

JOBSHEET IV

SEARCHING

4.1 Tujuan Praktikum

Setelah melakukan materi praktikum ini, mahasiswa mampu:

1. Menjelaskan mengenai algoritma Searching
2. Membuat dan mendeklarasikan struktur algoritma Searching
3. Menerapkan dan mengimplementasikan algoritma Searching

4.2 Praktikum 1

Pada percobaan ini, kita akan mengimplementasikan penggunaan unordered sequential search pada array of object.

4.2.1 Langkah-langkah Percobaan

1. Pada Project **StrukturData** yang sudah dibuat pada Minggu sebelumnya, buat package dengan nama **minggu5**, kemudian buat class baru dengan nama **Mahasiswa**. Tambahkan atribut kode, nim, namaMahasiswa dan ipk, serta berikan konstruktor berparameter seperti gambar berikut ini.

```
public class Mahasiswa {  
  
    public String nim;  
    public String namaMahasiswa;  
    public double ipk;  
  
    public Mahasiswa(String id, String name, double gpa){  
        nim=id;  
        namaMahasiswa=name;  
        ipk=gpa;  
    }  
  
}
```

2. Buat class baru dengan nama **MahasiswaMain** tetap pada package **minggu5**. Buat class main, kemudian lakukan instansiasi objek array dan isikan datanya. Kemudian tambahkan kode program untuk melakukan pencarian menggunakan Sequential Search.

```

public class MahasiswaMain {
    public static void main(String[] args) {
        Mahasiswa[] mhs = new Mahasiswa[10];
        mhs[0] = new Mahasiswa("19970506001", "Alfatih", 3.9);
        mhs[1] = new Mahasiswa("19910421002", "Gitta", 3.8);
        mhs[2] = new Mahasiswa("19950322002", "Amanda", 3.75);
        mhs[3] = new Mahasiswa("19980129001", "Kevin", 3.5);
        mhs[4] = new Mahasiswa("19990208002", "Windy", 3.92);
        mhs[5] = new Mahasiswa("19970710001", "Belva", 3.6);
        mhs[6] = new Mahasiswa("19920602001", "James", 3.7);
        mhs[7] = new Mahasiswa("19901106002", "Aruna", 3.3);
        mhs[8] = new Mahasiswa("19981005002", "Sabrina", 3.55);
        mhs[9] = new Mahasiswa("19991201001", "Agus", 3.65);

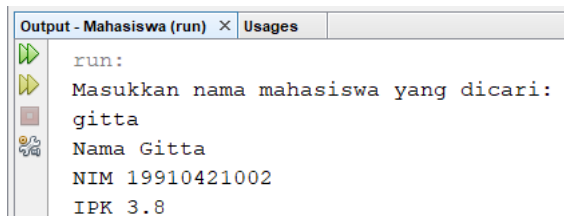
        Scanner sc = new Scanner(System.in);
        System.out.println("Masukkan nama mahasiswa yang dicari: ");
        String cari = sc.nextLine();
        for (int i = 0; i < mhs.length; i++) {
            if (cari.equalsIgnoreCase(mhs[i].namaMahasiswa)) {
                System.out.println("Nama " + mhs[i].namaMahasiswa);
                System.out.println("NIM " + mhs[i].nim);
                System.out.println("IPK " + mhs[i].ipk);
            }
        }
    }
}

```

3. Jalankan (run) class **MahasiswaMain** dan amati hasilnya.

4.2.2 Verifikasi Hasil Percobaan

Samakan hasil compile kode program Anda dengan gambar berikut ini.



4.2.3 Pertanyaan

1. Perhatikan class **MahasiswaMain**, jelaskan fungsi kode program berikut ini!

```
if (cari.equalsIgnoreCase(mhs[i].namaMahasiswa))
```

2. Lakukan modifikasi pada program tersebut, sehingga jika data yang dicari tidak ditemukan, maka akan menampilkan "Data tidak ditemukan"!
3. Lakukan modifikasi pada program tersebut agar pencarian dilakukan bukan berdasarkan nama mahasiswa, melainkan berdasarkan IPK!

4.3 Praktikum 2

Pada percobaan ini, kita akan mengimplementasikan penggunaan ordered sequential search pada array of object.

4.3.1 Langkah-langkah Percobaan

1. Pada package **minggu5**, buat class baru dengan nama **OrderedSearch**. Tambahkan atribut **arr**, serta berikan konstruktor berparameter. Tambahkan pula method **Cari** dan **Tampilkan** seperti gambar berikut ini.

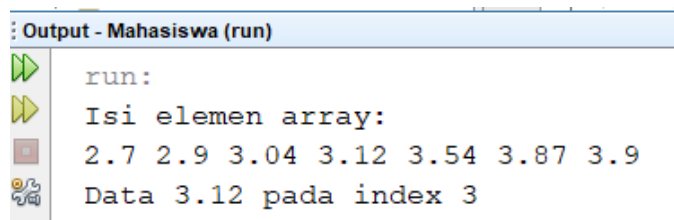
```
public class OrderedSearch {  
    public double[] arr;  
  
    public OrderedSearch(double arrayNilai[]) {  
        arr = new double[arrayNilai.length];  
        for(int i = 0; i<arr.length; i++){  
            arr[i]=arrayNilai[i];  
        }  
    }  
  
    public int cari(double keyword) {  
        int index = -1;  
        for(int i = 0; i<arr.length; i++){  
            if(keyword==arr[i]) {  
                index=i;  
                break;  
            } else {  
                if(keyword<arr[i]) {  
                    break;  
                }  
            }  
        }  
        return index;  
    }  
  
    public void tampilkan() {  
        for(int i=0; i<arr.length;i++){  
            System.out.print(arr[i]+" ");  
        }  
        System.out.println("");  
    }  
}
```

2. Buat class baru dengan nama **PencarianMain** tetap pada package **minggu5**. Buat class main, buat array dan isikan datanya seperti gambar berikut ini.

```
public class PencarianMain {  
    public static void main(String[] args) {  
        double[] data={2.7, 2.9, 3.04, 3.12, 3.54, 3.87, 3.9};  
        OrderedSearch os= new OrderedSearch(data);  
        System.out.println("Isi elemen array:");  
        os.tampilkan();  
        double key=3.12;  
        int index= os.cari(key);  
        if(index!=-1){  
            System.out.println("Data " + key+ "pada index " + index);  
        } else{  
            System.out.println("Data " + key+" tidak ditemukan");  
        }  
    }  
}
```

3. Jalankan (run) class **PencarianMain** dan amati hasilnya.

4.3.2 Verifikasi Hasil Percobaan



```
Output - Mahasiswa (run)  
run:  
Isi elemen array:  
2.7 2.9 3.04 3.12 3.54 3.87 3.9  
Data 3.12 pada index 3
```

4.3.3 Pertanyaan

1. Perhatikan class **PencarianMain**, jelaskan fungsi dari kode kode program berikut ini!
`OrderedSearch os= new OrderedSearch(data);`
2. Perhatikan class **Pencarian**, pada method **Cari** terdapat dua kali **break**. Jelaskan fungsi masing-masing break tersebut pada kode program tersebut!
3. Lakukan modifikasi pada program tersebut sehingga data array dapat bersifat dinamis (dapat diinputkan oleh pengguna)!

4.4 Tugas

1. Terdapat sebuah array satu dimensi sebagai berikut:

Index	0	1	2	3	4	5	6	7	8	9
Array	17	20	26	33	37	41	53	65	73	83

Buatlah program untuk mencari sebuah nilai x dan tentukan jenis sequential search manakah yang optimal dan cocok untuk array diatas dengan:

- a. $x=41$
 - b. $x=50$
2. Terdapat sebuah array dua dimensi yang menggambarkan nama mahasiswa pada 3 kelompok yang berbeda dalam satu kelas. Baris menunjukkan kelompok, sedangkan kolom menunjukkan data mahasiswa dalam satu kelompok dengan contoh seperti di bawah ini.

Urutan		1	2	3	4	5
Kelompok	Indeks	0	1	2	3	4
1	0	"Alfi"	"Safira"	"Tiara"	"Arif"	"Erdi"
2	1	"Yudha"	"Nisa"	"Yulia"	"Fauzan"	"Dwi"
3	2	"Dewa"	"Ana"	"Abdul"	"Dani"	"Ammar"

Berdasarkan data tersebut, buatlah program untuk melakukan pencarian data kelompok dan urutan mahasiswa, dimana data nama mahasiswa yang dicari diinputkan melalui keyboard!

Contoh hasil tampilan dari pencarian seperti di bawah ini

```
Output - Mahasiswa (run)
run:
Daftar Kelompok Mahasiswa
Kelompok 1 :   Alfi   Safira  Tiara   Arif   Erdi
Kelompok 2 :   Yudha  Nisa    Yulia  Fauzan  Dwi
Kelompok 3 :   Dewa   Ana    Abdul  Dani   Ammar
Masukkan nama mahasiswa yang dicari
fauzan
Mahasiswa bernama fauzan berada pada kelompok ke-2 dan urutan ke-4 dalam kelompok
```

3. Terdapat sebuah array satu dimensi sebagai berikut:

Index	0	1	2	3	4	5	6	7	8	9
Array	12	15	6	3	70	51	83	15	3	83

Buatlah program untuk mencari dan mencetak isi array yang nilainya terbesar, serta mencetak di posisi mana saja nilai terbesar tersebut berada!

4. Buatlah program untuk melakukan pencarian data penduduk berdasarkan NIK. Setiap penduduk mempunyai catatan data yang terdiri dari NIK, nama, alamat, dan jenis kelamin. Jumlah penduduk

dan detail data penduduk diinputkan melalui keyboard. Kemudian data yang dicari juga diinputkan melalui keyboard!

```
Output - Searching (run) x
run:
Masukkan jumlah penduduk: 3
Penduduk ke-1
NIK: 175032
Nama: Sandra
Alamat: Surabaya
Jenis Kelamin: Perempuan

Penduduk ke-2
NIK: 174209
Nama: Damar
Alamat: Malang
Jenis Kelamin: Laki-laki

Penduduk ke-3
NIK: 154023
Nama: Rohman
Alamat: Surabaya
Jenis Kelamin: Laki-laki

--- DATA PENDUDUK ---
NO      NIK      NAMA      ALAMAT      JENIS KELAMIN
1       175032   Sandra   Surabaya    Perempuan
2       174209   Damar    Malang      Laki-laki
3       154023   Rohman   Surabaya    Laki-laki

--- CARI DATA PENDUDUK ---
Masukkan data yang dicari: 174209
2       174209   Damar    Malang      Laki-laki
BUILD SUCCESSFUL (total time: 1 minute 15 seconds)
```