

Laporan Hasil Praktikum Algoritma Struktur Data

Jobsheet 7



Febryan Akhmad Taajuddin

244107020180

Kelas 1E

Program Studi Teknik Informatika

Jurusan Teknologi Informasi

Politeknik Negeri Malang

2025

Percobaan 1

1. Tambahkan method sequentialSearching pada class MahasiswaBerprestasi10

```
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;
}
```

2. Buat method tampilPosisi pada class MahasiswaBerprestasi10

```
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");
    }
}
```

3. Buat method tampilDataSearch pada class MahasiswaBerprestasi10

```
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");
    }
}
```

4. Modifikasi class MahasiswaDemo10 untuk melakukan pencarian data dengan algoritma sequential searching

```
        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: ");
            double ipk = sc.nextDouble();
            sc.nextLine();

            list.tambah(new Mahasiswa10(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.sequentialSearching(cari);
        int pss = (int)posisi;
        list.tampilPosisi(cari, pss);
        list.tampilDataSearch(posisi, pss);
    }
}
```

5. 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: fia
Kelas: 2
IPK: 3,3
```

```

Nama: adi
NIM: 111
IPK: 3.6
Kelas: 2
-----
Nama: tio
NIM: 222
IPK: 3.8
Kelas: 2
-----
Nama: ila
NIM: 333
IPK: 3.0
Kelas: 2
-----
Nama: lia
NIM: 444
IPK: 3.5
Kelas: 2
-----
Nama: fia
NIM: 555
IPK: 3.3
Kelas: 2
-----
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

```

Pertanyaan percobaan 1

1. Jelaskan perbedaan method tampilDataSearch dan tampilPosisi pada class MahasiswaBerprestasi
2. Jelaskan fungsi break pada program dibawah ini

```

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

```

Jawaban percobaan 1

1. Method tampilPosisi hanya fokus pada posisi data yang ditemukan, sedangkan method tampilDataSearch lebih detail mahasiswa berdasarkan posisi data yang ditemukan
2. Untuk menghentikan perulangan ketika kondisi listMhs[j].ipk == cari. Jika data mahasiswa yang dicari sudah ditemukan, perulangan tidak akan dilanjutkan ke iterasi selanjutnya

Percobaan 2

1. Tambahkan method findBinarySearch pada class MahasiswaBerprestasi

```
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 pada class MahasiswaBerprestasi lalu 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,jumlahMhs-1);
int pss2= (int)posisi2;
list.tampilPosisi(cari, pss2);
list.tampilDataSearch(cari, pss2);
```

3. 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
```

```

Nama: adi
NIM: 111
IPK: 3.1
Kelas: 2
-----
Nama: ila
NIM: 222
IPK: 3.2
Kelas: 2
-----
Nama: lia
NIM: 333
IPK: 3.3
Kelas: 2
-----
Nama: susi
NIM: 444
IPK: 3.5
Kelas: 2
-----
Nama: anita
NIM: 555
IPK: 3.7
Kelas: 2
-----
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

```

Pertanyaan percobaan 2

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.

Jawaban percobaan 2

1. Proses divide terjadi pada method findBinarySearch

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

2. Proses divide terjadi pada method findBinarySearch

```

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. Jika IPK dimasukkan tidak urut maka program tidak bisa berjalan, karena Algoritma binary search hanya bekerja pada data yang sudah terurut (baik secara ascending maupun descending). Hal ini karena binary search membagi data menjadi dua bagian berdasarkan nilai tengah, dengan asumsi bahwa data di sebelah kiri lebih kecil dan data di sebelah kanan lebih besar (atau sebaliknya, tergantung urutan).

4. Modifikasi kode program

```
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;
}
```

- Output

```
Masukkan data mahasiswa ke-1
NIM: 1
Nama: adi
Kelas: 1e
IPK: 3,8
Masukkan data mahasiswa ke-2
NIM: 2
Nama: budi
Kelas: 1a
IPK: 3,7
Masukkan data mahasiswa ke-3
NIM: 3
Nama: cinta
Kelas: 1b
IPK: 3,5
Masukkan data mahasiswa ke-4
NIM: 4
Nama: diko
Kelas: 1c
IPK: 3,4
Masukkan data mahasiswa ke-5
NIM: 5
Nama: egie
Kelas: 1d
IPK: 3,2
```

```
-----
Pencarian data
-----
Masukkan IPK mahasiswa yang dicari
IPK: 3,2
-----
menggunakan binary search
-----
data mahasiswa dengan IPK :3.2 ditemukan pada indeks 4
nim      : 5
nama     : egie
kelas    : 1d
ipk      : 3.2
```

5. Modifikasi kode program

- Class MahasiswaBerprestasi10

```
MahasiswaBerprestasi10(int jumlah) {  
    listMhs = new Mahasiswa10[jumlah];  
    idx = 0;  
}
```

- Class MahasiswaDemo10

```
System.out.print("Masukkan jumlah mahasiswa: ");  
int jumlahMhs = sc.nextInt();  
sc.nextLine();  
  
MahasiswaBerprestasi10 list = new  
MahasiswaBerprestasi10(jumlahMhs);  
for (int i = 0; i < jumlahMhs; i++) {
```

- Output

```
Masukkan jumlah mahasiswa: 3  
Masukkan data mahasiswa ke-1  
NIM: 1  
Nama: adi  
Kelas: 1e  
IPK: 3,8  
Masukkan data mahasiswa ke-2  
NIM: 2  
Nama: budi  
Kelas: 1a  
IPK: 3,7  
Masukkan data mahasiswa ke-3  
NIM: 3  
Nama: cinta  
Kelas: 1b  
IPK: 3,5
```

```
-----  
Pencarian data  
-----  
Masukkan IPK mahasiswa yang dicari  
IPK: 3,5  
-----  
menggunakan binary search  
-----  
data mahasiswa dengan IPK :3.5 ditemukan pada indeks 2  
nim      : 3  
nama     : cinta  
kelas    : 1b  
ipk      : 3.5
```


Latihan Praktikum

1. Berikut kode programnya

a. PencarianDataSequential10

```
int[] PencarianDataSequential10(String cari) {  
    int[] posisi = new int[idx];  
    int jumlahData = 0;  
  
    for (int j = 0; j < idx; j++) {  
        if (dataDosen[j] != null &&  
dataDosen[j].nama.equalsIgnoreCase(cari)) {  
            posisi[jumlahData] = j;  
            jumlahData++;  
        }  
    }  
  
    if (jumlahData == 0) {  
        return new int[0];  
    } else {  
        int[] hasilCari = new int[jumlahData];  
        for (int i = 0; i < jumlahData; i++) {  
            hasilCari[i] = posisi[i];  
        }  
        return hasilCari;  
    }  
}
```

b. PencarianDataBinary10

```
int[] PencarianDataBinary10(int cariUsia) {
    bubbleSort();

    int posisiAwal = cariPosisiAwalBinary(cariUsia, 0, idx - 1);

    if (posisiAwal == -1) {
        return new int[0];
    }

    int[] hasil = new int[idx];
    int jumlahHasil = 0;

    int left = posisiAwal;
    while (left >= 0 && dataDosen[left].usia == cariUsia) {
        hasil[jumlahHasil++] = left;
        left--;
    }

    int right = posisiAwal + 1;
    while (right < idx && dataDosen[right].usia == cariUsia) {
        hasil[jumlahHasil++] = right;
        right++;
    }

    int[] hasilFinal = new int[jumlahHasil];
    for (int i = 0; i < jumlahHasil; i++) {
        hasilFinal[i] = hasil[i];
    }

    return hasilFinal;
}
```

c. Menampilkan hasil pencarian dengan peringatan jika lebih dari 1 hasil

```
void tampilHasilPencarian(String jenisData, String kriteria, int[]
posisi) {
    if (posisi.length == 0) {
        System.out.println(jenisData + " dengan " + kriteria + "
tidak ditemukan");
        return;
    }

    if (posisi.length > 1) {
        System.out.println("PERINGATAN: Ditemukan " +
posisi.length + " data " + jenisData +
" dengan " + kriteria + " yang sama!");
    }

    System.out.println("Hasil pencarian " + jenisData + " dengan "
+ kriteria + ":");
    for (int i = 0; i < posisi.length; i++) {
        System.out.println("Hasil ke-" + (i+1) + ":");
        dataDosen[posisi[i]].tampil();
        System.out.println("-----");
    }
}
```

- Output

```
Masukkan data dosen ke-1
Kode: 1
Nama: Zainul
Jenis kelamin (L/P): L
Usia: 30
-----
```

```
Masukkan data dosen ke-2
Kode: 2
Nama: Adi
Jenis kelamin (L/P): L
Usia: 20
-----
```

```
Masukkan data dosen ke-3
Kode: 3
Nama: Yaya
Jenis kelamin (L/P): P
Usia: 29
-----
```

```
Masukkan data dosen ke-4
Kode: 4
Nama: Biry
Jenis kelamin (L/P): L
Usia: 21
-----
```

```
Masukkan data dosen ke-5
Kode: 5
Nama: Sugeng
Jenis kelamin (L/P): L
Usia: 28
-----
```

```
===== MENU Pencarian =====
1. Pencarian berdasarkan nama (Sequential Search)
2. Pencarian berdasarkan usia (Binary Search)
0. Keluar
Pilihan: 1
Pencarian Sequential berdasarkan nama
Masukkan nama dosen yang dicari: Biry
Hasil pencarian dosen dengan nama Biry:
Hasil ke-1:
Kode: 4
Nama: Biry
Jenis kelamin: L
Usia: 21
-----
```

```
===== MENU Pencarian =====
1. Pencarian berdasarkan nama (Sequential Search)
2. Pencarian berdasarkan usia (Binary Search)
0. Keluar
Pilihan: 2
Pencarian Binary berdasarkan usia
Masukkan usia dosen yang dicari: 29
Hasil pencarian dosen dengan usia 29:
Hasil ke-1:
Kode: 3
Nama: Yaya
Jenis kelamin: P
-----
```

```
===== MENU Pencarian =====
1. Pencarian berdasarkan nama (Sequential Search)
2. Pencarian berdasarkan usia (Binary Search)
0. Keluar
Pilihan: 0
Terima kasih telah menggunakan program ini!
```