NAMA : Alexander Agung Raya
NIM : 2341720040
NO ABSEN : 03
KELAS : 1F
MATERI : SEARCHING

LAPORAN PRAKTIKUM ALGORITMA DAN STRUKTUR DATA

**FILE NAME =03_Alexander Agung Raya _1F_P7*

* *Pertemuan 7*

SEARCHING

Percobaan 1 :

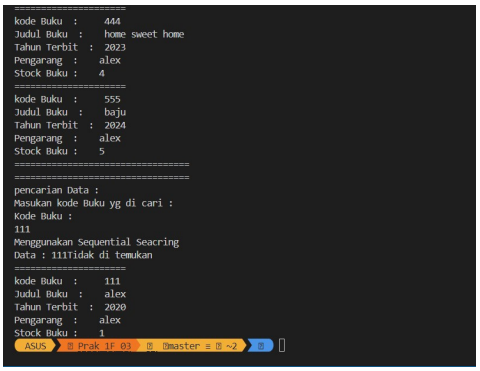
Buku03 :

Bukumain03 :

PercarianBuku03 :

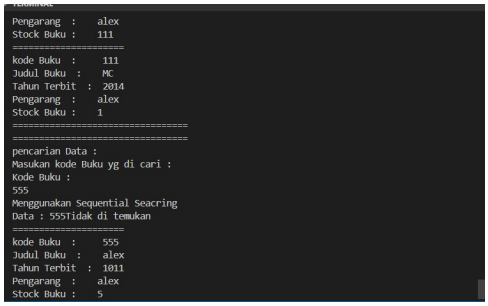
NAMA : Alexander Agung Raya
NIM : 2341720040
NO ABSEN : 03
KELAS : 1F
MATERI : SEARCHING

Hasil :



Pertayaan :

1. Jelaskan fungsi break yang ada pada method FindSeqSearch!
   Jawab : Break di gunakan untuk menghentikan suatu perualangan jika tidak di break akan akan terus di ulang
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 screenshoot untuk bukti dengan kode Buku yang acak. Jelaskan Mengapa hal tersebut bisa terjadi?
   Jawab :
   Bisa Hasil juga akan tetap di jalankan



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 :

NAMA : Alexander Agung Raya
NIM : 2341720040
NO ABSEN : 03
KELAS : 1F
MATERI : SEARCHING

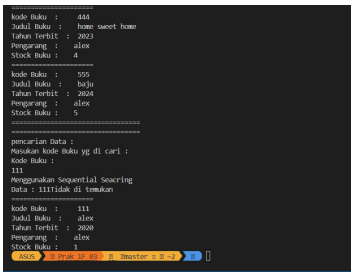
```
Pengarang  :   alex
Stock Buku :   111
====================
kode Buku  :    111
Judul Buku  :   MC
Tahun Terbit  :  2014
Pengarang  :   alex
Stock Buku :   1
==============================
==============================
pencarian Data :
Masukan kode Buku yg di cari :
Kode Buku :
555
Menggunakan Sequential Seacring
Data : 555Tidak di temukan
====================
kode Buku  :    555
Judul Buku  :   alex
Tahun Terbit  :  1011
Pengarang  :   alex
Stock Buku :   5
```

Bukumain03 :

pencariianncaribuku03 :

NAMA : Alexander Agung Raya
NIM : 2341720040
NO ABSEN : 03
KELAS : 1F
MATERI : SEARCHING

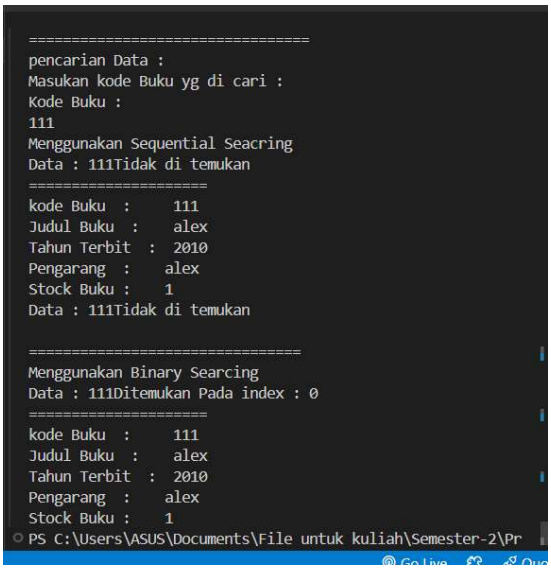
Hasil :



Percobaan 2 :

NAMA : Alexander Agung Raya
NIM : 2341720040
NO ABSEN : 03
KELAS : 1F
MATERI : SEARCHING

Hasil :



Pertayaan :

1. Tunjukkan pada kode program yang mana proses divide dijalankan!
   Jawab :
   ```
   int mid;
        if (right >= left) {
           mid = (right) /2;
   ```
2. Tunjukkan pada kode program yang mana proses conquer dijalankan!
   Jawab :
   ```
   else if (listBK[mid].KodeBuku > cari) {
     return FindBinarySearch(cari, left, mid - 1);
   } else {
     return FindBinarySearch(cari, mid + 1, right);
   }
   ```
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 :
   Jika data Kode Buku yang dimasukkan tidak urut, program masih dapat berjalan, namun hasil pencarian dengan metode binary search mungkin tidak akurat

NAMA : Alexander Agung Raya
NIM : 2341720040
NO ABSEN : 03
KELAS : 1F
MATERI : SEARCHING

```
==================
kode Buku  :    555
Judul Buku  :   alex
Tahun Terbit  : 2011
Pengarang  :   alex
Stock Buku :    5
==================
kode Buku  :    777
Judul Buku  :   alex
Tahun Terbit  : 2012
Pengarang  :   7
Stock Buku :    7
==================
kode Buku  :    222
Judul Buku  :   alex
Tahun Terbit  : 2019
Pengarang  :   alex
Stock Buku :    2
==================
kode Buku  :    999
Judul Buku  :   alex
Tahun Terbit  : 2020
Pengarang  :   alex
Stock Buku :    8
==============================
==============================
pencarian Data :
Masukan kode Buku yg di cari :
Kode Buku :
777
Menggunakan Sequential Seacring
Data : 777Tidak di temukan
==================
kode Buku  :    777
Judul Buku  :   alex
Tahun Terbit  : 2012
Pengarang  :   7
Stock Buku :    7
Data : 777Tidak di temukan
```

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 :

Jika kode buku yang dimasukkan dari yang terbesar ke terkecil dan kita mencari elemen dengan kode buku 20210 menggunakan binary search, hasilnya akan sesuai. Hal ini karena binary search berfungsi dengan baik saat data dalam keadaan terurut.

```
Kode Buku :
20210
Menggunakan Sequential Seacring
Data : 20210Tidak di temukan
======================
kode Buku  :    20210
Judul Buku  :   kabel
Tahun Terbit  : 2080
Pengarang  :   alex
Stock Buku :    10
Data : 20210Tidak di temukan

==============================
Menggunakan Binary Searcing
Data : 20210Tidak di temukan
==================
kode Buku  :    20210
Judul Buku  :   kabel
Tahun Terbit  : 2080
Pengarang  :   alex
Stock Buku :    10
```

NAMA : Alexander Agung Raya
NIM : 2341720040
NO ABSEN : 03
KELAS : 1F
MATERI : SEARCHING

Percobaan 3 :

margeSorting :

```java
package P7.MergeSortTest;

public class MergeSorting03 {
  public void MergeSort(int[] data) {
    if (data == null || data.length <= -1) {
      return;
    }
    Sort(data, 0, data.length - 1);
  }
  public void Merge(int data [], int left, int middle, int right) {
        int[] temp = new int[data.length];
        for(int i = left; i <= right; i++) {
            temp[i] = data[i];
        }
        int a = left;
        int b = middle +1;
        int c = left;

        while (a <= middle && b <= right) {
            if (temp[a] <= temp[b]) {
                data[c] = temp[a];
                a++;
            } else {
                data[c] = temp [b];
                b++;
            }
            c++;
        }
        int s = middle-a;
        for (int i = 0; i <= s; i++) {
            data[c + i] = temp[a + i];
        }
    }
    public void Sort(int data [], int left, int right){
        if (left < right) {
            int middle = (left + right)/2;
            Sort(data, left, middle);
            Sort(data, middle + 1, right);
            Merge(data, left, middle, right);
        }
    }
    public void printArray(int arr[]) {
        for (int i = 0; i < arr.length; i++) {
            System.out.println(arr[i] + " ");
        }
        System.out.println();
    }
}

```

MargeSortingMain :

```java
package P7.MergeSortTest;

public class MergeSortMain03 {
    public static void main(String[] args) {
        int data [] = {10,40,30,50,70,20,100,90};
        System.out.println("sorting dan merge Sort ");
        MergeSorting03 mSort = new MergeSorting03();
        System.out.println("Data Awal");
        mSort.printArray(data);
        mSort.MergeSort(data);
        System.out.println("Setelah Di urutkan :");
        mSort.printArray(data);
    }
}

```

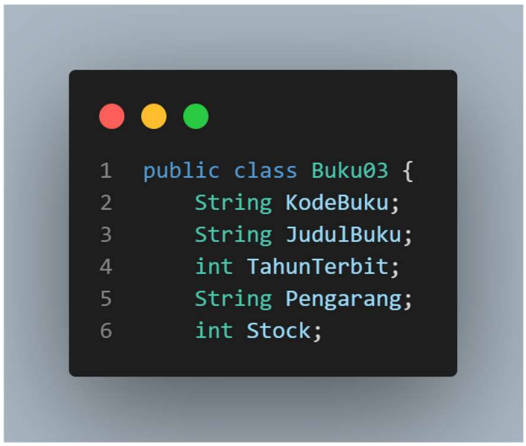
NAMA : Alexander Agung Raya
NIM : 2341720040
NO ABSEN : 03
KELAS : 1F
MATERI : SEARCHING

Hasil :

Latihan :

1. Modifikasi percobaan searching diatas dengan ketentuan berikut ini
   - Ubah tipe data dari kode Buku yang awalnya int menjadi String
   - Tambahkan method untuk pencarian kode Buku (bertipe data String) dengan menggunakan sequential search dan binary search.

   Jawab :



```
1  public class Buku03 {
2      String KodeBuku;
3      String JudulBuku;
4      int TahunTerbit;
5      String Pengarang;
6      int Stock;
```

NAMA : Alexander Agung Raya
NIM : 2341720040
NO ABSEN : 03
KELAS : 1F
MATERI : SEARCHING

```java
public BukuB3(String KodeBuku,String JudulBuku,int TahunTerbit, String Pengarang, int Stock) {
    this.KodeBuku = KodeBuku;
    this.JudulBuku = JudulBuku;
    this.TahunTerbit = TahunTerbit;
    this.Pengarang = Pengarang;
    this.Stock = Stock;
}
```

```java
public int FindBinarySearch(String cari, int left, int right) {
    if (right >= left) {
        int mid = left + (right - left) / 2;
        if (cari.equalsIgnoreCase(listBK[mid].KodeBuku)) {
            return (mid);
        }
        else if (Integer.valueOf(cari)>Integer.valueOf(listBK[mid].KodeBuku)) {
            return FindBinarySearch(cari, 0, mid - 1);
        } else if (Integer.valueOf(cari)<Integer.valueOf(listBK[mid].KodeBuku)){
            return FindBinarySearch(cari, mid + 1, right);
        }
    }
    return -1;
}
```

```java
public int FindSeqSearch(String cari) {
        int posisi = -1;
        for(int j = 0; j < listBK.length; j++) {
            if (listBK[j].KodeBuku == cari) {
                posisi = j;
                break;
            }
        }
        return posisi;
    }
```

2. Modifikasi percobaan searching diatas dengan ketentuan berikut ini
   - Tambahkan method pencarian judul buku menggunakan sequential search dan binary search. Sebelum dilakukan searching dengan binary search data harus dilakukan pengurutan dengan menggunakan algoritma Sorting (bebas pilih algoritma sorting apapun)! Sehingga ketika input data acak, maka algoritma searching akan tetap berjalan
   - Buat aturan untuk mendeteksi hasil pencarian judul buku yang lebih dari 1 hasil dalam bentuk kalimat peringatan! Pastikan algoritma yang diterapkan sesuai dengan kasus yang diberikan!

   Jawab :

NAMA : Alexander Agung Raya
NIM : 2341720040
NO ABSEN : 03
KELAS : 1F
MATERI : SEARCHING

Hasil :