

**Laporan Hasil Praktikum Algoritma Dan Struktur Data**  
**Jobsheet 7**



Disusun Oleh :

Nama : Fadhil Taufiqurrachman  
NIM : 244107020090  
Kelas : Teknik Informatika 1E

**Program Studi Teknik Informatika**  
**Jurusan Teknologi Informasi**  
**Politeknik Negeri Malang 2025**

## 7.2 Percobaan 1 : Searching / Pencarian Menggunakan Algoritma Sequential Search

### 7.2.1 Kode Program

Kode program pada class Mahasiswa08 :

```
package Jobsheet7;

public class Mahasiswa08 {
    String nim, nama, kelas;
    double ipk;

    public Mahasiswa08() {

    }

    public Mahasiswa08(String nm, String name, String kls, double ip) {
        nim = nm;
        nama = name;
        kelas = kls;
        ipk = ip;
    }

    void tampilInformasi() {
        System.out.println("NIM      : " + nim);
        System.out.println("Nama    : " + nama);
        System.out.println("Kelas  : " + kelas);
        System.out.println("IPK     : " + ipk);
    }
}
```

### Kode program pada class MahasiswaBerprestasi08 :

```
package Jobsheet7;

public class MahasiswaBerprestasi08 {
    Mahasiswa08 listMhs[] = new Mahasiswa08[5];
    int idx;

    void tambah(Mahasiswa08 m) {
        if (idx < listMhs.length) {
            listMhs[idx] = m;
            idx++;
        } else {
            System.out.println("Data Sudah Penuh!");
        }
    }

    void tampil() {
        for (Mahasiswa08 m : listMhs) {
            m.tampilInformasi();
            System.out.println("-----");
        }
    }

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

    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");
        }
    }

    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");
        }
    }
}
```

### Kode program pada class MahasiswaDemo08 :

```
package Jobsheet7;

import java.util.Scanner;
public class MahasiswaDemo08 {
    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);
        int jumMhs = 5;
        MahasiswaBerprestasi08 list = new MahasiswaBerprestasi08();

        for (int i = 0; i < jumMhs; 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      : ");
            String ipk = input.nextLine();
            Double ipk = Double.parseDouble(ipk);
            System.out.println("-----");
            list.tambah(new Mahasiswa08(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 = input.nextDouble();

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

### 7.2.2 Verifikasi

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

```

NIM      : 111
Nama     : Adi
Kelas   : 2
IPK      : 3.6
-----
NIM      : 222
Nama     : Tio
Kelas   : 2
IPK      : 3.8
-----
NIM      : 333
Nama     : Ila
Kelas   : 2
IPK      : 3.0
-----
NIM      : 444
Nama     : Lia
Kelas   : 2
IPK      : 3.5
-----
NIM      : 555
Nama     : Fia
Kelas   : 2
IPK      : 3.3
-----
-----
Pencarian Data
-----
Masukkan IPK Mahasiswa Yang Dicari
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

```

### 7.2.3 Pertanyaan

1. Method `tampilPosisi` merupakan method untuk menampilkan hasil dari data yang dicari terdapat pada indeks berapa. -1 merupakan indeks jika data yang dicari tidak ditemukan.

Method `tampilDataSearch` merupakan method untuk menampilkan data yang dicari jika ditemukan. Jika tidak maka akan mencetak data tidak ditemukan.

2. Fungsi break pada kode perulangan di bawah ini adalah sebagai penghenti. Dalam artian jika kondisi tersebut terpenuhi atau data yang dicari ditemukan, maka akan menjalankan instruksi di dalam kondisi tersebut, setelah itu langsung menghentikan perulangan.

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

## 7.3 Percobaan 2 : Searching / Pencarian Menggunakan Binary Search

### 7.3.1 Kode Program

Menambahkan kode program pada class MahasiswaBerprestasi08 :

```
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;
}
```

Menambahkan kode program pada class MahasiswaDemo08 :

```
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);
```



### 7.3.2 Verifikasi

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

```

NIM      : 111
Nama     : Adi
Kelas   : 2
IPK      : 3.1
-----
NIM      : 222
Nama     : Ila
Kelas   : 2
IPK      : 3.2
-----
NIM      : 333
Nama     : Lia
Kelas   : 2
IPK      : 3.3
-----
NIM      : 444
Nama     : Susi
Kelas   : 2
IPK      : 3.5
-----
NIM      : 555
Nama     : Anita
Kelas   : 2
IPK      : 3.7
-----
-----
Pencarian Data
-----
Masukkan IPK Mahasiswa Yang Dicari
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

```

### 7.3.3 Pertanyaan

1. Proses Divide pada kode program terdapat pada pemecahan array menjadi dua bagian menggunakan indeks mid. Berikut di bawah ini adalah kode programnya.

```

if (right >= left) {
    mid = (left + right) / 2;
}

```

2. Proses Conquernya yakni terletak pada penyelesaian sub masalah secara rekursif setelah proses Divide. Berikut adalah kode programnya.

```
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);
}
```

3. Program akan tetap berjalan dan tidak terjadi error, namun nanti akan ada kemungkinan terdapat kesalahan pada letak indeks dan hasilnya.
4. Hasil dari Binary search tidak akan sesuai, tidak ditemukan. Ada dua cara untuk membenarkan kode program tersebut. Yang pertama yaitu dengan mengganti kondisi pada kode program berikut menjadi :

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

Yang awalnya lebih dari, menjadi kurang dari. Namun perlu diperhatikan bahwa kondisi ini hanya akan sesuai ketika data yang dimasukkan secara Descending. Ketika kita menginputkan data yang Ascending pada kode program tersebut, maka akan tidak sesuai hasilnya.

5. Memodifikasi kode program pada class MahasiswaDemo08, yang awalnya jumlah data tetap 5 menjadi dinamis sesuai input pengguna seperti berikut.

```
System.out.print("Masukkan Jumlah Data Mahasiswa : ");
int jumMhs = input.nextInt();
```

Kemudian memodifikasi dan menambahkan kode program pada class MahasiswaBerprestasi08 menjadi seperti berikut.

```
Mahasiswa08 listMhs[];
int idx;

void jumlahData(int jumlah) {
    Mahasiswa08 listMhs[] = new Mahasiswa08[jumlah];
    this.listMhs = listMhs;
}
```

## 7.4 Latihan Praktikum

### 7.4.1 Kode Program

Kode program pada class Dosen08 :

```
package Jobsheet7;

public class Dosen08 {
    String kode, nama;
    boolean jenisKelamin;
    int usia;

    Dosen08(String kd, String name, boolean jk, int age) {
        kode = kd;
        nama = name;
        jenisKelamin = jk;
        usia = age;
    }

    void tampilkanData() {
        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);
        System.out.println("-----");
    }
}
```

### Kode program pada class DataDosen08 :

```
package Jobsheet7;

public class DataDosen08 {
    Dosen08 dataDosen[] = new Dosen08[10];
    int idx;

    void tambah(Dosen08 dsn) {
        if (idx < dataDosen.length) {
            dataDosen[idx] = dsn;
            idx++;
        } else {
            System.out.println("Data Sudah Penuh!");
        }
    }

    void tampil() {
        if (dataDosen[0] == null) {
            System.out.println("Data Kosong!");
        } else {
            System.out.println("Data Dosen : ");
            for (Dosen08 dsn : dataDosen) {
                dsn.tampilkanData();
            }
        }
    }

    void sortingASC() {
        for (int i = 0; i < dataDosen.length - 1; i++) {
            for (int j = 1; j < dataDosen.length - i; j++) {
                if (dataDosen[j].usia < dataDosen[j - 1].usia) {
                    Dosen08 dummy = dataDosen[j];
                    dataDosen[j] = dataDosen[j - 1];
                    dataDosen[j - 1] = dummy;
                }
            }
        }
    }

    void sortingDSC() {
        for (int i = 0; i < dataDosen.length; i++) {
            Dosen08 temp = dataDosen[i];
            int j = i;
            while (j > 0 && dataDosen[j - 1].usia < temp.usia) {
                dataDosen[j] = dataDosen[j - 1];
                j--;
            }
            dataDosen[j] = temp;
        }
    }
}
```

```

void PencarianDataSequential08(String cariNama) {
    int dosen = 0;
    for (int i = 0; i < dataDosen.length; i++) {
        if (dataDosen[i].nama.equalsIgnoreCase(cariNama)) {
            dosen++;
            if (dosen > 1) {
                System.out.println("Data Dosen Yang Dicari Ditemukan
Lebih Dari Satu!");
                break;
            }
            System.out.println("Data Dosen Ditemukan!");
            dataDosen[i].tampilkanData();
        }
    }
    if (dosen == 0) {
        System.out.println("Data Dosen Tidak Ditemukan!");
    }
    System.out.println();
}

void pencarianDataBinary08(int cariUsia) {
    int left = 0, right = idx - 1;
    boolean dataAda = false;
    while (left <= right) {
        int mid = (left + right) / 2;
        if (dataDosen[mid].usia == cariUsia) {
            System.out.println("Data Dosen Ditemukan!");
            dataAda = true;
            dataDosen[mid].tampilkanData();
            for (int i = mid + 1; i <= right; i++) {
                if (dataDosen[i].usia == cariUsia) {
                    System.out.println("Data Dosen Yang Dicari
Ditemukan Lebih Dari Satu!");
                    break;
                }
            }
            for (int i = mid - 1; i >= left; i--) {
                if (dataDosen[i].usia == cariUsia) {
                    System.out.println("Data Dosen Yang Dicari
Ditemukan Lebih Dari Satu!");
                    break;
                }
            }
            break;
        } else if (dataDosen[mid].usia < cariUsia) {
            left = mid + 1;
        } else {
            right = mid - 1;
        }
    }
    if (!dataAda) {
        System.out.println("Data Dosen Tidak Ditemukan!");
    }
    System.out.println();
}
}

```

### Kode program pada class DosenMain08 :

```
package Jobsheet7;

import java.util.Scanner;
public class DosenMain08 {
    static Scanner input = new Scanner(System.in);
    static int menu;
    static DataDosen08 data = new DataDosen08();

    void menampilkanMenu() {
        System.out.println("=====");
        System.out.println("===== Pilihan Menu =====");
        System.out.println("1. Tambah Data Dosen.           \n" +
            "2. Tampil Data Dosen.           \n" +
            "3. Sorting ASC Usia Dosen.       \n" +
            "4. Sorting DSC Usia Dosen.       \n" +
            "5. Search Nama Dosen.           \n" +
            "6. Search Usia Dosen.           \n" +
            "7. Keluar.                       ");
        System.out.println("=====");
        System.out.print("Masukkan Nomor Menu : ");
        menu = input.nextInt();
        input.nextLine();
        System.out.println();
    }

    void menambahkanData() {
        for (int i = 0; i < 10; i++) {
            System.out.println("Masukkan Data Dosen Ke-" + (i + 1));
            System.out.print("Masukkan Kode Dosen           : ");
            String kode = input.nextLine();
            System.out.print("Masukkan Nama Dosen           : ");
            String nama = input.nextLine();
            System.out.print("Jenis Kelamin L (true/false)? : ");
            boolean jenisKelamin = input.nextBoolean();
            System.out.print("Masukkan Usia Dosen           : ");
            int usia = input.nextInt();
            input.nextLine();
            System.out.println("-----");
            Dosen08 dsn = new Dosen08(kode, nama, jenisKelamin, usia);
            data.tambah(dsn);
        }
    }

    void menampilkanDataDosen() {
        data.tampil();
    }

    void mengurutkanASC() {
        data.sortingASC();
        data.tampil();
    }

    void mengurutkanDSC() {
        data.sortingDSC();
        data.tampil();
    }
}
```

```

public static void main(String[] args) {
    do {
        DosenMain08 app = new DosenMain08();
        app.menampilkanMenu();
        switch (app.menu) {
            case 1:
                app.menambahData();
                break;
            case 2:
                app.menampilkanDataDosen();
                break;
            case 3:
                app.mengurutkanASC();
                break;
            case 4:
                app.mengurutkanDSC();
                break;
            case 5:
                System.out.print("Masukkan Nama Dosen yang dicari :
");
                String cariNamaDosen = input.nextLine();
                data.PencarianDataSequential08(cariNamaDosen);
                break;
            case 6:
                System.out.print("Masukkan Usia Dosen yang dicari :
");
                int cariUsiaDosen = input.nextInt();
                data.sortingASC();
                data.pencarianDataBinary08(cariUsiaDosen);
                break;
            case 7:
                System.out.println("Terima Kasih!");
                break;
            default:
                System.out.println("Nomor Menu Tidak Ada!");
                break;
        }
    } while (menu != 7);
}
}

```



### 7.4.2 Hasil Output

```
=====
===== Pilihan Menu =====
1. Tambah Data Dosen.
2. Tampil Data Dosen.
3. Sorting ASC Usia Dosen.
4. Sorting DSC Usia Dosen.
5. Search Nama Dosen.
6. Search Usia Dosen.
7. Keluar.
=====
Masukkan Nomor Menu : 1

Masukkan Data Dosen Ke-1
Masukkan Kode Dosen      : DS001
Masukkan Nama Dosen      : Amelia Ningrum
Jenis Kelamin L (true/false)? : false
Masukkan Usia Dosen      : 52
-----
Masukkan Data Dosen Ke-2
Masukkan Kode Dosen      : DS015
Masukkan Nama Dosen      : Budi Harianto
Jenis Kelamin L (true/false)? : true
Masukkan Usia Dosen      : 45
-----
Masukkan Data Dosen Ke-3
Masukkan Kode Dosen      : DS028
Masukkan Nama Dosen      : Maya Lestari
Jenis Kelamin L (true/false)? : false
Masukkan Usia Dosen      : 38
-----
Masukkan Data Dosen Ke-4
Masukkan Kode Dosen      : DS033
Masukkan Nama Dosen      : Chandra Wijaya
Jenis Kelamin L (true/false)? : true
Masukkan Usia Dosen      : 48
-----
```

```
Masukkan Data Dosen Ke-5
Masukkan Kode Dosen      : DS033
Masukkan Nama Dosen      : Chandra Wijaya
Jenis Kelamin L (true/false)? : true
Masukkan Usia Dosen      : 48
-----

Masukkan Data Dosen Ke-6
Masukkan Kode Dosen      : DS047
Masukkan Nama Dosen      : Rina Dewi
Jenis Kelamin L (true/false)? : false
Masukkan Usia Dosen      : 35
-----

Masukkan Data Dosen Ke-7
Masukkan Kode Dosen      : DS059
Masukkan Nama Dosen      : Agung Santoso
Jenis Kelamin L (true/false)? : true
Masukkan Usia Dosen      : 28
-----

Masukkan Data Dosen Ke-8
Masukkan Kode Dosen      : DS084
Masukkan Nama Dosen      : Dian Anggraini
Jenis Kelamin L (true/false)? : false
Masukkan Usia Dosen      : 32
-----

Masukkan Data Dosen Ke-9
Masukkan Kode Dosen      : DS084
Masukkan Nama Dosen      : Dian Anggraini
Jenis Kelamin L (true/false)? : false
Masukkan Usia Dosen      : 32
-----

Masukkan Data Dosen Ke-10
Masukkan Kode Dosen      : DS096
Masukkan Nama Dosen      : Yusuf Pratama
Jenis Kelamin L (true/false)? : true
Masukkan Usia Dosen      : 30
-----
```

```
=====
===== Pilihan Menu =====
1. Tambah Data Dosen.
2. Tampil Data Dosen.
3. Sorting ASC Usia Dosen.
4. Sorting DSC Usia Dosen.
5. Search Nama Dosen.
6. Search Usia Dosen.
7. Keluar.
```

```
=====
```

Masukkan Nomor Menu : 2

Data Dosen :

Kode Dosen : DS001  
Nama Dosen : Amelia Ningrum  
Jenis Kelamin : Perempuan  
Usia : 52

-----  
Kode Dosen : DS015  
Nama Dosen : Budi Harianto  
Jenis Kelamin : Laki-laki  
Usia : 45

-----  
Kode Dosen : DS028  
Nama Dosen : Maya Lestari  
Jenis Kelamin : Perempuan  
Usia : 38

-----  
Kode Dosen : DS033  
Nama Dosen : Chandra Wijaya  
Jenis Kelamin : Laki-laki  
Usia : 48

-----  
Kode Dosen : DS033  
Nama Dosen : Chandra Wijaya  
Jenis Kelamin : Laki-laki  
Usia : 48

-----

```
=====
===== Pilihan Menu =====
1. Tambah Data Dosen.
2. Tampil Data Dosen.
3. Sorting ASC Usia Dosen.
4. Sorting DSC Usia Dosen.
5. Search Nama Dosen.
6. Search Usia Dosen.
7. Keluar.
```

```
=====
```

Masukkan Nomor Menu : 5

Masukkan Nama Dosen yang dicari : Dian Anggraini

Data Dosen Ditemukan!

Kode Dosen : DS084

Nama Dosen : Dian Anggraini

Jenis Kelamin : Perempuan

Usia : 32

-----

Data Dosen Yang Dicari Ditemukan Lebih Dari Satu!

```
=====
===== Pilihan Menu =====
1. Tambah Data Dosen.
2. Tampil Data Dosen.
3. Sorting ASC Usia Dosen.
4. Sorting DSC Usia Dosen.
5. Search Nama Dosen.
6. Search Usia Dosen.
7. Keluar.
```

```
=====
```

Masukkan Nomor Menu : 6

Masukkan Usia Dosen yang dicari : 48

Data Dosen Ditemukan!

Kode Dosen : DS033

Nama Dosen : Chandra Wijaya

Jenis Kelamin : Laki-laki

Usia : 48

-----

Data Dosen Yang Dicari Ditemukan Lebih Dari Satu!

```
=====
===== Pilihan Menu =====
1. Tambah Data Dosen.
2. Tampil Data Dosen.
3. Sorting ASC Usia Dosen.
4. Sorting DSC Usia Dosen.
5. Search Nama Dosen.
6. Search Usia Dosen.
7. Keluar.
```

```
=====
```

Masukkan Nomor Menu : 7

Terima Kasih!

