

**LAPORAN PRAKTIKUM ALGORITMA DAN STRUKTUR DATA
SEARCHING**



AHMAD SOFYAN BADAWI

244107020073

KELAS TI-1B

**PRODI D-IV TEKNIK INFORMATIKA
JURUSAN TEKNOLOGI INFORMASI
POLITEKNIK NEGERI MALANG**

2025

1. Percobaan Praktikum

1.1 Percobaan 1 (Sequential Searching)

- Pada pertemuan Jobsheet 7 ini akan menggunakan class Mahasiswa, MahasiswaBerprestasi, dan MahasiswaDemo pada pertemuan Jobsheet 6 sebelumnya
- Buat folder baru bernama Jobsheet7 di dalam repository Praktikum ASD, kemudian buka ketiga class dari Jobsheet 6 tersebut dan copy ke folder Jobsheet 7
- 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;  
            break;  
        }  
    }  
    return posisi;  
}
```

- Buatlah method tampilPosisi bertipe void dan Deklarasikan isi dari method tampilPosisi pada class MahasiswaBerprestasi

```
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");  
    }  
}
```

- 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");  
    }  
}
```

- Pada class MahasiswaDemo, tambahkan kode program berikut ini untuk melakukan pencarian data dengan algoritma sequential searching.

```
int jumMhs = 5;  
  
for (int i = 0; i < jumMhs; i++) {  
    System.out.println("Masukkan data mahasiswa ke-" + (i+1));  
    System.out.print(s:"NIM\t: ");  
    String nim = sc.nextLine();  
}
```

```

System.out.println();
//sequential search
System.out.println(x:"-----");
System.out.println(x:"Pencarian data");
System.out.println(x:"-----");
System.out.println(x:"masukkan ipk mahasiswa yang dicari");
System.out.print(s:"IPK: ");
double cari = sc.nextDouble();

System.out.println(x:"-----");
System.out.println(x:"menggunakan sequential searching");
System.out.println(x:"-----");
double posisi = list.sequentialSearching(cari);
int pss = (int) posisi;
list.tampilPosisi(cari, pss);
list.tampilDataSearch(cari, pss);

```

- Jalankan dan amati hasilnya.

1.1.1 Pertanyaan

1. Jelaskan perbedaan metod tampilDataSearch dan tampilPosisi pada class MahasiswaBerprestasi!
 - Method tampilDataSearch digunakan untuk mencari dan menampilkan data yang dicari
 - Method tampilPosisi digunakan untuk mencari dan menampilkan dimana posisi indeks data yang sudah ditemukan itu
2. Jelaskan fungsi break pada kode program dibawah ini!

```

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

```

- Fungsi dari break dalam kode tersebut adalah menghentikan looping secara langsung ketika kondisi if sudah terpenuhi. Jika kondisi if (listMhs[j].ipk == cari) true maka variabel posisi diisi dengan nilai indeks j tempat IPK ditemukan. Kemudian break; menghentikan loop, sehingga pencarian berhenti ketika data pertama yang cocok sudah ditemukan.

1.2 Percobaan 2 (Binary Searching)

- 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, mid+1, right); //
        } else {
            return findBinarySearch(cari, left, mid-1); //
        }
    }
    return -1;
}

```

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

```

System.out.println();
//binary search
System.out.println(x:"-----");
System.out.println(x:"menggunakan binary search");
System.out.println(x:"-----");
double posisi2 = list.findBinarySearch(cari, left:0, jumMhs-1);
int pss2 = (int) posisi2;
list.tampilPosisi(cari, pss2);
list.tampilDataSearch(cari, pss2);
}

```

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

1.2.1 Pertanyaan

1. Tunjukkan pada kode program yang mana proses divide dijalankan!

```

mid = (left+right)/2;

```

2. Tunjukkan pada kode program yang mana proses conquer dijalankan!

```

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

```

3. Jika data IPK yang dimasukkan tidak urut. Apakah program masih dapat berjalan? Mengapa demikian!
 - Tidak bisa karena pada konsep Binary Search dia menggunakan divide conquer dimana dia akan memulai mencari data dari tengah dan kemudian akan mengecek nilai yang ada di tengah apakah lebih kecil atau lebih besar yang nantinya akan mempengaruhi pengecekan selanjutnya akan berlanjut ke arah kanan atau kiri. Namun, apabila data tidak diurut maka akan memungkinkan dia mengecek data yang sebagian saja dan bisa saja datanya tidak ditemukan karena hanya mengecek satu bagian saja
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

```

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, mid+1, right); //
        } else {
            return findBinarySearch(cari, left, mid-1); //
        }
    }
    return -1;
}

```

5. Modifikasilah program diatas yang mana jumlah mahasiswa yang di inputkan sesuai dengan masukan dari keyboard.
 - Ada pada file yang sudah saya push

2. Latihan Praktikum

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!

Jawaban ada pada file yang saya up di github