

# Jobsheet 6

## Praktikum Algoritma & Struktur Data



Rafi Ody Prasetyo  
(2341720180)

D-IV Teknik Informatika  
Politeknik Negeri Malang  
Semester 2  
2024

## Percobaan 1

### Code:

Buku21.java

```
1 public class Buku21 {
2
3     public int kodeBuku, tahunTerbit, stock;
4     public String judulBuku, pengarang;
5
6     public Buku21(int kodeBuku, String judulBuku, int tahunTerbit, String pengarang, int stock)
7     {
8         this.kodeBuku = kodeBuku;
9         this.judulBuku = judulBuku;
10        this.tahunTerbit = tahunTerbit;
11        this.pengarang = pengarang;
12        this.stock = stock;
13    }
14
15    Codeium: Refactor | Explain | Generate Javadoc | X
16    public void tampilDataBuku()
17    {
18        System.out.println(x:"=====");
19        System.out.println("Kode Buku: " + kodeBuku);
20        System.out.println("Judul Buku: " + judulBuku);
21        System.out.println("Tahun Terbit: " + tahunTerbit);
22        System.out.println("Pengarang: " + pengarang);
23        System.out.println("Stock: " + stock);
24    }
25 }
```

PencarianBuku21.java

```
1 public class PencarianBuku21{
2
3     Buku21 listBk[] = new Buku21[5];
4     int idx;
5
6     Codeium: Refactor | Explain | Generate Javadoc | X
7     public void tambah(Buku21 m)
8     {
9         if (idx < listBk.length) {
10             listBk[idx] = m;
11             idx++;
12         }
13         else {
14             System.out.println(x:"Data sudah penuh!");
15         }
16     }
17
18     Codeium: Refactor | Explain | Generate Javadoc | X
19     public void tampil()
20     {
21         for (Buku21 m : listBk) {
22             m.tampilDataBuku();
23         }
24     }
25 }
```

```

24     public int FindSeqSearch(int cari)
25     {
26         int posisi = 2;
27         for (int i = 0; i < listBk.length; i++) {
28             if (listBk[i].kodeBuku == cari) {
29                 posisi = i;
30                 break;
31             }
32         }
33         return posisi;
34     }
35
36     Codeium: Refactor | Explain | Generate Javadoc | X
37     public void Tampilposisi(int x, int pos)
38     {
39         if (pos != -1) {
40             System.out.println("Data: " + x + " ditemukan pada indeks " + pos);
41         } else {
42             System.out.println("Data: " + x + " tidak ditemukan");
43         }
44     }
45 }

```

```

45     public void tampilData(int x, int pos)
46     {
47         if (pos != -1) {
48             System.out.println("Kode Buku\t: " + x);
49             System.out.println("Judul\t\t: " + listBk[pos].judulBuku);
50             System.out.println("Tahun\t\t: " + listBk[pos].tahunTerbit);
51             System.out.println("Pengarang\t: " + listBk[pos].pengarang);
52             System.out.println("Stock\t\t: " + listBk[pos].stock);
53         } else {
54             System.out.println("Data: " + x + " tidak ditemukan");
55         }
56     }
57
58 }

```

## BukuMain21.java

```
1  import java.util.Scanner;
2
3  Codeium: Refactor | Explain
4  public class BukuMain21 {
5
6      Run | Debug | Codeium: Refactor | Explain | Generate Javadoc | X
7      public static void main(String[] args)
8      {
9          Scanner s = new Scanner(System.in);
10         Scanner s1 = new Scanner(System.in);
11
12         PencarianBuku21 data = new PencarianBuku21();
13         int jumBuku = 5;
14
15         System.out.println(x:"-----");
16         System.out.println(x:"Masukkan data buku secara urut dari KodeBuku terkecil: ");
17         for (int i = 0; i < jumBuku; i++) {
18             System.out.println(x:"-----");
19             System.out.print(s:"Kode Buku \t: ");
20             int kodeBuku = s.nextInt();
21             System.out.print(s:"Judul Buku \t: ");
22             String judulBuku = s1.nextLine();
23             System.out.print(s:"Tahun Terbit \t: ");
24             int tahunTerbit = s.nextInt();
25             System.out.print(s:"Pengarang \t: ");
26             String pengarang = s1.nextLine();
27             System.out.print(s:"Stock \t\t: ");
28             int stock = s.nextInt();
29
30             Buku21 m = new Buku21(kodeBuku, judulBuku, tahunTerbit, pengarang, stock);
31             data.tambah(m);
32         }
33     }
34 }
```

```
32         System.out.println(x:"-----");
33         System.out.println(x:"Data Keseluruhan Mahasiswa: ");
34         System.out.println(x:"-----");
35         System.out.println(x:"Data Keseluruhan Buku: ");
36         data.tampil();
37
38         System.out.println(x:"_____");
39         System.out.println(x:"_____");
40         System.out.println(x:"Pencarian Data: ");
41         System.out.println(x:"Masukkan kode buku yang dicari: ");
42         System.out.print(s:"Kode Buku: ");
43         int cari = s.nextInt();
44         System.out.println(x:"Menggunakan sequential search");
45         int posisi = data.FindSeqSearch(cari);
46         data.Tampilposisi(cari, posisi);
47         data.tampilData(cari, posisi);
48
49         s.close();
50         s1.close();
51     }
52
53 }
```

## Output:

```
Data Keseluruhan Buku:
=====
Kode Buku: 111
Judul Buku: Algoritma
Tahun Terbit: 2019
Pengarang: Wahyuni
Stock: 5
=====
Kode Buku: 123
Judul Buku: Big Data
Tahun Terbit: 2020
Pengarang: Susilo
Stock: 3
=====
Kode Buku: 125
Judul Buku: Desain UI
Tahun Terbit: 2021
Pengarang: Supriadi
Stock: 3
=====
Kode Buku: 126
Judul Buku: Web Programming
Tahun Terbit: 2022
Pengarang: Pustaka Adi
Stock: 2
=====
Kode Buku: 127
Judul Buku: Etika Mahasiswa
Tahun Terbit: 2023
Pengarang: Darmawan Adi
Stock: 2
=====
```

```
-----
Pencarian Data:
Masukkan kode buku yang dicari:
Kode Buku: 126
Menggunakan sequential search
Data: 126 ditemukan pada indeks 3
Kode Buku      : 126
Judul           : Web Programming
Tahun          : 2022
Pengarang      : Pustaka Adi
Stock          : 2
```

## Pertanyaan

1. Jelaskan fungsi break yang ada pada method FindSeqSearch!

### Jawab:

Fungsi break pada method tersebut untuk menghentikan perulangan apabila nilai pada variabel cari sama dengan index dari array listBK pada bagian kode buku.

2. Jika Data Kode Buku yang dimasukkan tidak terurut dari kecil ke besar.

Apakah program masih dapat berjalan? Apakah hasil yang dikeluarkan benar?

Tunjukkan hasil screenshot untuk bukti dengan kode Buku yang acak.

Jelaskan Mengapa hal tersebut bisa terjadi?

### Jawab:

Jika data kode buku dimasukkan tidak terurut dari kecil ke besar, program akan tetap berjalan tetapi hasil pencarian menggunakan metode pencarian sekuensial mungkin tidak akan sesuai dengan yang diharapkan. Alasan utama mengapa hasilnya mungkin

tidak benar adalah karena metode pencarian sekuensial diasumsikan bahwa data telah diurutkan. Jika tidak, pencarian sekuensial tidak akan menemukan hasil yang diinginkan dengan cepat atau bahkan dapat memberikan hasil yang salah jika data terletak di posisi yang diharapkan berdasarkan urutan yang tidak teratur.

```
Data Keseluruhan Buku:
=====
Kode Buku: 103
Judul Buku: Book A
Tahun Terbit: 2000
Pengarang: Author A
Stock: 10
=====
Kode Buku: 102
Judul Buku: Book B
Tahun Terbit: 1998
Pengarang: Author B
Stock: 15
=====
Kode Buku: 105
Judul Buku: Book C
Tahun Terbit: 2005
Pengarang: Author C
Stock: 8
=====
Kode Buku: 101
Judul Buku: Book D
Tahun Terbit: 1995
Pengarang: Author D
Stock: 20
=====
Kode Buku: 104
Judul Buku: Book E
Tahun Terbit: 1995
Pengarang: Author E
Stock: 12
=====
```

```
Pencarian Data:
Masukkan kode buku yang dicari:
Kode Buku: 101
Menggunakan sequential search
Data: 101 ditemukan pada indeks 3
Kode Buku      : 101
Judul          : Book D
Tahun          : 1995
Pengarang      : Author D
Stock          : 20
```

3. Buat method baru dengan nama FindBuku menggunakan konsep sequential search dengan tipe method dari FindBuku adalah BukuNoAbsen. Sehingga Anda bisa memanggil method tersebut pada class BukuMain seperti gambar berikut :

```
Buku dataBuku = data.FindBuku(cari);
dataBuku.tampilDataBuku();
```

**Jawab:**

```
58     public Buku21 findBuku(int kodeBuku)
59     {
60         for (int i = 0; i < listBk.length; i++) {
61             if (listBk[i].kodeBuku == kodeBuku) {
62                 return listBk[i];
63             }
64         }
65         return null;
66     }
```

```
49     //pertanyaan no.3 percobaan 1
50     Buku21 dataBuku21 = data.findBuku(cari);
51     dataBuku21.tampilDataBuku();
```

## Percobaan 2

Code:

PencarianBuku21.java

```
68     public int FindBinarySearch(int cari, int left, int right)
69     {
70         int mid;
71         if (right >= left) {
72             mid = left + (right - left) / 2;
73             if (cari == listBk[mid].kodeBuku) {
74                 return mid;
75             } else if (listBk[mid].kodeBuku > cari) {
76                 return FindBinarySearch(cari, left, mid - 1);
77             } else {
78                 return FindBinarySearch(cari, mid + 1, right);
79             }
80         }
81         return -1;
82     }
```

BukuMain21.java

```
53     System.out.println(x:"_____");
54     System.out.println(x:"Menggunakan Binary Search");
55     posisi = data.FindBinarySearch(cari, left:0, data.listBk.length - 1);
56     data.Tampilposisi(cari, posisi);
57     data.tampilData(cari, posisi);
```

Output:

```
-----
Data Keseluruhan Buku:
=====
Kode Buku: 111
Judul Buku: Algoritma
Tahun Terbit: 2019
Pengarang: Budi
Stock: 2
=====
Kode Buku: 120
Judul Buku: Sejarah
Tahun Terbit: 2020
Pengarang: Alex
Stock: 3
=====
Kode Buku: 210
Judul Buku: Filsafat
Tahun Terbit: 2008
Pengarang: Jalaludin Rumi
Stock: 5
=====
Kode Buku: 300
Judul Buku: Geologi
Tahun Terbit: 2020
Pengarang: Andy F Noya
Stock: 2
=====
Kode Buku: 350
Judul Buku: Sosial Politik
Tahun Terbit: 2005
Pengarang: Rocky Gerung
Stock: 1
-----
-----
```

```
-----
Pencarian Data:
Masukkan kode buku yang dicari:
Kode Buku: 210
Menggunakan sequential search
Data: 210 ditemukan pada indeks 2
Kode Buku      : 210
Judul          : Filsafat
Tahun          : 2008
Pengarang      : Jalaludin Rumi
Stock          : 5
-----
Menggunakan Binary Search
Data: 210 ditemukan pada indeks 2
Kode Buku      : 210
Judul          : Filsafat
Tahun          : 2008
Pengarang      : Jalaludin Rumi
Stock          : 5
-----
```



## Pertanyaan

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

Jawab:

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

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

Jawab:

```
if (cari == listBk[mid].kodeBuku) {  
    return mid;
```

3. Jika data Kode Buku yang dimasukkan tidak urut. Apakah program masih dapat berjalan? Mengapa demikian! Tunjukkan hasil screenshoot untuk bukti dengan kode Buku yang acak. Jelaskan Mengapa hal tersebut bisa terjadi?

Jawab:

Binary search bekerja dengan asumsi bahwa data dalam array terurut secara teratur, misalnya dari yang terkecil hingga yang terbesar. Ini karena algoritma binary search membagi rentang pencarian menjadi dua bagian dan memutuskan di mana harus melanjutkan pencarian berdasarkan perbandingan dengan elemen di posisi tengah.

```
-----  
Pencarian Data:  
Masukkan kode buku yang dicari:  
Kode Buku: 505  
Menggunakan sequential search  
Data: 505 ditemukan pada indeks 4  
Kode Buku      : 505  
Judul          : Desain Grafis  
Tahun         : 2020  
Pengarang     : Necha  
Stock         : 5  
-----  
Menggunakan Binary Search  
Data: 505 tidak ditemukan  
Data: 505 tidak ditemukan
```

4. Jika Kode Buku yang dimasukkan dari Kode Buku terbesar ke terkecil (missal : 20215, 20214, 20212, 20211, 20210) dan elemen yang dicari adalah 20210.

Bagaimana hasil dari binary search? Apakah sesuai? Jika tidak sesuai maka ubahlah kode program binary seach agar hasilnya sesuai!

Jawab:

Binary search pada kondisi ini tidak dapat mencari data pada kondisi data descending. Agar data dapat ditemukan kita harus mengubah kondisi pada else if, dari yang sebelumnya  $>$  menjadi  $<$ .

Before:

```
if (right >= left) {
    mid = left + (right - left) / 2;
    if (cari == listBk[mid].kodeBuku) {
        return mid;
    } else if ([listBk[mid].kodeBuku > cari]) {
        return FindBinarySearch(cari, left, mid - 1);
    } else {
        return FindBinarySearch(cari, mid + 1, right);
    }
}
```

Output:

```
-----
Pencarian Data:
Masukkan kode buku yang dicari:
Kode Buku: 20210
Menggunakan sequential search
Data: 20210 ditemukan pada indeks 4
Kode Buku      : 20210
Judul          : Jurassic Park
Tahun          : 2007
Pengarang      : Christoper Nolan
Stock          : 2
-----
Menggunakan Binary Search
Data: 20210 tidak ditemukan
Data: 20210 tidak ditemukan
```

After:

```
if (right >= left) {
    mid = left + (right - left) / 2;
    if (cari == listBk[mid].kodeBuku) {
        return mid;
    } else if ([listBk[mid].kodeBuku < cari]) {
        return FindBinarySearch(cari, left, mid - 1);
    } else {
        return FindBinarySearch(cari, mid + 1, right);
    }
}
```

Output:

```
-----
Pencarian Data:
Masukkan kode buku yang dicari:
Kode Buku: 20210
Menggunakan sequential search
Data: 20210 ditemukan pada indeks 4
Kode Buku      : 20210
Judul          : Fisika
Tahun          : 2010
Pengarang      : Kustono
Stock          : 4
-----
Menggunakan Binary Search
Data: 20210 ditemukan pada indeks 4
Kode Buku      : 20210
Judul          : Fisika
Tahun          : 2010
Pengarang      : Kustono
Stock          : 4
```

## Percobaan 3

### Code:

MergeSorting21.java

```
1 public class MergeSorting21 {
2
3     Codeium: Refactor | Explain | Generate Javadoc | X
4     public int[] mergeSort(int[] data)
5     {
6         return sort(data, left:0, data.length - 1);
7     }
8
9     Codeium: Refactor | Explain | Generate Javadoc | X
10    public int[] merge(int data[], int left, int middle,int right)
11    {
12        int[] temp = new int[data.length];
13        for (int i = left; i <= right; i++) {
14            temp[i] = data[i];
15        }
16        int a = left;
17        int b = middle + 1;
18        int c = left;
19
20        while (a <= middle && b <= right) {
21            if (temp[a] <= temp[b]) {
22                data[c] = temp[a];
23                a++;
24            } else {
25                data[c] = temp[b];
26                b++;
27            }
28            c++;
29        }
30    }
```

```
28    int s = middle - a;
29    for (int i = 0; i <= s; i++) {
30        data[c + i] = temp[a + i];
31    }
32    return data;
33 }
34
35 Codeium: Refactor | Explain | Generate Javadoc | X
36 public int[] sort(int data[], int left, int right)
37 {
38     if (left < right) {
39         int middle = (left + right) / 2;
40         sort(data, left, middle);
41         sort(data, middle + 1, right);
42         return merge(data, left, middle, right);
43     }
44     return data;
45 }
46
47 Codeium: Refactor | Explain | Generate Javadoc | X
48 public void printArray(int arr[])
49 {
50     for (int i = 0; i < arr.length; i++) {
51         System.out.println(arr[i] + " ");
52     }
53     System.out.println();
54 }
```

## MergeSortingMain21.java

```
2 public class MergeSortingMain21 {  
3  
4     Run | Debug | Codeium: Refactor | Explain | Generate Javadoc | X  
5     public static void main(String[] args) {  
6         int data[] = {10, 40, 30, 50, 70, 20, 100, 90};  
7         System.out.println(x:"Sorting dengan merge sort");  
8         MergeSorting21 mSort = new MergeSorting21();  
9         mSort.printArray(data);  
10        mSort.mergeSort(data);  
11        System.out.println(x:"Setelah diurutkan");  
12        mSort.printArray(data);  
13    }  
14 }
```

Output:

```
Sorting dengan merge sort  
10  
40  
30  
50  
70  
20  
100  
90  
  
Setelah diurutkan  
10  
20  
30  
40  
50  
70  
90  
100
```

## Praktikum

**Github:** <https://github.com/rafiody16/Praktikum-Algoritma-dan-Struktur-Data--smt-2-/tree/main/Jobsheet%206>

```
-----  
Data Kode Buku Sesudah dilakukan sorting:
```

```
=====  
Kode Buku: 23D  
Judul Buku: Filsafat  
Tahun Terbit: 2004  
Pengarang: Enzo Ali  
Stock: 250
```

```
=====  
Kode Buku: 15F  
Judul Buku: Fisika  
Tahun Terbit: 2005  
Pengarang: Reynaldi  
Stock: 101
```

```
=====  
Kode Buku: 12A  
Judul Buku: Geologi  
Tahun Terbit: 2019  
Pengarang: Simanjuntak  
Stock: 120
```

```
=====  
Kode Buku: 10B  
Judul Buku: Biologi  
Tahun Terbit: 2005  
Pengarang: Lucas Enembe  
Stock: 105
```

```
=====  
Kode Buku: 47E  
Judul Buku: Logaritma  
Tahun Terbit: 2004  
Pengarang: I Made Andika  
Stock: 210
```

```
-----  
Kode Buku: 10B  
Menggunakan sequential search  
Data: 10B ditemukan pada indeks 3  
Kode Buku      : 10B  
Judul          : Biologi  
Tahun          : 2005  
Pengarang      : Lucas Enembe  
Stock          : 105
```

```
-----  
Menggunakan Binary Search  
Data: 10B ditemukan pada indeks 3  
Kode Buku      : 10B  
Judul          : Biologi  
Tahun          : 2005  
Pengarang      : Lucas Enembe  
Stock          : 105
```

-----  
Data Judul Buku Sesudah dilakukan sorting:

=====

Kode Buku: 10B

Judul Buku: Biologi

Tahun Terbit: 2005

Pengarang: Lucas Enembe

Stock: 105

=====

Kode Buku: 23D

Judul Buku: Filsafat

Tahun Terbit: 2004

Pengarang: Enzo Ali

Stock: 250

=====

Kode Buku: 15F

Judul Buku: Fisika

Tahun Terbit: 2005

Pengarang: Reynaldi

Stock: 101

=====

Kode Buku: 12A

Judul Buku: Geologi

Tahun Terbit: 2019

Pengarang: Simanjuntak

Stock: 120

=====

Kode Buku: 47E

Judul Buku: Logaritma

Tahun Terbit: 2004

Pengarang: I Made Andika

Stock: 210

-----  
Judul Buku: Geologi

Menggunakan sequential search

Data: Geologi ditemukan pada indeks 2

Kode Buku : Geologi

Judul : Fisika

Tahun : 2005

Pengarang : Reynaldi

Stock : 101

-----  
Menggunakan Binary Search

Data: Geologi ditemukan pada indeks 2

Kode Buku : Geologi

Judul : Fisika

Tahun : 2005

Pengarang : Reynaldi

Stock : 101

