
        biodataMahasiswa[i].nama = masukan.next();
        System.out.print("Alamat : ");
        biodataMahasiswa[i].alamat = masukan.next();
        System.out.print("Umur : ");
        biodataMahasiswa[i].umur = masukan.nextInt();
        System.out.print("Jenis Kelamin : ");
        biodataMahasiswa[i].jekel = masukan.next().charAt(0);
        System.out.print("Hobi ke-0 : ");
        biodataMahasiswa[i].hobi[0] = masukan.next();
        System.out.print("Hobi ke-1 : ");
        biodataMahasiswa[i].hobi[1] = masukan.next();
        System.out.print("Hobi ke-2 : ");
        biodataMahasiswa[i].hobi[2] = masukan.next();
        System.out.print("IPK : ");
        biodataMahasiswa[i].ipk = masukan.nextFloat();
        N++;
    }

    System.out.println("\n==== Sebelum Penambahan Data ====");
    tampilanData(biodataMahasiswa);

    // Tambah data di belakang
    tambahDataDiBelakang(biodataMahasiswa);

    System.out.println("\n==== Setelah Penambahan Data ====");
    tampilanData(biodataMahasiswa);

    // Pencarian data
    System.out.println("\n==== PENCARIAN DATA MAHASISWA ====");
    cariDataLinear(biodataMahasiswa);
}

}
```

```

Masukkan data awal ke-1
Nama : Fidelia
Alamat : Yogyakarta
Umur : 90
Jenis Kelamin : P
Hobi ke-0 : o
Hobi ke-1 : p
Hobi ke-2 : q
IPK : 10.0

Masukkan data awal ke-2
Nama : yogi
Alamat : dihati
Umur : 88
Jenis Kelamin : boti
Hobi ke-0 : d
Hobi ke-1 : g
Hobi ke-2 : f
IPK : 2.0

```

Pembahasan : Program di atas merupakan program Java yang berfungsi untuk mengelola data biodata mahasiswa menggunakan array objek dengan operasi **penambahan data, penampilan data, dan pencarian data menggunakan metode Linear Search.** Program diawali dengan kelas `formatBiodata` yang menyimpan atribut setiap mahasiswa seperti nama, alamat, umur, jenis kelamin, tiga hobi, dan IPK. Dalam kelas utama `Program4_1`, terdapat variabel `N` untuk menghitung jumlah data yang tersimpan. Pada fungsi `tambahDataDiBelakang()`, pengguna dapat memasukkan data mahasiswa baru yang akan disimpan di posisi terakhir array. Fungsi `tampilkanData()` digunakan untuk menampilkan seluruh data mahasiswa dalam bentuk tabel yang rapi di konsol. Selanjutnya, fungsi `cariDataLinear()` memungkinkan pengguna mencari data mahasiswa berdasarkan nama dengan melakukan pencarian berurutan (linear) dari indeks pertama hingga terakhir; jika nama ditemukan, program menampilkan posisi data, dan jika tidak ditemukan, menampilkan pesan bahwa data tidak ada. Dalam `main()`, program diawali dengan mengisi dua data awal mahasiswa secara manual melalui input pengguna, kemudian menampilkan data sebelum dan sesudah penambahan data baru, serta melakukan pencarian berdasarkan nama yang dimasukkan. Secara keseluruhan, program ini memperlihatkan bagaimana konsep **array objek, input pengguna, fungsi modular, dan pencarian linear** diterapkan dalam pengolahan data sederhana berbasis teks.

Praktik 2

```

import java.util.Scanner;

class formatBiodata
{
    String nama;
    String alamat;
    int umur;
    char jekel;
    String hobi[] = new String[3];

```

```

        float ipk;
    }

public class Program3_2
{
    public static int N = 0;

    //
    //--- Fungsi untuk Menambah Data Di Tengah ---
    //
    public static void tambahDataDiTengah(formatBiodata biodataMahasiswa[])
    {
        //bagian membuat record sementara untuk menampung data baru-----
        formatBiodata biodataMahasiswaBaru = new formatBiodata();
        //bagian entri data baru ke penyimpan sementara-----
        Scanner masukan = new Scanner(System.in);
        int bacaTombol = 0;
        System.out.print("Silakan masukkan nama anda : ");
        biodataMahasiswaBaru.nama = masukan.next();
        System.out.print("Silakan masukkan alamat anda : ");
        biodataMahasiswaBaru.alamat = masukan.next();
        System.out.print("Silakan masukkan umur anda : ");
        biodataMahasiswaBaru.umur = masukan.nextInt();
        System.out.print("Silakan masukkan Jenis Kelamin anda : ");
        try { bacaTombol = System.in.read(); }
        catch (java.io.IOException e) { }
        biodataMahasiswaBaru.jekel = (char)bacaTombol;
        System.out.println("Silakan masukkan hobi (maks 3) : ");
        System.out.print("hobi ke-0 : ");
        biodataMahasiswaBaru.hobi[0] = masukan.next();
        System.out.print("hobi ke-1 : ");
        biodataMahasiswaBaru.hobi[1] = masukan.next();
        System.out.print("hobi ke-2 : ");
        biodataMahasiswaBaru.hobi[2] = masukan.next();
        System.out.print("Silakan masukkan IPK anda : ");
        biodataMahasiswaBaru.ipk = masukan.nextFloat();
        //bagian menentukan posisi target T -----
        int T;
        System.out.print("Pada posisi ke berapa data akan dimasukkan ? : ");
        T = masukan.nextInt();
        //bagian menggeser isi larik mulai dari Belakang s/d T selangkah ke belakang
        for (int i = N - 1; i >= T; i--)
        { biodataMahasiswa[i + 1] = biodataMahasiswa[i]; }
        //bagian memindahkan data baru ke larik ke-T-----
        biodataMahasiswa[T] = biodataMahasiswaBaru;
        //memperbarui banyaknya data (N), banyaknya data bertambah satu-----
        N++;
    }

    public static void tampilkanData(formatBiodata biodataMahasiswa[])
    {
        System.out.println("\nNAMA ALAMAT UMUR JEKEL HOBI[0] HOBI[1] HOBI[2] IPK");
        for (int i = 0; i < N; i++)

```

```

    {
        System.out.print(i + ". ");
        System.out.print(biodataMahasiswa[i].nama + "\t");
        System.out.print(biodataMahasiswa[i].alamat + "\t");
        System.out.print(biodataMahasiswa[i].umur + "\t");
        System.out.print(biodataMahasiswa[i].jekel + "\t");
        System.out.print(biodataMahasiswa[i].hobi[0] + "\t");
        System.out.print(biodataMahasiswa[i].hobi[1] + "\t");
        System.out.print(biodataMahasiswa[i].hobi[2] + "\t");
        System.out.println(biodataMahasiswa[i].ipk);
    }
}

public static void main(String[] args)
{
    // Buat array dan isi awal
    formatBiodata biodataMahasiswa[] = new formatBiodata[10];
    for (int i = 0; i < 10; i++) {
        biodataMahasiswa[i] = new formatBiodata();
    }

    // Tambahkan 2 data awal (supaya bisa lihat posisi tengah)
    Scanner masukan = new Scanner(System.in);
    for (int i = 0; i < 2; i++) {
        System.out.println("\nMasukkan data awal ke-" + (i + 1));
        System.out.print("Nama : ");
        biodataMahasiswa[i].nama = masukan.next();
        System.out.print("Alamat : ");
        biodataMahasiswa[i].alamat = masukan.next();
        System.out.print("Umur : ");
        biodataMahasiswa[i].umur = masukan.nextInt();
        System.out.print("Jenis Kelamin : ");
        biodataMahasiswa[i].jekel = masukan.next().charAt(0);
        System.out.print("Hobi ke-0 : ");
        biodataMahasiswa[i].hobi[0] = masukan.next();
        System.out.print("Hobi ke-1 : ");
        biodataMahasiswa[i].hobi[1] = masukan.next();
        System.out.print("Hobi ke-2 : ");
        biodataMahasiswa[i].hobi[2] = masukan.next();
        System.out.print("IPK : ");
        biodataMahasiswa[i].ipk = masukan.nextFloat();
        N++;
    }

    System.out.println("\n==== Sebelum Penambahan Data ====");
    tampilkanData(biodataMahasiswa);

    tambahDataDiTengah(biodataMahasiswa);

    System.out.println("\n==== Setelah Penambahan Data ====");
    tampilkanData(biodataMahasiswa);
}
}

```

```

Masukkan data awal ke-1
Nama : piping
Alamat : yogyakarta
Umur : 19
Jenis Kelamin : p
Hobi ke-0 : manut
Hobi ke-1 : eeeeeee
Hobi ke-2 : gatau
IPK : 4.0

Masukkan data awal ke-2
Nama : fidelia
Alamat : dihatimu
Umur : 15
Jenis Kelamin : p
Hobi ke-0 : gatau
Hobi ke-1 : saya
Hobi ke-2 : juga
IPK : 5.0

==== Sebelum Penambahan Data ===

NAMA ALAMAT UMUR JEKEL HOBI[0] HOBI[1] HOBI[2] IPK
0. piping      yogyakarta    19      p      manut    eeeeeee   gatau    4.0
1. fidelia    dihatimu     15      p      gatau    saya     juga     5.0
Silakan masukkan nama anda :

```

Pembahasan : Program di atas merupakan program Java yang digunakan untuk menambah data mahasiswa di posisi tengah dalam sebuah array objek. Kelas `formatBiodata` digunakan untuk mendefinisikan struktur data mahasiswa yang berisi atribut seperti `nama`, `alamat`, `umur`, `jekek` (jenis kelamin), tiga elemen `hobi`, dan `ipk`. Dalam kelas utama `Program3_2`, variabel `N` berfungsi menyimpan jumlah data yang sudah terisi dalam array `biodataMahasiswa`. Pada awal program, pengguna diminta mengisi dua data mahasiswa pertama agar array memiliki isi awal. Fungsi `tampilkanData()` digunakan untuk menampilkan seluruh data mahasiswa dalam format tabel sederhana di konsol. Fungsi utama dari program ini, yaitu `tambahDataDiTengah()`, memungkinkan pengguna menambahkan satu data baru ke posisi tertentu (misalnya di tengah array). Prosesnya dilakukan dengan membuat objek baru `biodataMahasiswaBaru`, meminta pengguna mengisi seluruh data, lalu menentukan posisi target `T` tempat data baru akan disisipkan. Setelah itu, seluruh elemen array mulai dari indeks terakhir hingga posisi `T` digeser satu langkah ke belakang agar posisi `T` menjadi kosong, kemudian data baru dimasukkan ke posisi tersebut, dan variabel `N` ditambah satu. Hasil akhir kemudian ditampilkan kembali untuk menunjukkan bahwa data baru berhasil dimasukkan di posisi yang diinginkan. Program ini menggambarkan penerapan konsep **manipulasi array**, **input data dinamis**, dan **operasi penyisipan data di tengah array menggunakan pergeseran elemen dalam pemrograman Java**.

Praktik 3

```
import java.util.Scanner;

class formatBiodata {
    String nama;
    String alamat;
    int umur;
    char jekel;
    String hobi[] = new String[3];
    float ipk;
}

public class Program4_2 {
    public static int N = 0;

    public static void tambahDataDiBelakang(formatBiodata biodataMahasiswa[]) {
        formatBiodata biodataMahasiswaBaru = new formatBiodata();
        Scanner masukan = new Scanner(System.in);
        int bacaTombol = 0;
        System.out.print("Silakan masukkan nama anda : ");
        biodataMahasiswaBaru.nama = masukan.next();
        System.out.print("Silakan masukkan alamat anda : ");
        biodataMahasiswaBaru.alamat = masukan.next();
        System.out.print("Silakan masukkan umur anda : ");
        biodataMahasiswaBaru.umur = masukan.nextInt();
        System.out.print("Silakan masukkan Jenis Kelamin anda : ");
        try { bacaTombol = System.in.read(); }
        catch (java.io.IOException e) {}
        biodataMahasiswaBaru.jekel = (char)bacaTombol;
        System.out.println("Silakan masukkan hobi (maks 3) : ");
        System.out.print("hobi ke-0 : ");
        biodataMahasiswaBaru.hobi[0] = masukan.next();
        System.out.print("hobi ke-1 : ");
        biodataMahasiswaBaru.hobi[1] = masukan.next();
        System.out.print("hobi ke-2 : ");
        biodataMahasiswaBaru.hobi[2] = masukan.next();
        System.out.print("Silakan masukkan IPK anda : ");
        biodataMahasiswaBaru.ipk = masukan.nextFloat();
        biodataMahasiswa[N] = biodataMahasiswaBaru;
        N++;
    }

    public static void tampilkanData(formatBiodata biodataMahasiswa[]) {

        System.out.println("\nNAMA\tALAMAT\tUMUR\tJEKEL\tHABI[0]\tHABI[1]\tHABI[2]\tIPK");
        for (int i = 0; i < N; i++) {
            System.out.print(i + ". ");
            System.out.print(biodataMahasiswa[i].nama + "\t");
            System.out.print(biodataMahasiswa[i].alamat + "\t");
            System.out.print(biodataMahasiswa[i].umur + "\t");
            System.out.print(biodataMahasiswa[i].jekel + "\t");
            System.out.print(biodataMahasiswa[i].hobi[0] + "\t");
            System.out.print(biodataMahasiswa[i].hobi[1] + "\t");
        }
    }
}
```

```

        System.out.print(biodataMahasiswa[i].hobi[2] + "\t");
        System.out.println(biodataMahasiswa[i].ipk);
    }
}

public static void mencariDataBiner(formatBiodata biodataMahasiswa[]) {
    String kataKunci;
    int lokasi = -1;
    boolean statusKetemu = false;
    Scanner masukan = new Scanner(System.in);
    System.out.print("Masukkan kata kunci pencarian : ");
    kataKunci = masukan.next();
    int atas, bawah, tengah;
    atas = 0;
    bawah = N - 1;
    tengah = (atas + bawah) / 2;
    while ((statusKetemu == false) && (bawah - atas != 1)) {
        System.out.println(biodataMahasiswa[tengah].nama + " <--> " + kataKunci);
        if (kataKunci.compareTo(biodataMahasiswa[tengah].nama) < 0) {
            bawah = tengah;
            tengah = (atas + bawah) / 2;
        } else if (kataKunci.compareTo(biodataMahasiswa[tengah].nama) > 0) {
            atas = tengah;
            tengah = (atas + bawah) / 2;
        } else {
            statusKetemu = true;
            lokasi = tengah;
        }
    }
    if (statusKetemu == true) {
        System.out.println("Data yang anda cari KETEMU di larik ke :" + lokasi);
    } else {
        System.out.println("Maaf, nama yang anda cari tidak ditemukan ");
    }
}

public static void main(String[] args) {
    formatBiodata biodataMahasiswa[] = new formatBiodata[10];
    for (int i = 0; i < biodataMahasiswa.length; i++) {
        biodataMahasiswa[i] = new formatBiodata();
    }
    Scanner masukan = new Scanner(System.in);
    for (int i = 0; i < 2; i++) {
        System.out.println("\nMasukkan data awal ke-" + (i + 1));
        System.out.print("Nama : ");
        biodataMahasiswa[i].nama = masukan.next();
        System.out.print("Alamat : ");
        biodataMahasiswa[i].alamat = masukan.next();
        System.out.print("Umur : ");
        biodataMahasiswa[i].umur = masukan.nextInt();
        System.out.print("Jenis Kelamin : ");
        biodataMahasiswa[i].jekel = masukan.next().charAt(0);
        System.out.print("Hobi ke-0 : ");
    }
}

```

```
        biodataMahasiswa[i].hobi[0] = masukan.next();
        System.out.print("Hobi ke-1 : ");
        biodataMahasiswa[i].hobi[1] = masukan.next();
        System.out.print("Hobi ke-2 : ");
        biodataMahasiswa[i].hobi[2] = masukan.next();
        System.out.print("IPK : ");
        biodataMahasiswa[i].ipk = masukan.nextFloat();
        N++;
    }

    System.out.println("\n==== Sebelum Penambahan Data ====");
    tampilanData(biodataMahasiswa);
    tambahDataDiBelakang(biodataMahasiswa);
    System.out.println("\n==== Setelah Penambahan Data ====");
    tampilanData(biodataMahasiswa);

    System.out.println("\n==== Pencarian Data (Binary Search) ====");
    mencariDataBiner(biodataMahasiswa);
}
}
```

Masukkan data awal ke-1
Nama : ping
Alamat : kaltim
Umur : 19
Jenis Kelamin : P
Hobi ke-0 : d
Hobi ke-1 : f
Hobi ke-2 : g
IPK : 4.0

Masukkan data awal ke-2
Nama : rian
Alamat : klaten
Umur : 99
Jenis Kelamin : L
Hobi ke-0 : G
Hobi ke-1 : H
Hobi ke-2 : j
IPK : 99.9

```

==== Sebelum Penambahan Data ====
NAMA    ALAMAT   UMUR    JEKEL    HOBI[0]  HOBI[1]  HOBI[2]  IPK
0. ping kaltim 19      P        d        f        g        4.0
1. rian klaten 99      L        G        H        j        99.9
Silakan masukkan nama anda : yogi
Silakan masukkan alamat anda : bantul
Silakan masukkan umur anda : 88
Silakan masukkan Jenis Kelamin anda : B
Silakan masukkan hobi (maks 3) :
hobi ke-0 : d
hobi ke-1 : f
hobi ke-2 : g
Silakan masukkan IPK anda : 0.0

==== Setelah Penambahan Data ====
NAMA    ALAMAT   UMUR    JEKEL    HOBI[0]  HOBI[1]  HOBI[2]  IPK
0. ping kaltim 19      P        d        f        g        4.0
1. rian klaten 99      L        G        H        j        99.9
2. yogi bantul 88      B        d        f        g        0.0

==== Pencarian Data (Binary Search) ====
Masukkan kata kunci pencarian : ping
rian <--> ping
Maaf, nama yang anda cari tidak ditemukan
Press any key to continue . . .

```

Pembahasan : Program di atas merupakan program Java yang digunakan untuk menyimpan, menampilkan, menambah, dan mencari data mahasiswa menggunakan algoritma pencarian biner (Binary Search). Kelas `formatBiodata` berfungsi sebagai struktur data untuk menyimpan atribut setiap mahasiswa seperti nama, alamat, umur, jekel (jenis kelamin), tiga hobi, dan ipk. Dalam kelas utama `Program4_2`, variabel statis `N` digunakan untuk mencatat jumlah data yang telah dimasukkan ke dalam array `biodataMahasiswa`. Pada awal program, pengguna diminta mengisi dua data awal mahasiswa secara manual. Kemudian, fungsi `tambahDataDiBelakang()` memungkinkan pengguna menambahkan satu data baru di posisi paling belakang array dengan cara membuat objek baru `biodataMahasiswaBaru` dan mengisinya berdasarkan input pengguna. Fungsi `tampilkanData()` digunakan untuk menampilkan seluruh data mahasiswa dalam bentuk tabel sederhana di konsol. Selanjutnya, program menjalankan fungsi `mencariDataBiner()`, yaitu implementasi Binary Search untuk mencari nama mahasiswa berdasarkan kata kunci yang dimasukkan. Algoritma ini bekerja dengan menentukan batas atas (bawah), batas bawah (atas), dan posisi tengah (tengah), kemudian membandingkan kata kunci dengan nama mahasiswa di posisi tengah menggunakan metode `compareTo()`. Jika hasil pencocokan belum ditemukan, batas pencarian akan disesuaikan hingga data ditemukan atau pencarian berhenti. Apabila

nama ditemukan, program menampilkan posisi (indeks) data tersebut di array; jika tidak, ditampilkan pesan bahwa nama tidak ditemukan. Program ini menggabungkan konsep **array objek, input data dinamis, penambahan data, serta pencarian cepat menggunakan algoritma Binary Search**.

LATIHAN

```
import java.util.Scanner;

class formatBiodata {
    String nama;
    String alamat;
    int umur;
    char jekel;
    String hobi[] = new String[3];
    float ipk;
}

//LATIHAN
//--- Fungsi untuk Menampilkan Data Berdasarkan Syarat (Jenis Kelamin L) ---
public class LATIHAN {
    public static int N = 0;

    public static void tampilanDataLakiLaki(formatBiodata biodataMahasiswa[]) {
        System.out.println("\n==== Data Mahasiswa Berjenis Kelamin L ===");
        boolean adaData = false;

        for (int i = 0; i < N; i++) {
            if (biodataMahasiswa[i] != null && biodataMahasiswa[i].jekel == 'L') {
                adaData = true;
                System.out.println("Nama : " + biodataMahasiswa[i].nama);
                System.out.println("Alamat : " +
biodataMahasiswa[i].alamat);
                System.out.println("Jenis Kelamin : " + biodataMahasiswa[i].jekel);
                System.out.println("Hobi : "
                    + biodataMahasiswa[i].hobi[0] + ", "
                    + biodataMahasiswa[i].hobi[1] + ", "
                    + biodataMahasiswa[i].hobi[2]);
                System.out.println("IPK : " + biodataMahasiswa[i].ipk);
                System.out.println("-----");
            }
        }

        if (!adaData) {
            System.out.println("Tidak ada data mahasiswa dengan jenis kelamin L.");
        }
    }

    // =====
    // Bagian tambahan agar bisa dijalankan dan input manual
    // =====
    public static void main(String[] args) {
```

```
Scanner masukan = new Scanner(System.in);
formatBiodata[] biodataMahasiswa = new formatBiodata[10];

System.out.print("Masukkan jumlah mahasiswa: ");
N = masukan.nextInt();

for (int i = 0; i < N; i++) {
    biodataMahasiswa[i] = new formatBiodata();
    System.out.println("\nMasukkan data mahasiswa ke-" + (i + 1));
    System.out.print("Nama: ");
    biodataMahasiswa[i].nama = masukan.next();
    System.out.print("Alamat: ");
    biodataMahasiswa[i].alamat = masukan.next();
    System.out.print("Umur: ");
    biodataMahasiswa[i].umur = masukan.nextInt();
    System.out.print("Jenis Kelamin (L/P): ");
    biodataMahasiswa[i].jekel = masukan.next().charAt(0);
    System.out.print("Hobi ke-1: ");
    biodataMahasiswa[i].hobi[0] = masukan.next();
    System.out.print("Hobi ke-2: ");
    biodataMahasiswa[i].hobi[1] = masukan.next();
    System.out.print("Hobi ke-3: ");
    biodataMahasiswa[i].hobi[2] = masukan.next();
    System.out.print("IPK: ");
    biodataMahasiswa[i].ipk = masukan.nextFloat();
}

tampilkanDataLakiLaki(biodataMahasiswa);
}
```

```
Masukkan jumlah mahasiswa: 2

Masukkan data mahasiswa ke-1
Nama: Piping
Alamat: d
Umur: 3
Jenis Kelamin (L/P): d
Hobi ke-1: d
Hobi ke-2: f
Hobi ke-3: g
IPK: 3

Masukkan data mahasiswa ke-2
Nama: yoga
Alamat: d
Umur: 3
Jenis Kelamin (L/P): L
Hobi ke-1: e
Hobi ke-2: r
Hobi ke-3: t
IPK: 5

==== Data Mahasiswa Berjenis Kelamin L ====
Nama : yoga
Alamat : d
Jenis Kelamin : L
Hobi : e, r, t
IPK : 5.0
-----
Press any key to continue . . .
```

Pembahasan : Program di atas merupakan program Java yang digunakan untuk menampilkan data mahasiswa yang berjenis kelamin laki-laki (L) dari sekumpulan data biodata mahasiswa yang diinput secara manual oleh pengguna. Program ini terdiri dari kelas `formatBiodata` yang berfungsi sebagai wadah (template) untuk menyimpan informasi setiap mahasiswa seperti nama, alamat, umur, jenis kelamin (jekel), tiga hobi, dan IPK. Dalam kelas utama `LATIHAN`, terdapat fungsi `tampilkanDataLakiLaki()` yang bertugas menelusuri seluruh array `biodataMahasiswa` menggunakan perulangan `for` hingga indeks terakhir (berdasarkan nilai N). Setiap elemen dicek apakah objek biodata tidak `null` dan apakah `jekel` bernilai 'L'; jika ya, maka semua datanya akan ditampilkan di layar secara terformat. Jika tidak ada satu pun mahasiswa berjenis kelamin L, program akan menampilkan pesan bahwa data tidak ditemukan. Pada fungsi `main()`, pengguna diminta menginput jumlah mahasiswa, lalu mengisi data biodata masing-masing (nama, alamat, umur, jenis kelamin, hobi, dan IPK). Setelah semua data dimasukkan, fungsi `tampilkanDataLakiLaki()` dipanggil untuk menampilkan hanya mahasiswa laki-laki, sedangkan mahasiswa perempuan tidak akan muncul dalam output.

D. PEMBAHASAN TUGAS

Tugas 1

```
import java.util.Scanner;

class formatBiodata {
    String nama;
    String alamat;
    int umur;
    char jekel;
    String hobi[] = new String[3];
    float ipk;
}

public class Tugas {
    public static int N = 0;

    // --- Fungsi Menampilkan Data ---
    public static void tampilanData(formatBiodata biodataMahasiswa[]) {

        System.out.println("\nNAMA\tALAMAT\tUMUR\tJEKEL\tHABI[0]\tHABI[1]\tHABI[2]\tIPK");
        for (int i = 0; i < N; i++) {
            System.out.print(i + ". ");
            System.out.print(biodataMahasiswa[i].nama + "\t");
            System.out.print(biodataMahasiswa[i].alamat + "\t");
            System.out.print(biodataMahasiswa[i].umur + "\t");
            System.out.print(biodataMahasiswa[i].jekel + "\t");
            System.out.print(biodataMahasiswa[i].hobi[0] + "\t");
            System.out.print(biodataMahasiswa[i].hobi[1] + "\t");
            System.out.print(biodataMahasiswa[i].hobi[2] + "\t");
            System.out.println(biodataMahasiswa[i].ipk);
        }
    }

    // --- Fungsi Menghapus Data Berdasarkan Nama (Linear Search) ---
    public static void hapusData(formatBiodata biodataMahasiswa[]) {
        Scanner masukan = new Scanner(System.in);
        System.out.print("\nMasukkan nama yang ingin dihapus: ");
        String namaHapus = masukan.next();

        boolean ditemukan = false;
        int posisi = -1;

        // Pencarian Linear
        for (int i = 0; i < N; i++) {
            if (namaHapus.equals(biodataMahasiswa[i].nama)) {
                ditemukan = true;
                posisi = i;
                break;
            }
        }

        if (ditemukan) {
            // Geser data ke kiri mulai dari posisi yang dihapus
            for (int i = posisi; i < N - 1; i++) {
                biodataMahasiswa[i] = biodataMahasiswa[i + 1];
            }
        }
    }
}
```

```

    }
    N--;
    System.out.println("Data dengan nama '" + namaHapus + "' berhasil
dihapus!");
} else {
    System.out.println("Data dengan nama '" + namaHapus + "' TIDAK ditemukan.
Penghapusan dibatalkan.");
}
}

// --- Program Utama ---
public static void main(String[] args) {
    formatBiodata biodataMahasiswa[] = new formatBiodata[10];
    for (int i = 0; i < biodataMahasiswa.length; i++) {
        biodataMahasiswa[i] = new formatBiodata();
    }

    Scanner masukan = new Scanner(System.in);

    // Isi awal 2 data
    for (int i = 0; i < 2; i++) {
        System.out.println("\nMasukkan data awal ke-" + (i + 1));
        System.out.print("Nama : ");
        biodataMahasiswa[i].nama = masukan.next();
        System.out.print("Alamat : ");
        biodataMahasiswa[i].alamat = masukan.next();
        System.out.print("Umur : ");
        biodataMahasiswa[i].umur = masukan.nextInt();
        System.out.print("Jenis Kelamin : ");
        biodataMahasiswa[i].jekel = masukan.next().charAt(0);
        System.out.print("Hobi ke-0 : ");
        biodataMahasiswa[i].hobi[0] = masukan.next();
        System.out.print("Hobi ke-1 : ");
        biodataMahasiswa[i].hobi[1] = masukan.next();
        System.out.print("Hobi ke-2 : ");
        biodataMahasiswa[i].hobi[2] = masukan.next();
        System.out.print("IPK : ");
        biodataMahasiswa[i].ipk = masukan.nextFloat();
        N++;
    }

    // Tampilkan data awal
    System.out.println("\n== DATA MAHASISWA ==");
    tampilanData(biodataMahasiswa);

    // Proses hapus data
    System.out.println("\n== PENGHAPUSAN DATA MAHASISWA ==");
    hapusData(biodataMahasiswa);

    // Tampilkan data setelah penghapusan
    System.out.println("\n== Setelah Penghapusan Data ==");
    tampilanData(biodataMahasiswa);
}
}

```

```

Masukkan data awal ke-1
Nama : ping
Alamat : kaltim
Umur : 15
Jenis Kelamin : P
Hobi ke-0 : a
Hobi ke-1 : s
Hobi ke-2 : d
IPK : 9.0

Masukkan data awal ke-2
Nama : ilham
Alamat : lombok
Umur : 100
Jenis Kelamin : L
Hobi ke-0 : a
Hobi ke-1 : s
Hobi ke-2 : d
IPK : 4.0
==== DATA MAHASISWA ====
NAMA    ALAMAT   UMUR    JEKEL    HOBI[0]  HOBI[1]  HOBI[2]  IPK
0. ping kaltim 15      P        a        s        d        9.0
1. ilham       lombok  100     L        a        s        d        4.0
==== PENGHAPUSAN DATA MAHASISWA ====
Masukkan nama yang ingin dihapus: ilham
Data dengan nama 'ilham' berhasil dihapus!
==== Setelah Penghapusan Data ====
NAMA    ALAMAT   UMUR    JEKEL    HOBI[0]  HOBI[1]  HOBI[2]  IPK
0. ping kaltim 15      P        a        s        d        9.0
Press any key to continue . . .

```

Pembahasan : Program di atas merupakan program Java yang berfungsi untuk **menghapus data mahasiswa berdasarkan nama menggunakan metode pencarian linear (Linear Search)**. Program diawali dengan kelas `formatBiodata` yang digunakan untuk menyimpan atribut setiap mahasiswa seperti nama, alamat, umur, jenis kelamin (jekel), tiga hobi, dan IPK. Dalam kelas utama `Tugas`, terdapat variabel global `N` sebagai penghitung jumlah data mahasiswa yang aktif. Fungsi `tampilkanData()` digunakan untuk menampilkan seluruh isi array mahasiswa dalam format tabel. Sementara itu, fungsi `hapusData()` bertugas mencari data mahasiswa berdasarkan nama yang dimasukkan pengguna. Proses pencarian dilakukan dengan **perulangan linear (for loop)** yang membandingkan satu per satu elemen `nama` di array hingga ditemukan kecocokan. Jika nama ditemukan, program akan menghapus data tersebut dengan cara **menggeser seluruh elemen setelahnya ke kiri** sehingga urutan data tetap rapat dan `N` dikurangi satu. Namun, jika nama tidak ditemukan, maka program akan menampilkan pesan bahwa penghapusan dibatalkan. Pada bagian `main()`,

pengguna diminta memasukkan dua data mahasiswa awal, lalu program menampilkan seluruh data, melakukan proses penghapusan berdasarkan nama yang dicari, dan akhirnya menampilkan kembali data mahasiswa setelah proses penghapusan dilakukan.

E. KESIMPULAN

Kesimpulan yang dapat saya ambil dari praktikum **“Pengelolaan Data pada Array/Larik: Penambahan dan Penghapusan Data”** adalah bahwa saya belajar bagaimana cara menyimpan, menambah, dan menghapus data secara terstruktur menggunakan array di dalam program Java. Melalui praktikum ini, saya memahami bahwa array memiliki ukuran tetap, sehingga proses penambahan atau penghapusan data harus dilakukan dengan cara manual, seperti menempatkan data baru pada indeks kosong atau menggeser elemen agar tidak ada celah setelah penghapusan. Saya juga mempelajari bagaimana menerapkan **pencarian linear (Linear Search)** untuk menemukan data tertentu sebelum dihapus, serta pentingnya menjaga konsistensi jumlah data dengan variabel penghitung (N). Dengan praktikum ini, saya menjadi lebih paham tentang logika dasar pengelolaan data dalam array dan bagaimana mengimplementasikannya secara efektif dalam program.