

**LAPORAN HASIL PRAKTIKUM
ALGORITMA DAN STRUKTUR DATA
JOBSHEET 6**



Disusun Oleh :

Nama : Nawaf Azril Annaufal

Nim : 244107020047

Kelas : TI 1E

**PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNOLOGI INFORMASI
POLINEMA
2025**

Percobaan 1

1. Hasil penulisan kode program Class Mahasiswa21.java

```
package Praktikum_ASD.Jobsheet6;

public class Mahasiswa21 {

    String nim;
    String nama;
    String kelas;
    double ipk;

    Mahasiswa21(String nim, String nama, String kelas, double ipk) {
        this.nim = nim;
        this.nama = nama;
        this.kelas = kelas;
        this.ipk = ipk;
    }

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

}
```

Class MahasiswaBerprestasi21.java

```
package Praktikum_ASD.Jobsheet6;

public class MahasiswaBerprestasi21 {

    Mahasiswa21[] listMhs = new Mahasiswa21[5];
    int idx;

    void tambah(Mahasiswa21 mhs) {
        if (idx < listMhs.length) {
            listMhs[idx] = mhs;
            idx++;
        } else {
            System.out.println("Data sudah penuh!");
        }
    }

    void tampil() {
        System.out.println("Daftar Mahasiswa Berprestasi:");
        for (Mahasiswa21 mhs : listMhs) {
            mhs.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 index ke-" + 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);
        }
    }
}
```

Class MahasiswaDemo21.java

```
package Praktikum_ASD.Jobsheet6;

import java.util.Scanner;

public class MahasiswaDemo21 {

    public static void main(String[] args) {
        MahasiswaBerprestasi21 list = new MahasiswaBerprestasi21();
        Scanner sc = new Scanner(System.in);
        int jumlahMhs = 5;

        for (int i = 0; i < jumlahMhs; 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      : ");
            String ipk = sc.nextLine();
            Double ipk = Double.parseDouble(ipk);
            System.out.println("-----");
            list.tambah(new Mahasiswa21(nim, nama, kelas, ipk));
        }

        list.tampil();
        System.out.println("-----");
        System.out.println("Pencarian Data Mahasiswa");
        System.out.println("-----");
        System.out.print("Masukkan IPK yang dicari: ");
        System.out.println("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);

        sc.close();
    }
}
```

2. Hasil run kode program

```
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     : sia
Kelas   : 2
IPK      : 3.3
-----

Daftar Mahasiswa Berprestasi:
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: sia
Kelas: 2
IPK: 3.3

-----
Pencarian Data Mahasiswa
-----
Masukkan IPK yang dicari:
IPK: 3.5
Menggunakan Sequential Searching
Data Mahasiswa dengan IPK 3.5 ditemukan pada index ke-3
Nim      : 444
Nama     : lia
Kelas   : 2
IPK      : 3.5
ezreals_ @ LAPTOP-ODCVBL7M in ~/CodeHack>
```

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. Method TampilPosisi digunakan untuk menunjukkan apakah data ditemukan atau tidak, beserta indeksnya, sedangkan tampilDataSearch digunakan untuk menampilkan informasi lengkap mahasiswa jika ditemukan.
2. Fungsi break pada kode tersebut digunakan untuk mengentikan perulangan jika data sudah ditemukan

Percobaan 2

1. Hasil penulisan kode program

Class MahasiswaBerprestasi21.java

```
package Praktikum_ASD.Jobsheet6;

public class MahasiswaBerprestasi21 {

    Mahasiswa21[] listMhs = new Mahasiswa21[5];
    int idx;

    void tambah(Mahasiswa21 mhs) {
        if (idx < listMhs.length) {
            listMhs[idx] = mhs;
            idx++;
        } else {
            System.out.println("Data sudah penuh!");
        }
    }

    void tampil() {
        System.out.println("Daftar Mahasiswa Berprestasi:");
        for (Mahasiswa21 mhs : listMhs) {
            mhs.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;
    // }
```

```

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;
}

void tampilPosisi(double x, int pos) {
    if (pos != -1) {
        System.out.println("Data Mahasiswa dengan IPK " + x + " ditemukan
pada index ke-" + 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);
    }
}
}

```


Class MahasiswaDemo21.java

```
package Praktikum_ASD.Jobsheet6;

import java.util.Scanner;

public class MahasiswaDemo21 {

    public static void main(String[] args) {
        MahasiswaBerprestasi21 list = new MahasiswaBerprestasi21();
        Scanner sc = new Scanner(System.in);
        int jumlahMhs = 5;

        for (int i = 0; i < jumlahMhs; 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      : ");
            String ipk = sc.nextLine();
            Double ipk = Double.parseDouble(ipk);
            System.out.println("-----");
            list.tambah(new Mahasiswa21(nim, nama, kelas, ipk));
        }

        list.tampil();
        System.out.println("-----");
        System.out.println("Pencarian Data Mahasiswa");
        System.out.println("-----");
        System.out.println("Masukkan IPK yang dicari: ");
        System.out.print("IPK: ");
        double cari = sc.nextDouble();

        System.out.println("Menggunakan Binary Search");
        double posisi = list.findBinarySearch(cari, 0, jumlahMhs - 1);
        int pss = (int) posisi;
        list.tampilPosisi(cari, pss);
        list.tampilDataSearch(cari, pss);

        sc.close();
    }
}
```

2. Hasil run kode program

```
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
-----
```

```
Daftar Mahasiswa Berprestasi:
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 Mahasiswa
-----
Masukkan IPK yang dicari:
IPK: 3.7
Menggunakan Binary Search
Data Mahasiswa dengan IPK 3.7 ditemukan pada index ke-4
Nim      : 555
Nama     : anita
Kelas   : 2
IPK      : 3.7
ezreals_ @ LAPTOP-ODCVBL7M in ~/CodeHack>
```

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 (missal : 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 seach agar hasilnya sesuai
5. Modifikasilah program diatas yang mana jumlah mahasiswa yang di inputkan sesuai dengan masukan dari keyboard

Jawab

1. Ini adalah kode program yang menjalankan proses divide

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

2. Ini adalah kode program yang menjalankan proses conquer

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

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

3. Jika data yang dimasukkan tidak urut, program dapat berjalan dengan baik asal menggunakan algoritma sequential search, karena algoritma ini tidak memerlukan pengurutan diawal.
4. Jika diberikan data yang descending maka binary search saat ini tidak akan berjalan dengan baik karena masih menggunakan pengecekan secara ascending, ini adalah perubahan untuk menyesuaikan ke data descending.

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

5. Kode program untuk memasukkan jumlah mahasiswa berdasarkan inputan dari keyboard

```
System.out.println("Masukkan Jumlah Mahasiswa: ");  
    int jumlhMhs = sc.nextInt();  
    sc.nextLine();
```

Tugas

1. Hasil penulisan kode program

Class Dosen21.java

```
package Praktikum_ASD.Jobsheet6;

public class Dosen21 {
    String kode;
    String nama;
    boolean jenisKelamin;
    int usia;

    public Dosen21(String kd, String name, boolean jk, int age) {
        this.kode = kd;
        this.nama = name;
        this.jenisKelamin = jk;
        this.usia = age;
    }

    public 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("-----");
    }
}
```

Class DataDosen21.java

```
package Praktikum_ASD.Jobsheet6;

public class DataDosen21 {
    Dosen21[] dataDosen = new Dosen21[10];
    int idx = 0;

    public void tambah(Dosen21 dsn) {
        if (idx < dataDosen.length) {
            dataDosen[idx] = dsn;
            idx++;
        } else {
            System.out.println("Data dosen sudah penuh!");
        }
    }

    public void tampil() {
        if (idx == 0) {
            System.out.println("Data dosen kosong!");
        } else {
            System.out.println("Daftar Dosen:");
            for (int i = 0; i < idx; i++) {
                dataDosen[i].tampil();
            }
        }
    }

    public void SortingASC() {
        for (int i = 0; i < idx - 1; i++) {
            for (int j = 0; j < idx - i - 1; j++) {
                if (dataDosen[j].usia > dataDosen[j + 1].usia) {
                    Dosen21 temp = dataDosen[j];
                    dataDosen[j] = dataDosen[j + 1];
                    dataDosen[j + 1] = temp;
                }
            }
        }
        System.out.println("Data berhasil diurutkan secara ASCENDING (Termuda ke Tertua).");
    }

    public 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;
                }
            }
            Dosen21 temp = dataDosen[idxMax];
            dataDosen[idxMax] = dataDosen[i];
            dataDosen[i] = temp;
        }
        System.out.println("Data berhasil diurutkan secara DESCENDING (Tertua ke Termuda).");
    }
}
```

```

public void PencarianDataSequential(String namaCari) {
    boolean ditemukan = false;
    int jumlah = 0;

    System.out.println("Hasil pencarian dosen dengan nama \"" + namaCari +
"\":");
    for (int i = 0; i < idx; i++) {
        if (dataDosen[i].nama.equalsIgnoreCase(namaCari)) {
            dataDosen[i].tampil();
            ditemukan = true;
            jumlah++;
        }
    }

    if (!ditemukan) {
        System.out.println("Dosen dengan nama \"" + namaCari + "\" tidak
ditemukan.");
    } else if (jumlah > 1) {
        System.out.println("Ditemukan lebih dari 1 dosen dengan nama \"" +
namaCari + "\".");
    }

}

public void PencarianDataBinary(int usiaCari) {
    SortingASC();
    int kiri = 0, kanan = idx - 1;
    int jumlah = 0;
    boolean ditemukan = false;

    while (kiri <= kanan) {
        int mid = (kiri + kanan) / 2;
        if (dataDosen[mid].usia == usiaCari) {
            for (int i = mid; i >= 0 && dataDosen[i].usia == usiaCari; i--) {
                dataDosen[i].tampil();
                jumlah++;
            }
            for (int i = mid + 1; i < idx && dataDosen[i].usia == usiaCari;
i++) {
                dataDosen[i].tampil();
                jumlah++;
            }
            ditemukan = true;
            break;
        } else if (dataDosen[mid].usia < usiaCari) {
            kiri = mid + 1;
        } else {
            kanan = mid - 1;
        }
    }

    if (!ditemukan) {
        System.out.println("Dosen dengan usia " + usiaCari + " tidak
ditemukan.");
    } else if (jumlah > 1) {
        System.out.println("Ditemukan lebih dari 1 dosen dengan usia " +
usiaCari + ".");
    }

}

}

```

Class DosenDemo21.java

```
package Praktikum_ASD.Jobsheet6;

import java.util.Scanner;

public class DosenDemo21 {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        DataDosen21 daftarDosen = new DataDosen21();
        int pilihan;

        do {
            System.out.println("\n=== MENU DATA DOSEN ===");
            System.out.println("1. Tambah Data");
            System.out.println("2. Tampil Data");
            System.out.println("3. Sorting ASC (Usia Termuda ke Tertua)");
            System.out.println("4. Sorting DSC (Usia Tertua ke Termuda)");
            System.out.println("5. Pencarian Data Dosen Berdasarkan Nama");
            System.out.println("6. Pencarian Data Dosen Berdasarkan Usia");
            System.out.println("0. Keluar");
            System.out.print("Pilih menu: ");
            pilihan = sc.nextInt();
            sc.nextLine();

            switch (pilihan) {
                case 1:
                    if (daftarDosen.idx < 10) {
                        System.out.print("Masukkan Kode Dosen: ");
                        String kode = sc.nextLine();
                        System.out.print("Masukkan Nama Dosen: ");
                        String nama = sc.nextLine();
                        System.out.print("Masukkan Jenis Kelamin (L = Laki-laki, P = Perempuan): ");

                        char jk = sc.next().charAt(0);
                        boolean jenisKelamin = (jk == 'L' || jk == 'l');
                        System.out.print("Masukkan Usia Dosen: ");
                        int usia = sc.nextInt();
                        sc.nextLine();

                        Dosen21 dsn = new Dosen21(kode, nama, jenisKelamin,
                        usia);

                        daftarDosen.tambah(dsn);
                        System.out.println("Data dosen berhasil ditambahkan!");
                    } else {
                        System.out.println("Data sudah penuh!");
                    }
                    break;

                case 2:
                    daftarDosen.tampil();
                    break;

                case 3:
                    daftarDosen.SortingASC();
                    daftarDosen.tampil();
                    break;
            }
        } while (pilihan != 0);
    }
}
```



```

case 4:
    daftarDosen.SortingDSC();
    daftarDosen.tampil();
    break;

case 5:
    System.out.print("Masukkan nama dosen yang dicari: ");
    String namaCari = sc.nextLine();
    daftarDosen.PencarianDataSequential(namaCari);
    break;

case 6:
    System.out.print("Masukkan usia dosen yang dicari: ");
    int usiaCari = sc.nextInt();
    daftarDosen.PencarianDataBinary(usiaCari);
    break;

case 0:
    System.out.println("Terima kasih, program selesai.");
    break;

default:
    System.out.println("Pilihan tidak valid. Silakan coba
lagi.");
    }
    } while (pilihan != 0);

    sc.close();
    }
}

```