

Nama : Zaskia Maulidina Mutiara Hati

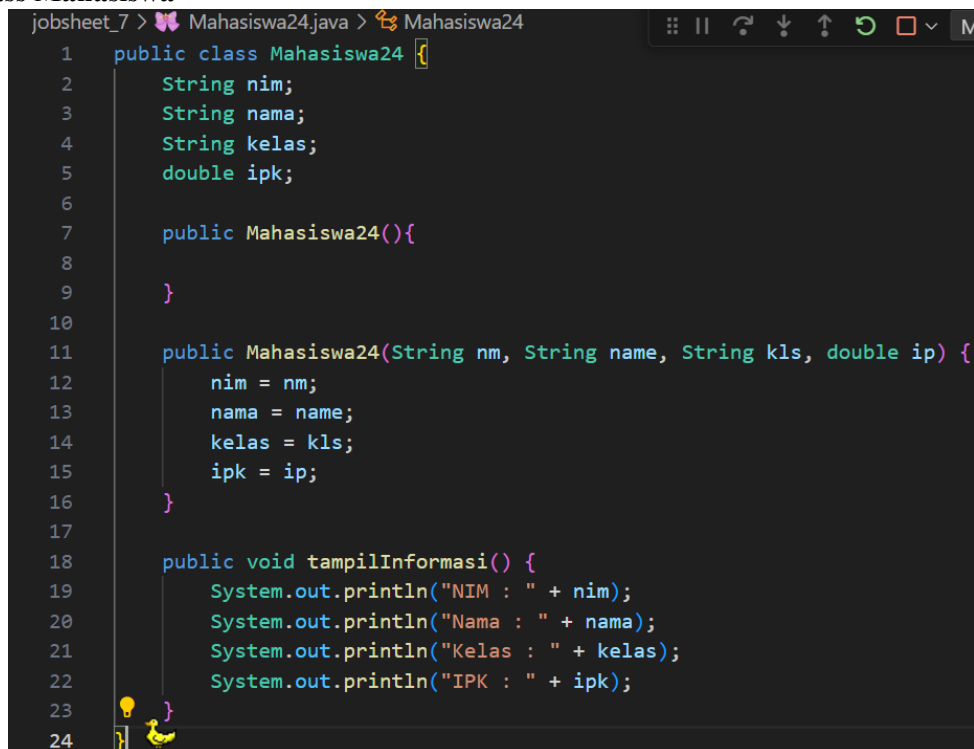
NIM : 244107060056

Kelas : SIB 1A

JOB SHEET 7 ALGORITMA STRUKTUR DATA

Sequential Search

- **Class Mahasiswa**



```
jobsheet_7 > Mahasiswa24.java > Mahasiswa24
1  public class Mahasiswa24 {
2      String nim;
3      String nama;
4      String kelas;
5      double ipk;
6
7      public Mahasiswa24(){
8
9      }
10
11     public Mahasiswa24(String nm, String name, String kls, double ip) {
12         nim = nm;
13         nama = name;
14         kelas = kls;
15         ipk = ip;
16     }
17
18     public void tampilInformasi() {
19         System.out.println("NIM : " + nim);
20         System.out.println("Nama : " + nama);
21         System.out.println("Kelas : " + kelas);
22         System.out.println("IPK : " + ipk);
23     }
24 }
```

- **Class Mahasiswa Berprestasi**

```

jobsheet_7 > MahasiswaBerprestasi24.java > MahasiswaBerprestasi24 > tampilDataSearch(double, int)
1 public class MahasiswaBerprestasi24 {
2     Mahasiswa24 [] listMhs = new Mahasiswa24 [5];
3     int idx;
4
5     void tambah(Mahasiswa24 m){
6         if (idx < listMhs.length){
7             listMhs[idx] = m;
8             idx++;
9         } else {
10             System.out.println(x:"Data sudah penuh!!");
11         }
12     }
13
14     void tampil(){
15         for (Mahasiswa24 m : listMhs){
16             m.tampilInformasi();
17             System.out.println(x:"-----");
18         }
19     }
20
21     int sequentialSearching(double cari) {
22         int posisi = -1;
23         for (int j=0; j < listMhs.length; j++){
24             if (listMhs[j].ipk == cari){
25                 posisi = j;
26                 break;
27             }
28         }
29         return posisi;
30     }
31
32     void tampilPosisi(double x, int pos) {
33         if (pos!= -1){
34             System.out.println("data mahasiswa dengan IPK : " + x + "ditemukan pada indeks " + pos);
35         } else {
36             System.out.println("data " + x + "tidak ditemukan");
37         }
38     }
39
40     void tampilDataSearch(double x, int pos) {
41         if (pos != -1) {
42             System.out.println("nim\t : " + listMhs[pos].nim);
43             System.out.println("nama\t : " + listMhs[pos].nama);
44             System.out.println("kelas\t : " + listMhs[pos].kelas);
45             System.out.println("ipk\t : " + x);
46         } else {
47             System.out.println("data mahasiswa dengan IPK " + x + "tidak ditemukan");
48         }
49     }
50 }

```

- **Class Mahasiswa Demo**

```
jobsheet_7 > MahasiswaDemo24.java > MahasiswaDemo24
1  import java.util.Scanner;
2
3  public class MahasiswaDemo24 {
4      Run | Debug
5      public static void main(String[] args) {
6          Scanner input24 = new Scanner(System.in);
7
8          MahasiswaBerprestasi24 list = new MahasiswaBerprestasi24();
9          int jumlah = 5;
10
11         for (int i = 0; i < jumlah; i++) {
12             System.out.println("Masukkan data mahasiswa ke-" + (i + 1));
13             System.out.print(s:"NIM : ");
14             String nim = input24.nextLine();
15             System.out.print(s:"Nama : ");
16             String nama = input24.nextLine();
17             System.out.print(s:"Kelas : ");
18             String kelas = input24.nextLine();
19             System.out.print(s:"IPK : ");
20             String ip = input24.nextLine();
21             double ipk = Double.parseDouble(ip);
22             System.out.println(x:"-----");
23             list.tambah(new Mahasiswa24(nim, nama, kelas, ipk));
24
25         list.tampil();
26         //melakukan pencarian data sequential
27         System.out.println(x:"-----");
28         System.out.println(x:"Pencarian Data");
29         System.out.println(x:"Masukkan ipk mahasiswa yang dicari: ");
30         System.out.print(s:"IPK: ");
31         double cari = input24.nextDouble();
32
33         System.out.println(x:"menggunakan sequential searching");
34         double posisi = list.sequentialSearching(cari);
35         int pss = (int) posisi;
36         list.tampilPosisi(cari, pss);
37         list.tampilDataSearch(cari, pss);
38     }
```

- **Output**

```

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 : 3.5ditemukan pada indeks 3
nim      : 444
nama     : lia
kelas    : 2
ipk      : 3.5

```

- **Pertanyaan Sequential Search**

1. Jelaskan perbedaan metod tampilDataSearch dan tampilPosisi pada class MahasiswaBerprestasi!
 - Method tampilDataSearch digunakan untuk mencetak informasi lengkap mahasiswa. Sedangkan method tampilPosisi hanya menampilkan posisi(indeks) dari mahasiswa dengan IPK tertentu.

2. Jelaskan fungsi break pada kode program dibawah ini

```

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

```

- Fungsi break pada kode program ini untuk menghentikan looping setelah menemukan data yang dicari.

Binary Search

- Class Mahasiswa Berprestasi

```
32     int findBinarySearch(double cari, int left, int right) {
33         int mid;
34         if (right >= left) {
35             mid = (left + right) / 2;
36             if (cari == listMhs[mid].ipk) {
37                 return (mid);
38             } else if (listMhs[mid].ipk > cari) {
39                 return findBinarySearch(cari, left, mid - 1);
40             } else {
41                 return findBinarySearch(cari, mid + 1, right);
42             }
43         }
44         return -1;
45     }
```

- Class Mahasiswa Demo

```
41         //melakukan pencarian data Binaary
42         System.out.println(x:"-----");
43         System.out.println(x:"Pencarian Data");
44         System.out.println(x:"-----");
45         System.out.println(x:"Masukkan ipk mahasiswa yang dicari: ");
46         System.out.print(s:"IPK: ");
47         double cari2 = input24.nextDouble();
48         System.out.println(x:"-----");
49         System.out.println(x:"menggunakan binary searching");
50         System.out.println(x:"-----");
51         double posisi2 = list.findBinarySearch(cari2, left:0, jumlah - 1);
52         int pss2 = (int) posisi2;
53         list.tampilPosisi(cari2, pss2);
54         list.tampilDataSearch(cari2, pss2);
55     }
56 }
```

- Output

```
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 searching
-----
data mahasiswa dengan IPK : 3.7ditemukan pada indeks 4
nim      : 555
nama     : anita
kelas    : 2
ipk      : 3.7
```

- **Pertanyaan Binary Search**

1. Tunjukkan pada kode program yang mana proses divide dijalankan!
 - Divide terjadi saat kode membagi array menjadi dua bagian berdasarkan nilai tengah (mid).

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

- Kode ini menentukan elemen tengah dari array dan membagi pencarian menjadi dua bagian:

- Bagian kiri → Jika elemen yang dicari lebih kecil dari nilai tengah.
- Bagian kanan → Jika elemen yang dicari lebih besar dari nilai tengah.

2. Tunjukkan pada kode program yang mana proses conquer dijalankan!
 - Conquer terjadi saat kode memeriksa nilai tengah (mid) dan melakukan rekursi ke bagian yang sesuai.

```
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 data IPK yang dimasukkan tidak urut. Apakah program masih dapat berjalan? Mengapa demikian!
 - Tidak, karena Binary search hanya bekerja pada data yang terurut.
4. Jika IPK yang dimasukkan dari IPK terbesar ke terkecil (misal : 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 search agar hasilnya sesuai
 - Jika IPK disusun dari terbesar ke terkecil, maka binary search akan memberikan hasil yang tidak sesuai.

```
32 int findBinarySearch(double cari, int left, int right) {  
33     int mid;  
34     if (right >= left) {  
35         mid = (left + right) / 2;  
36         if (cari == listMhs[mid].ipk) {  
37             return (mid);  
38         } else if (listMhs[mid].ipk < cari) {  
39             return findBinarySearch(cari, left, mid - 1);  
40         } else {  
41             return findBinarySearch(cari, mid + 1, right);  
42         }  
43     }  
44     return -1;  
45 }
```

5. Modifikasilah program diatas yang mana jumlah mahasiswa yang di inputkan sesuai dengan masukan dari keyboard.

```
3 public class MahasiswaDemo24 {
    Run | Debug
4     public static void main(String[] args) {
5         Scanner input24 = new Scanner(System.in);
6
7         MahasiswaBerprestasi24 list = new MahasiswaBerprestasi24();
8         System.out.println(x:"Masukkan jumlah data mahasiswa : ");
9         int jumlah = input24.nextInt();
10        input24.nextLine();
11
12        for (int i = 0; i < jumlah; i++) {
13            System.out.println("Masukkan data mahasiswa ke-" + (i + 1));
14            System.out.print(s:"NIM : ");
15            String nim = input24.nextLine();
16            System.out.print(s:"Nama : ");
17            String nama = input24.nextLine();
18            System.out.print(s:"Kelas : ");
19            String kelas = input24.nextLine();
20            System.out.print(s:"IPK : ");
21            String ip = input24.nextLine();
22            double ipk = Double.parseDouble(ip);
23            System.out.println(x:"-----");
24            list.tambah(new Mahasiswa24(nim, nama, kelas, ipk));
25        }
    }
```

Latihan Praktikum

- Class Dosen

```
jobsheet_7 > Dosen24.java > Dosen24 > tampil()
1  public class Dosen24 {
2      String kode;
3      String nama;
4      String jenisKelamin;
5      int usia;
6
7      Dosen24(){
8
9      }
10
11     Dosen24(String kd, String name, String jk, int age){
12         kode = kd;
13         nama = name;
14         this.jenisKelamin = jk;
15         usia = age;
16     }
17
18     void tampil(){
19         System.out.println("Kode : "+ kode);
20         System.out.println("Nama : "+ nama);
21         System.out.println("Jenis Kelamin : "+ jenisKelamin);
22         System.out.println("Usia : "+ usia);
23     }
24 }
```


- **Class Data Dosen**

```

jobsheet_7 > DataDosen24.java > DataDosen24 > PencarianDataSequential24(Dosen24[], String
1  import java.util.Arrays;
2  public class DataDosen24 {
3      Dosen24 [] listDos = new Dosen24[3];
4      int jumlah = 0;
5
6      void tambah(Dosen24 d){
7          listDos[jumlah] = d;
8          jumlah++;
9      }
10
11     void tampil(){
12         System.out.println(x:"Data Dosen");
13         for (Dosen24 d : listDos){
14             d.tampil();
15             System.out.println(x:"-----");
16         }
17     }
18
19     int PencarianDataSequential24(Dosen24[] listDos, String cariNama) {
20         for (int j=0; j < listDos.length; j++){
21             if (listDos[j].nama.equalsIgnoreCase(cariNama)){
22                 return j;
23             }
24         }
25         return -1;
26     }
27
28     int PencarianDataBinary24(Dosen24[] listDos, String usiaStr) {
29         int usia = Integer.parseInt(usiaStr);
30         Arrays.sort(listDos, fromIndex:0, jumlah, (a, b) -> Integer.compare(a.usia, b.usia)); // Urutkan berdasarkan usia
31         int low = 0, high = jumlah - 1;
32         while (low <= high) {
33             int mid = (low + high) / 2;
34             if (listDos[mid].usia == usia) {
35                 return mid;
36             } else if (listDos[mid].usia < usia) {
37                 low = mid + 1;
38             } else {
39                 high = mid - 1;
40             }
41         }
42         return -1;
43     }
44
45     void tampilPosisiNama(String nama, int pos) {
46         if (pos != -1){
47             System.out.println("data dosen dengan nama : " + nama + "ditemukan pada indeks " + pos);
48         } else {
49             System.out.println("data " + nama + "tidak ditemukan");
50         }
51     }
52
53     void tampilPosisiUsia(String usia, int pos) {
54         if (pos != -1){
55             System.out.println("data dosen dengan usia : " + usia + "ditemukan pada indeks " + pos);
56         } else {
57             System.out.println("data " + usia + "tidak ditemukan");
58         }
59     }
60
61     void tampilDataSearch(String x, int pos) {
62         if (pos != -1) {
63             System.out.println("kode\t : " + listDos[pos].kode);
64             System.out.println("nama\t : " + listDos[pos].nama);
65             System.out.println("jenis kelamin\t : " + listDos[pos].jenisKelamin);
66             System.out.println("usia\t : " + x);
67         }
68     }
69 }

```

- **Class Dosen Demo**

```

jobsheet_7 > 🍷 DosenDemo24.java > 🚀 DosenDemo24 > 🏠 main(String[])
1  import java.util.Scanner;
2  public class DosenDemo24 {
    Run | Debug
3      public static void main(String[] args) {
4          Scanner input24 = new Scanner (System.in);
5
6          DataDosen24 data = new DataDosen24();
7          System.out.print(s:"Masukkan jumlah data dosen : ");
8          int jumlah = input24.nextInt();
9          input24.nextLine();
10
11         for (int i = 0; i < jumlah; i++) {
12             System.out.println("Masukkan data dosen ke-" + (i + 1));
13             System.out.print(s:"Kode : ");
14             String kode = input24.nextLine();
15             System.out.print(s:"Nama : ");
16             String nama = input24.nextLine();
17             System.out.print(s:"Jenis Kelamin : ");
18             String jk = input24.nextLine();
19             System.out.print(s:"Usia : ");
20             String usia = input24.nextLine();
21             int age = Integer.parseInt(usia);
22             System.out.println(x:"-----");
23             data.tambah(new Dosen24(kode, nama, jk, age));
24         }
25
26         data.tampil();
27
28         //melakukan pencarian data sequential
29         System.out.println(x:"-----");
30         System.out.println(x:"Pencarian Data");
31         System.out.println(x:"-----");
32         System.out.println(x:"Masukkan nama dosen yang dicari: ");
33         System.out.print(s:"Nama: ");
34         String cari = input24.nextLine();
35         System.out.println(x:"-----");
36         System.out.println(x:"menggunakan sequential searching");
37         System.out.println(x:"-----");
38         int posisi = data.PencarianDataSequential24(data.listDos, cari);
39         data.tampilPosisiNama(cari, posisi);
40
41         //melakukan pencarian data Binaary
42         System.out.println(x:"-----");
43         System.out.println(x:"Pencarian Data");
44         System.out.println(x:"-----");
45         System.out.println(x:"Masukkan usia dosen yang dicari: ");
46         System.out.print(s:"Usia: ");
47         String cari2 = input24.nextLine();
48         int posisi2 = data.PencarianDataBinary24(data.listDos, cari2);
49         System.out.println(x:"-----");
50         System.out.println(x:"menggunakan binary searching");
51         System.out.println(x:"-----");
52         data.tampilPosisiUsia(cari2, posisi2);

```

- **Output**

```
Masukkan jumlah data dosen : 3
Masukkan data dosen ke-1
Kode : 101
Nama : Zaskia
Jenis Kelamin : Perempuan
Usia : 23
-----
Masukkan data dosen ke-2
Kode : 102
Nama : Maulidina
Jenis Kelamin : Perempuan
Usia : 26
-----
Masukkan data dosen ke-3
Kode : 103
Nama : Mutiara
Jenis Kelamin : Perempuan
Usia : 29
-----
Data Dosen
Kode : 101
Nama : Zaskia
Jenis Kelamin : Perempuan
Usia : 23
-----
Kode : 102
Nama : Maulidina
Jenis Kelamin : Perempuan
Usia : 26
-----
Kode : 103
Nama : Mutiara
Jenis Kelamin : Perempuan
Usia : 29
-----
```

```
-----
Pencarian Data
```

```
-----
Masukkan nama dosen yang dicari:
Nama: Mutiara
```

```
-----
menggunakan sequential searching
```

```
-----
data dosen dengan nama : Mutiara ditemukan pada indeks 2
```

```
-----
Pencarian Data
```

```
-----
Masukkan usia dosen yang dicari:
Usia: 29
```

```
-----
menggunakan binary searching
```

```
-----
data dosen dengan usia : 29 ditemukan pada indeks 2
```