

LAPORAN PRAKTIKUM MINGGU KE-16
“Collection”



Disusun oleh:
Daffa Aqila Rahmatullah
2041720098

D4 TEKNIK INFORMATIKA
TEKNOLOGI INFORMASI
POLITEKNIK NEGERI MALANG
2021

LAPORAN

A. KODE PROGRAM

Kode program dipaste di sini menggunakan font yang berbeda, misalkan courier new.

Kode program jangan discreen shoot agar lebih jelas.

1. PRAKTIKUM 16.2.1 (Percobaan 1)

- **Class ContohList:**

```
import java.util.ArrayList;
import java.util.LinkedList;
import java.util.List;

public class ContohList {
    public static void main(String[] args) {
        List l = new ArrayList();
        l.add(1);
        l.add(2);
        l.add(3);
        l.add("Cireng");
        System.out.printf("Elemen 0: %d total elemen: %d
elemen terakhir: %s\n",
            l.get(0), l.size(), l.get(l.size()-1));

        l.add(4);
        l.remove(0);
        System.out.printf("Elemen 0: %d total elemen: %d
elemen terakhir: %s\n",
            l.get(0), l.size(), l.get(l.size()-1));

        List<String> names = new LinkedList<>();
        names.add("Noureen");
        names.add("Akhleema");
        names.add("Shannum");
        names.add("Uwais");
        names.add("Al-Qarni");

        System.out.printf("Elemen 0: %s total elemen: %s
elemen terakhir: %s\n",
            names.get(0), names.size(), names.get(names
.size()-1));
        names.set(0, "my kid");
        System.out.printf("Elemen 0: %s total elemen: %s
elemen terakhir: %s\n",
            names.get(0), names.size(), names.get(names
.size()-1));
        System.out.println("Names: "+names.toString());
    }
}
```

```
}
```

2. PRAKTIKUM 16.3.1 (Percobaan 2)

Class LoopCollection:

```
import java.util.Iterator;
import java.util.Stack;
public class LoopCollection {
    public static void main(String[] args) throws Exception {
        Stack<String> fruits = new Stack<>();
        fruits.push("Banana");
        fruits.add("Orange");
        fruits.add("Watermelon");
        fruits.add("Leci");
        fruits.push("Salak");

        for(String fruit : fruits){
            System.out.printf("%s ", fruit);
        }
        System.out.println("\n"+ fruits.toString());

        while(!fruits.empty()){
            System.out.printf("%s ", fruits.pop());
        }
        fruits.push("Melon");
        fruits.push("Durian");
        System.out.println("");
        for(Iterator<String> it = fruits.iterator();
it.hasNext();) {
            String fruit = it.next();
            System.out.printf("%s ", fruit);
        }
        System.out.println("");
        fruits.stream().forEach(e -> {
            System.out.printf("%s ", e);
        });
        System.out.println("");
        for(int i =0;i < fruits.size();i++){
            System.out.printf("%s ", fruits.get(i));
        }

    }
}
```

3. PRAKTIKUM 16.4.1 (Percobaan 3)

Class LoopCollection:

```
package Praktikum3;

/**
 *
 * @author ACHE
 */
public class Mahasiswa {
    String nim;
    String nama;
    String notelp;

    public Mahasiswa() {

    }

    public Mahasiswa(String nim, String nama, String
notelp) {
        this.nim = nim;
        this.nama = nama;
        this.notelp = notelp;
    }

    @Override
    public String toString() {
        return "Mahasiswa{" + "nim=" + nim + ", nama=" +
nama + ", notelp=" + notelp + '}';
    }
}
```

Class List Mahasiswa

```
package Praktikum3;

import java.util.ArrayList;
import java.util.Arrays;
```

```

import java.util.List;

/**
 *
 * @author ACHE
 */
public class ListMahasiswa {
    List<Mahasiswa> mahasiswas = new ArrayList<>();
    public void tambah(Mahasiswa...mahasiswa){
        mahasiswas.addAll(Arrays.asList(mahasiswa));
    }
    public void hapus(int index){
        mahasiswas.remove(index);
    }
    public void update(int index,Mahasiswa mhs){
        mahasiswas.set(index, mhs);
    }
    public void tampil(){
        mahasiswas.stream().forEach(mhs ->{
            System.out.println("" + mhs.toString());
        });
    }
    int LinierSearch(String nim){
        for (int i = 0; i < mahasiswas.size(); i++) {
            if (nim.equals(mahasiswas.get(i).nim)) {
                return i;
            }
        }
        return -1;
    }
    public static void main(String[] args) {
        ListMahasiswa lm = new ListMahasiswa();
        Mahasiswa m = new Mahasiswa("201234", "Noureen"
, "021xx1");
    }
}

```

```

        Mahasiswa m1 = new Mahasiswa("201235",
"Akhleema" , "021xx2");

        Mahasiswa m2 = new Mahasiswa("201236",
"Shannum" , "021xx3");

//menambahkan objek mahasiswa
lm.tambah(m, m1, m2);

lm.tampil();

lm.update(lm.LinierSearch("201235"), new
Mahasiswa("201235","Akhleema lela","021xx2"));

System.out.println("");

lm.tampil();

    }

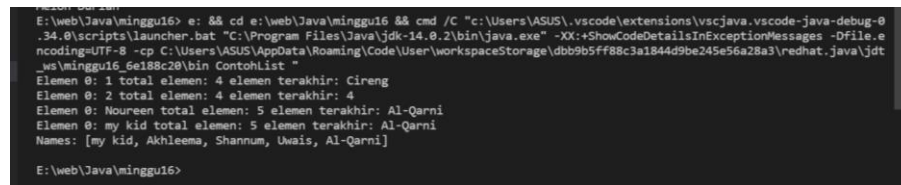
}

```

B. OUTPUT PROGRAM

Untuk hasil program silakan dilakukan screen shoot dengan tetap memperhatikan ukuran gambar agar dapat terlihat dengan jelas.

1. Praktikum 1

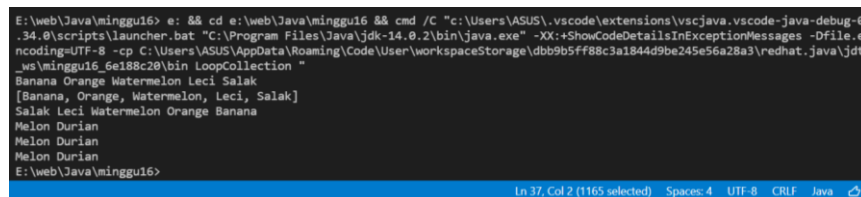


```

E:\web\Java\minggu16> e: && cd e:\web\Java\minggu16 && cmd /C "c:\Users\ASUS\.vscode\extensions\vscjava.vscode-java-debug-0
.34.0\scripts\launcher.bat "C:\Program Files\Java\jdk-14.0.2\bin\java.exe" -XX:+ShowCodeDetailsInExceptionMessages -Dfile.e
ncoding=UTF-8 -cp C:\Users\ASUS\AppData\Roaming\Code\User\workspaceStorage\dbb9b5ff88c3a1844d9be245e56a28a3\redhat.java\jdt
_ws\minggu16_6e188c20\bin\ContohList "
Elemen 0: 1 total elemen: 4 elemen terakhir: Cireng
Elemen 0: 2 total elemen: 4 elemen terakhir: 4
Elemen 0: Noreen total elemen: 5 elemen terakhir: Al-Qarni
Elemen 0: my kid total elemen: 5 elemen terakhir: Al-Qarni
Names: [my kid, Akhleema, Shannum, Uwais, Al-Qarni]
E:\web\Java\minggu16>

```

2. Praktikum 2

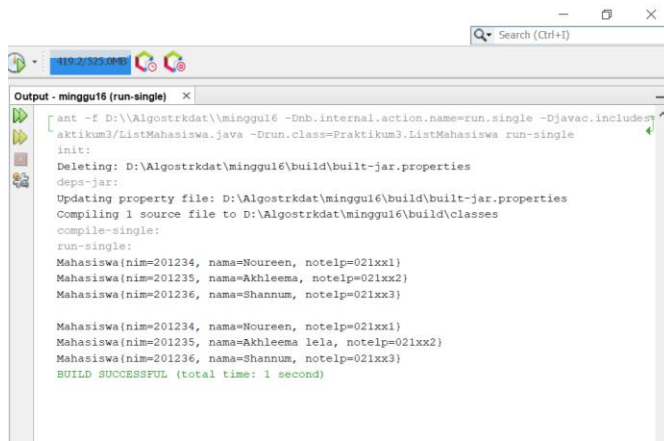


```

E:\web\Java\minggu16> e: && cd e:\web\Java\minggu16 && cmd /C "c:\Users\ASUS\.vscode\extensions\vscjava.vscode-java-debug-0
.34.0\scripts\launcher.bat "C:\Program Files\Java\jdk-14.0.2\bin\java.exe" -XX:+ShowCodeDetailsInExceptionMessages -Dfile.e
ncoding=UTF-8 -cp C:\Users\ASUS\AppData\Roaming\Code\User\workspaceStorage\dbb9b5ff88c3a1844d9be245e56a28a3\redhat.java\jdt
_ws\minggu16_6e188c20\bin\LoopCollection "
Banana Orange Watermelon Leci Salak
[Banana, Orange, Watermelon, Leci, Salak]
Salak Leci Watermelon Orange Banana
Melon Durian
Melon Durian
Melon Durian
E:\web\Java\minggu16>

```

3. Praktikum 3



C. PENJELASAN

Silakan diberikan penjelasan kode program yang Anda buat, potongan-potongan program yang menurut Anda penting.

13.2.2 Pertanyaan Percobaan

1. Perhatikan baris kode 25-36, mengapa semua jenis data bisa ditampung ke dalam sebuah ArrayList?

Jawaban: Karena dalam tipe data tersebut tidak ada kurung siku yg menyatakan tipe data apa yang bisa disimpan, sehingga bisa menyimpan atau menampung semua jenis data.

2. Modifikasi baris kode 25-36 seingga data yang ditampung hanya satu jenis atau spesifik tipe tertentu!

Jawaban:

```
import java.util.ArrayList;
import java.util.LinkedList;
import java.util.List;

public class ContohList {
    public static void main(String[] args) {
        List<String> l = new ArrayList<>();
        List<Integer> d = new ArrayList<>();
        d.add(1);
        d.add(2);
        d.add(3);
        l.add("Cireng");
        System.out.printf("Elemen 0: %d total elemen: %d
elemen terakhir: %s\n",
        d.get(0), d.size(), l.get(l.size() -
1));
        d.add(4);
```

```

        d.remove(0);
        System.out.printf("Elemen 0: %d total elemen: %d
elemen terakhir: %s\n",
        l.get(0), l.size(), l.get(l.size() - 1));
        LinkedList<String> names = new
        LinkedList<>();
        l.add("Noureen");
        l.add("Akhleema");
        l.add("Shannum");
        l.add("Uwais");
        l.add("Al-Qarni");
        System.out.printf("Elemen 0: %d total elemen: %d
elemen terakhir: %s\n",
        d.get(0), d.size(), d.get(d.size() -
1));
        l.set(0, "My kid");
        System.out.printf("Elemen 0: %s total elemen: %s
elemen terakhir: %s\n",
        l.get(0), l.size(), l.get(l.size() -
1));
    }
}

```

Hasil:

```

e:\web\Java\minggu16\Praktikum2>cd "e:\web\Java\minggu16\Praktikum2\" && javac ContohList.java && java ContohList
Note: ContohList.java uses unchecked or unsafe operations.
Note: Recompile with -Xlint:unchecked for details.
Elemen 0: 1 total elemen: 4 elemen terakhir: Cireng
Elemen 0: 2 total elemen: 4 elemen terakhir: 4
Elemen 0: Noureen total elemen: 5 elemen terakhir: Al-Qarni
Elemen 0: my kid total elemen: 5 elemen terakhir: Al-Qarni
Names: [my kid, Akhleema, Shannum, Uwais, Al-Qarni]
e:\web\Java\minggu16\Praktikum2>

```

- Ubah kode pada baris kode 38 menjadi seperti ini

Jawaban:

```

1  import java.util.ArrayList;
2  import java.util.LinkedList;
3  import java.util.List;
4
5  public class ContohList {
6      public static void main(String[] args) {
7          List l = new ArrayList();
8          l.add(1);
9          l.add(2);
10         l.add(3);
11         l.add("Cireng");
12         System.out.printf("Elemen 0: %d total elemen: %d elemen terakhir:
13         l.get(0), l.size(), l.get(l.size()-1));
14
15         l.add(4);
16         l.remove(0);
17         System.out.printf("Elemen 0: %d total elemen: %d elemen terakhir:
18         l.get(0), l.size(), l.get(l.size()-1));
19
20         LinkedList<String> names = new LinkedList<>();
21         names.add("Noureen");
22         names.add("Akhleema");
23         names.add("Shannum");
24         names.add("Uwais");
25         names.add("Al-Qarni");
26
27         System.out.printf("Elemen 0: %s total elemen: %s elemen terakhir:
28         names.get(0), names.size(), names.get(names.size()-1));
29         names.set(0, "my kid");

```



```

30      System.out.printf("Elemen 0: %s total elemen: %s elemen terakhir:
31          names.get(0),names.size(),names.get(names.size()-1));
32      System.out.println("Names: "+names.toString());
33
34  }

```

Hasil:

```

e: && cd e:\web\Java\minggu16 && cmd /C "c:\Users\ASUS\.vscode\extensions\vscjava.vscode-java-debug-0.34.0\scripts\launche
r.bat "C:\Program Files\Java\jdk-14.0.2\bin\java.exe" -XX:+ShowCodeDetailsInExceptionMessages -Dfile.encoding=UTF-8 -cp C:\
Users\ASUS\AppData\Roaming\Code\User\workspaceStorage\dbb9b5ff8c3a1844d9be245e56a28a3\redhat.java\jdt_ws\minggu16_6e188c20
\bin ContohList "
Elemen 0: 1 total elemen: 4 elemen terakhir: Cireng
Elemen 0: 2 total elemen: 4 elemen terakhir: 4
Elemen 0: Nourreen total elemen: 5 elemen terakhir: Al-Qarni
Elemen 0: my kid total elemen: 5 elemen terakhir: Al-Qarni
Names: [my kid, Akhleema, Shannum, Uwais, Al-Qarni]

```

4. Tambahkan juga baris berikut ini, untuk memberikan perbedaan dari tampilan yang sebelumnya

Jawaban:

```

import java.util.ArrayList;
import java.util.LinkedList;
import java.util.List;

public class ContohList {
    public static void main(String[] args) {
        ArrayList<Integer> al = new ArrayList();
        al.add(1);
        al.add(2);
        al.add(3);
        al.add(29);
        System.out.printf
        ("Element 0 : %d total elemen : %d elemen terakhir : %
s \n"
            ,al.get(0),al.size(),al.get(al.size() - 1));
        al.add(4);
        al.remove(0);
        System.out.printf
        ("Element 0 : %d total elemen : %d elemen terakhir : %
s \n"
            ,al.get(0),al.size(),al.get(al.size() - 1));
        LinkedList<String> names = new LinkedList<>();
        names.add("Nourreen");
        names.add("Akhleema");
        names.add("Shannum");
        names.add("Uwais");
        names.add("Al-Qarni");

        System.out.printf
        ("Element 0 : %s total elemen : %s elemen terakhir : %
s \n",names.get(0),names.size(),names.get(names.size() - 1));
        names.set(0, "My Kid");
        System.out.printf
        ("Element 0 : %s total elemen : %s elemen terakhir : %
s \n",names.get(0),names.size(),names.get(names.size() - 1));

```

```

        System.out.println("Names : "+names.toString());

        names.push("Daffa");
        System.out.printf
        ("Element 0 : %s total elemen : %s elemen terakhir : %
s \n",
        names.get(0),names.size(),names.get(names.size
() - 1));
        System.out.println("Names : "+names.toString());

    }
}

```

5. Dari penambahan kode tersebut, silakan dijalankan dan apakah yang dapat Anda jelaskan!

Jawaban: penambahan kode program tersebut menambahkan data “Daffa” pada baris pertama atau pada index-0 dan menambah size karena penambahan data tersebut, memanggil elemen terakhir dengan perintah getLast().

13.3.2 Pertanyaan Percobaan

1. Apakah perbedaan fungsi push() dan add() pada objek fruits?

Jawaban: push() merupakan method asli dari Stack dan sedangkan add() method yang berasal dari vector, karena Stack turunan dari Vector maka method Add() dapat digunakan pada Stack. Dan kedua nya sama-sama menambahkan data dari index pertama atau index ke-0

2. Silakan hilangkan baris 43 dan 44, apakah yang akan terjadi? Mengapa bisa demikian?

Jawaban:

```

praktikum1 > LoopCollection.java > LoopCollection > main(String[])
12     for(String fruit : fruits){
13         System.out.printf("%s ", fruit);
14     }
15     System.out.println("\n"+ fruits.toString());
16
17     while(!fruits.empty()){
18         System.out.printf("%s ", fruits.pop());
19     }
20     // fruits.push("Melon");
21     // fruits.push("Durian");
22     System.out.println("");
23     for(Iterator<String> it = fruits.iterator(); it.hasNext();){
24         String fruit = it.next();
25         System.out.printf("%s ", fruit);

```

Hasil:

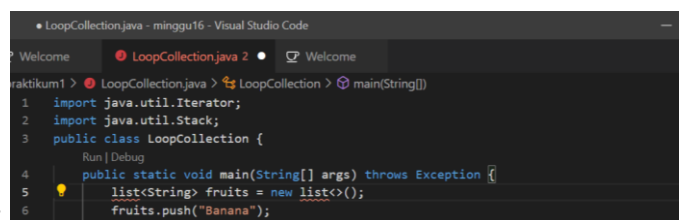
```
e:\web\Java\minggu16\Praktikum2>cd "e:\web\Java\minggu16\praktikum1" && javac LoopCollection.java && java LoopCollection
Banana Orange Watermelon Leci Salak
[Banana, Orange, Watermelon, Leci, Salak]
Salak Leci Watermelon Orange Banana
Melon Durian
Melon Durian
Melon Durian
e:\web\Java\minggu16\praktikum1>
```

Yang terjadi adalah data melon dan durian tidak muncul di output dikarenakan kedua data tersebut tidak dimasukkan atau tidak berada di dalam stack.

3. Jelaskan fungsi dari baris 46-49?

Jawaban: Fungsi dari baris tersebut adalah untuk mengakses dan menampilkan nilai dari seluruh element yang ada pada Stack. Menggunakan method hasNext() untuk mengecek kondisi apakah objek iterator dan masih mempunyai nilai pada elemen selanjutnya atau tidak. Sedangkan menggunakan method .next() untuk menampilkan nilai selanjutnya pada elemen.

4. Silakan ganti baris kode 25, Stack menjadi List dan apakah yang terjadi? Mengapa bisa demikian?



```
LoopCollection.java - minggu16 - Visual Studio Code
Welcome
LoopCollection.java 2
Welcome
praktikum1 > LoopCollection.java > LoopCollection > main(String[])
1 import java.util.Iterator;
2 import java.util.Stack;
3 public class LoopCollection {
4     public static void main(String[] args) throws Exception {
5         list<String> fruits = new list<>();
6         fruits.push("Banana");
```

Jawaban:

Akan error karena disana tidak terdapat method list

5. Ganti elemen terakhir dari dari objek fruits menjadi "Strawberry"!

Jawaban:

```
19     }
20     fruits.push("Melon");
21     fruits.push("Strawberry");
22     System.out.println("");
23     for(Iterator<String> it = fruits.iterator(); it.hasNext();){
24         String fruit = it.next();
25         System.out.printf("%s ", fruit);
26     }
```

Hasil:

```
E:\web\Java\minggu16> e: && cd e:\web\Java\minggu16 && cmd /C "c:\Users\ASUS\.vscode\extensions\vscjava.vscode-java-debug-0.34.0\scripts\launcher.bat "C:\Program Files\Java\jdk-14.0.2\bin\java.exe" -XX:+ShowCodeDetailsInExceptionMessages -Dfile.encoding=UTF-8 -cp C:\Users\ASUS\AppData\Roaming\Code\User\workspaceStorage\dbb9b5ff88c3a1844d9be245e56a28a3\redhat.java\jdt_ws\minggu16_6e188c20\bin LoopCollection "
Banana Orange Watermelon Leci Salak
[Banana, Orange, Watermelon, Leci, Salak]
Salak Leci Watermelon Orange Banana
Melon Strawberry
Melon Strawberry
Melon Strawberry
E:\web\Java\minggu16>
```

16.4.3 Pertanyaan Percobaan

1. Pada fungsi tambah() yang menggunakan unlimited argument itu menggunakan konsep apa? Dan kelebihan nya apa?

Jawab: Kelebihannya yaitu dapat mengambil argument tanpa batas tertentu dan memiliki lebih dari satu parameter.

2. Pada fungsi linearSearch() di atas, silakan diganti dengan fungsi binarySearch() dari collection!

Jawab:

Class Mahasiswa

```
public class Mahasiswa {
    String nim;
    String nama;
    String notelp;

    public Mahasiswa() {

    }

    public Mahasiswa(String nim, String nama, String
notelp) {
        this.nim = nim;
        this.nama = nama;
        this.notelp = notelp;
    }

    @Override
    public String toString() {
        return "Mahasiswa{" + "nim=" + nim + ",
nama=" + nama + ", notelp=" + notelp + '}';
    }
}
```

Class mhs Comparator15

```
import java.util.Comparator;
/**
 *
 * @author ACHE
 */
public class MhsComparator15 implements
Comparator<Mahasiswa>{
    public int compare(Mahasiswa mhs1, Mahasiswa
mhs2) {

        if (mhs1.nim == mhs2.nim) {
            return 0;
        } else {
            return -1;
        }
    }

    @Override
```

```

        public Comparator<Mahasiswa> reversed() {
            return Comparator.super.reversed();
        }
    }

Class ListMahasiswa
    import java.util.ArrayList;
    import java.util.Arrays;
    import java.util.Collections;
    //import java.util.Collections;
    import java.util.List;

    /**
     *
     * @author ACHE
     */
    public class ListMahasiswa {
        List<Mahasiswa> mahasiswas = new ArrayList<>();
        public void tambah(Mahasiswa...mahasiswa){
            mahasiswas.addAll(Arrays.asList(mahasiswa));
        }
        public void hapus(int index){
            mahasiswas.remove(index);
        }
        public void update(int index,Mahasiswa mhs){
            mahasiswas.set(index, mhs);
        }
        public void tampil(){
            mahasiswas.stream().forEach(mhs ->{
                System.out.println("" + mhs.toString());
            });
        }
        int LinierSearch(String nim){
            for (int i = 0; i < mahasiswas.size(); i++)
            {
                if (nim.equals(mahasiswas.get(i).nim)) {
                    return i;
                }
            }
            return -1;
        }
        // void ascendingSort() {
        //     this.mahasiswas.sort((Mahasiswa m1,
        Mahasiswa m2)
        //         -> m1.nama.compareTo(m2.nama));
        // }
        // void descending() {
        //     this.mahasiswas.sort((Mahasiswa m1,
        Mahasiswa m2)
        //         -> m2.nama.compareTo(m1.nama));
        // }
    }

```

```

        public static void main(String[] args) {
            ListMahasiswa lm = new ListMahasiswa();
            Mahasiswa m = new Mahasiswa("201234",
"Noureen" , "021xx1");
            Mahasiswa m1 = new Mahasiswa("201235",
"Akhleema" , "021xx2");
            Mahasiswa m2 = new Mahasiswa("201236",
"Shannum" , "021xx3");
            //menambahkan objek mahasiswa
            lm.tambah(m, m1, m2);
            lm.tampil();
            //            lm.update(lm.LinierSearch("201235"), new
Mahasiswa("201235","Akhleema lela","021xx2"));
            //            System.out.println("");
            //            lm.tampil();
            Mahasiswa key = new Mahasiswa("201235",
null, null);

            lm.update(Collections.binarySearch(lm.mahasiswas, key,
new MhsComparator15()), new Mahasiswa("201235",
"Akhleema", "021xx2"));
            System.out.println("");
            lm.tampil();

```

3. Tambahkan fungsi sorting baik secara ascending ataupun descending pada class tersebut!

Jawab:

Class Mahasiswa

```

    public class Mahasiswa {
        String nim;
        String nama;
        String notelp;

        public Mahasiswa() {

        }

        public Mahasiswa(String nim, String nama, String
notelp) {
            this.nim = nim;
            this.nama = nama;
            this.notelp = notelp;
        }

        @Override
        public String toString() {
            return "Mahasiswa{" + "nim=" + nim + ",
nama=" + nama + ", notelp=" + notelp + '}';
        }
    }

```

```

}
Class mhs Comparator15
import java.util.Comparator;
/**
 *
 * @author ACHE
 */
public class MhsComparator15 implements
Comparator<Mahasiswa>{
    public int compare(Mahasiswa mhs1, Mahasiswa
mhs2) {

        if (mhs1.nim == mhs2.nim) {
            return 0;
        } else {
            return -1;
        }
    }

    @Override
    public Comparator<Mahasiswa> reversed() {
        return Comparator.super.reversed();
    }
}

```

```

Class ListMahasiswa
import java.util.ArrayList;
import java.util.Arrays;
import java.util.Collections;
//import java.util.Collections;
import java.util.List;

/**
 *
 * @author ACHE
 */
public class ListMahasiswa {
    List<Mahasiswa> mahasiswas = new ArrayList<>();
    public void tambah(Mahasiswa...mahasiswa){
        mahasiswas.addAll(Arrays.asList(mahasiswa));
    }
    public void hapus(int index){
        mahasiswas.remove(index);
    }
}

```

```

    }
    public void update(int index,Mahasiswa mhs){
        mahasiswa.set(index, mhs);
    }
    public void tampil(){
        mahasiswa.stream().forEach(mhs ->{
            System.out.println("" + mhs.toString());
        });
    }
    int LinierSearch(String nim){
        for (int i = 0; i < mahasiswa.size(); i++)
        {
            if (nim.equals(mahasiswa.get(i).nim)) {
                return i;
            }
        }
        return -1;
    }
    void ascendingSort() {
        this.mahasiswa.sort((Mahasiswa m1,
Mahasiswa m2)
            -> m1.nama.compareTo(m2.nama));
    }

    void descending() {
        this.mahasiswa.sort((Mahasiswa m1,
Mahasiswa m2)
            -> m2.nama.compareTo(m1.nama));
    }
    public static void main(String[] args) {
        ListMahasiswa lm = new ListMahasiswa();
        Mahasiswa m = new Mahasiswa("201234",
"Noureen" , "021xx1");
    }

```



```

        Mahasiswa m1 = new Mahasiswa("201235",
"Akhleema" , "021xx2");
        Mahasiswa m2 = new Mahasiswa("201236",
"Shannum" , "021xx3");
        //menambahkan objek mahasiswa
        lm.tambah(m, m1, m2);
        lm.tampil();
        lm.update(lm.LinierSearch("201235"), new
Mahasiswa("201235","Akhleema lela","021xx2"));
        System.out.println("");
        lm.tampil();
//        Mahasiswa key = new Mahasiswa("201235",
null, null);
//
//
lm.update(Collections.binarySearch(lm.mahasiswas,
key, new MhsComparator15()), new Mahasiswa("201235",
"Akhleema", "021xx2"));
//        System.out.println("");
//        lm.tampil();

        System.out.println("Ascending : ");
        lm.ascendingSort();
        lm.tampil();

        System.out.println("Descending : ");
        lm.descending();
        lm.tampil();
    }

}

```

D. KESIMPULAN

Dari langkah langkah praktikum diatas saya jadi mengetahui bagaimana cara penerapan algoritma Collection dalam penyelesaian sebuah studi kasus. Juga mengetahui operasi apa saja yang ada dalam algoritma Collection. Sama dengan algoritma yang dipelajari sebelumnya algoritma Collection tidak hanya dapat menyimpan data primitive saja tetapi juga dapat menyimpan data bertipe String dan objek. Collection dapat menyimpan struktur linear maupun non linear. Collection sendiri merupakan implemementasi untuk library scanner pada java.

TUGAS

Tugas diberikan ketika diberikan intruksi untuk mengerjakan tugas, jangan lupa untuk menuliskan semua soalnya sebelum memberikan jawaban.

1. Implementasikan stack menggunakan collection dengan contoh kasus tumpukan buku perpustakaan sesuai dengan fitur-fitur yang ditunjukkan pada gambar di bawah ini!

- Class Buku

```
/*
 * To change this license header, choose License
Headers in Project Properties.
 * To change this template file, choose Tools |
Templates
 * and open the template in the editor.
 */
package Tugas1;

/**
 *
 * @author ACHE
 */
public class Buku {
    //mendeklarasikan atribut
    String isbn;
    String judul;
    String terbit;
    String penerbit;
    //membuat konstruktor default
    public Buku(){
    }
    //membuat konstruktor berparameter
    public Buku(String isbn, String judul, String
    terbit, String penerbit){
        this.isbn = isbn;
        this.judul = judul;
        this.terbit = terbit;
        this.penerbit = penerbit;
    }
    @Override
    public String toString(){
        return "Buku{" + "isbn=" + isbn + ", judul="
+ judul + ", terbit=" +
        terbit + ", penerbit=" + penerbit
+'}';
    }
}
```

- Class BukuMain

```
/*
```

```

    * To change this license header, choose License
    Headers in Project Properties.
    * To change this template file, choose Tools |
    Templates
    * and open the template in the editor.
    */
package Tugas1;

import java.util.Scanner;
import java.util.Stack;

/**
 *
 * @author ACHE
 */
public class BukuMain {

    /**
     * @param args the command line arguments
     */
    public static void main(String[] args) {
        //menginstansiasi objek
        Stack<Buku> buku = new Stack<>();
        Scanner sc = new Scanner(System.in);
        Scanner ab = new Scanner(System.in);
        int menu;
        char a;
        //membuat perulangan menu
        do {

System.out.println("*****");
            System.out.println("Data Buku
Perpustakaan");

System.out.println("*****");
            System.out.println("");
            System.out.println("1. Entry Judul
Buku");
            System.out.println("2. Ambil Buku
Teratas");
            System.out.println("3. Cetak Buku
Teratas");
            System.out.println("4. Info semua Judul
Buku");
            System.out.println("5. Keluar");
            System.out.print("Pilih: ");
            menu = sc.nextInt();
            if (menu == 1) {
                sc.nextLine();
                System.out.print("Isbn\t: ");

```

```

        String isbn = sc.nextLine();
        System.out.print("Judul\t: ");
        String judul = sc.next();
        System.out.print("Terbit\t: ");
        String terbit = ab.nextLine();
        System.out.print("Penerbit: ");
        String penerbit = ab.nextLine();
        Buku b = new Buku();
        buku.push (b = new Buku (isbn,
judul, terbit, penerbit));
        menu = sc.next().charAt(0);
        sc.nextLine();
    }else if (menu == 2){
        buku.pop();
    }else if (menu == 3) {
        System.out.print("Buku{" + "isbn=" +
(buku.get(buku.size() - 1).isbn)
+ ", " + "judul=" +
(buku.get(buku.size() - 1).judul) +
", "
+ "Terbit=" +
(buku.get(buku.size() - 1).terbit) +
", " +
"Penerbit=" +
(buku.get(buku.size() - 1).penerbit)
+ '}');
    }else if (menu == 4) {
        buku.stream().forEach(b -> {
            System.out.println("Buku{" +
"isbn=" + b.isbn + ", " + "judul=" +
b.judul
+ ", " + "terbit=" + b.terbit +
", " + "penerbit=" + b.penerbit +
'}');
        });
    } else {
        System.out.println("Maaf Menu yang anda
pilih tidak tersedia");
    }

    System.out.println("");
    System.out.print("Apakah ingin kembali
ke menu awal (y/n): ");
    a = sc.next().charAt(0);
    } while (a == 'y');
    }
}

```

Hasil:

```
Output - minggu16 (run-single) #2 x
[ ant -f D:\Algostrkdat\minggu16 -Dnb.internal.action.name=run.single -Djavac.includes=T
Drun.class=Tugas1.BukuMain run-single
init:
Deleting: D:\Algostrkdat\minggu16\build\build-jar.properties
deps-jar:
Updating property file: D:\Algostrkdat\minggu16\build\build-jar.properties
Compiling 1 source file to D:\Algostrkdat\minggu16\build\classes
compile-single:
run-single:
*****
Data Buku Perpustakaan
*****

1. Entry Judul Buku
2. Ambil Buku Teratas
3. Cetak Buku Teratas
4. Info semua Judul Buku
5. Keluar
Pilih: 1
Isbn : 202100001
Judul : Java for Dummies
Terbit : 2021
Penerbit: Andi Publisher

Apakah ingin kembali ke menu awal (y/n)y
*****
Data Buku Perpustakaan
*****

1. Entry Judul Buku
2. Ambil Buku Teratas
3. Cetak Buku Teratas
4. Info semua Judul Buku
5. Keluar
Pilih: 1
Isbn : YIC 004
Judul : Getting Started with C++ Audio Programming for Game Developepor
Terbit : 2013
Penerbit: Eu LLC

Apakah ingin kembali ke menu awal (y/n)y
*****
Data Buku Perpustakaan
*****

1. Entry Judul Buku
2. Ambil Buku Teratas
3. Cetak Buku Teratas
4. Info semua Judul Buku
5. Keluar
Pilih: 1
Isbn : YIC 003
Judul : Algoritma Notes For Professionals
Terbit : 2012
Penerbit: Turpis Nec LLC

Apakah ingin kembali ke menu awal (y/n)y
*****
Data Buku Perpustakaan
*****

1. Entry Judul Buku
2. Ambil Buku Teratas
3. Cetak Buku Teratas
4. Info semua Judul Buku
5. Keluar
Pilih: 1
Isbn : YIC 002
Judul : Understanding Software
Terbit : 2011
Penerbit: Est Foundation

ut - minggu16 (run-single) #2 x
Apakah ingin kembali ke menu awal (y/n)y
*****
Data Buku Perpustakaan
*****

1. Entry Judul Buku
2. Ambil Buku Teratas
3. Cetak Buku Teratas
4. Info semua Judul Buku
5. Keluar
Pilih: 1
Isbn : YIC 001
Judul : 3D Computer Vision
Terbit : 2010
Penerbit: Magma PC

Apakah ingin kembali ke menu awal (y/n)y
*****
Data Buku Perpustakaan
*****

1. Entry Judul Buku
2. Ambil Buku Teratas
3. Cetak Buku Teratas
4. Info semua Judul Buku
5. Keluar
Pilih: 4
Buku(isbn=202100001, judul=Java, terbit=2021, penerbit=Andi Publisher)
Buku(isbn=YIC 004, judul=Getting, terbit=2013, penerbit=Eu LLC)
Buku(isbn=YIC 003, judul=Algoritma, terbit=2012, penerbit=Turpis Nec LLC)
Buku(isbn=YIC 002, judul=Understanding, terbit=2011, penerbit=Est Foundation)
Buku(isbn=YIC 001, judul=3D, terbit=2010, penerbit=Magma PC)
```

```

Apakah ingin kembali ke menu awal (y/n)y
*****
Data Buku Perpustakaan
*****

1. Entry Judul Buku
2. Ambil Buku Teratas
3. Cetak Buku Teratas
4. Info semua Judul Buku
5. Keluar
Pilih: 3
Buku(isbn=YIC 001, judul=3D, Terbit=2010, Penerbit=Magma PC)

Apakah ingin kembali ke menu awal (y/n)y
*****
Data Buku Perpustakaan
*****

1. Entry Judul Buku
2. Ambil Buku Teratas
3. Cetak Buku Teratas
4. Info semua Judul Buku
5. Keluar
Pilih: 2
Buku(isbn=202100001, judul=Java, terbit=2021, penerbit=Andi Publisher)
Buku(isbn=YIC 004, judul=Getting, terbit=2013, penerbit=Eu LLC)
Buku(isbn=YIC 003, judul=Algoritma, terbit=2012, penerbit=Turpis Nec LLC)
Buku(isbn=YIC 002, judul=Understanding, terbit=2011, penerbit=Est Foundation)

Apakah ingin kembali ke menu awal (y/n)y
*****
Data Buku Perpustakaan
*****

1. Entry Judul Buku
2. Ambil Buku Teratas
3. Cetak Buku Teratas
4. Info semua Judul Buku
5. Keluar
Pilih: 5
Maaf Menu yang anda pilih tidak tersedia

Apakah ingin kembali ke menu awal (y/n)n
BUILD SUCCESSFUL (total time: 8 minutes 50 seconds)

```

Penjelasan:

Pada program ini menggunakan implementasi stack dengan menggunakan collection yang ditandai dengan import java.util.Stack;

2. Buatlah implementasi program daftar nilai mahasiswa semester, minimal memiliki 3 class yaitu Mahasiswa, Nilai, dan Mata Kuliah. Khusus untuk data Mahasiswa dan Mata Kuliah harus sudah diinisialisasi, sehingga ketika memasukkan data nilai data mahasiswa dan mata kuliah cukup menginputkan nim ataupun kode mata kuliah. Yang paling penting adalah antara objek mahasiswa, matakuliah, dan nilai harus saling terhubung.

- Class Mahasiswa

```

public class Mahasiswa {
    //mendeklarasikan atribut
    String nim;
    String nama;
    String notelp;
    //membuat konstruktor default
    public Mahasiswa() {

    }
    //membuat konstruktor berparameter
    public Mahasiswa(String nim, String nama, String
notelp) {
        this.nim = nim;
        this.nama = nama;

```

```

        this.notelp = notelp;
    }

```

- **Class Nilai**

```

public class Nilai {
    //mendeklarasikan atribut
    String kodenilai;
    String kodematakuliah;
    String nimmhs;
    Float nilai;

    //membuat konstruktor default
    public Nilai() {

    }

    //membuat konstruktor berparameter
    Nilai(String kodenilai, float nilai, String
    kodematakuliah, String nimmhs) {
        this.kodenilai = kodenilai;
        this.nilai = nilai;
        this.kodematakuliah = kodematakuliah;
        this.nimmhs = nimmhs;
    }

```
- **Class mataKuliah**

```

public class mataKuliah {
    //mendeklarasikan atribut
    String kode;
    String matakuliah;
    String sks;

    //membuat konstruktor default
    public mataKuliah(){

    } //membuat konstruktor berparameter
    public mataKuliah(String kode, String
    matakuliah, String sks) {
        this.kode = kode;
        this.matakuliah = matakuliah;
        this.sks = sks;
    }
}

```
- **Class MahasiswaMain**

```

/*
 * To change this license header, choose License
Headers in Project Properties.
 * To change this template file, choose Tools |
Templates
 * and open the template in the editor.
 */
package Tugas2;

```



```

import java.util.Scanner;
import java.util.List;
import java.util.ArrayList;
/**
 *
 * @author ACHE
 */
public class mahasiswaMain {

    /**
     * @param args the command line arguments
     */
    List<Mahasiswa> mhs = new ArrayList<>();
    List<mataKuliah> mk = new ArrayList<>();
    List<Nilai> nli = new ArrayList<>();
    static Scanner sc = new Scanner(System.in);
    public static void main(String[] args) {
        mahasiswaMain obj = new mahasiswaMain();
        int pilih;
        obj.setDataMHSdanMatkul();
        do {

System.out.println("*****");
System.out.println("SISTEM PENGOLAHAN
DATA NILAI MAHASISWA SEMESTER");

System.out.println("*****");
        System.out.println("1. Input Nilai");
        System.out.println("2. Tampil Nilai");
        System.out.println("3. Mencari Nilai
Mahasiswa");
        System.out.println("4. Urut Data
Nilai");
        System.out.println("5. Keluar");

System.out.println("*****");
        System.out.print("Pilih : ");
        pilih = sc.nextInt();
        if (pilih == 1) {
            obj.menusatu();
        } else if (pilih == 2) {
            obj.menudua();
        } else if (pilih == 3) {
            obj.menutiga();
        } else if (pilih == 4) {
            obj.menuempat();
        } else {

```

```

        System.out.println("anda telah keluar
dari program");
    }
    System.out.println("");
    }while (pilih >= 1 && pilih <= 4);
    }
    //membuat method untuk menampung data
    void setDataMHSdanMatkul(){
        mhs.add(new Mahasiswa("20001",
"Thalhah", "021xxx"));
        mhs.add(new Mahasiswa("20002", "Zubair",
"021xxx"));
        mhs.add(new Mahasiswa("20003", "Abdur-
Rahman", "021xxx"));
        mhs.add(new Mahasiswa("20004", "Sa'ad",
"021xxx"));
        mhs.add(new Mahasiswa("20005", "Sa'id",
"021xxx"));
        mhs.add(new Mahasiswa("20006",
"Ubaidah", "021xxx"));

        mk.add(new mataKuliah("00001", "Internet
of Things", "3"));
        mk.add(new mataKuliah("00002",
"Algoritma dan Struktur Data", "2"));
        mk.add(new mataKuliah("00003",
"Algoritma dan Pemrograman", "2"));
        mk.add(new mataKuliah("00004",
"Praktikum Algoritma dan Struktur Data", "3"));
        mk.add(new mataKuliah("00005",
"Praktikum Algoritma dan Pemrograman", "3"));
    }
    void menuSatu(){
        sc.nextLine();
        System.out.println("Masukkan Data ");
        System.out.print("Kode : ");
        String kode = sc.next();
        System.out.print("Nilai : ");
        float nilai = sc.nextFloat();
        System.out.println("\nDAFTAR
MAHASISWA");
    }

    System.out.println("*****
*****");
    System.out.println("NIM\tNama\t\tTelp");
    for (Mahasiswa mahasiswa : mhs) {
        System.out.println(mahasiswa.nim +
"\t" +
        mahasiswa.nama + "\t\t" +
mahasiswa.notelp);
    }

```

```

    }
    boolean ketemu = true;
    String pilihmahasiswa; int konfirmasi2 =
0;
    do {
        System.out.print("Pilih Mahasiswa by
NIM : ");
        pilihmahasiswa = sc.next();
        for (Mahasiswa mahasiswa : mhs) {
            if
(pilihmahasiswa.equals(mahasiswa.nim)) {
                ketemu = false;
                konfirmasi2++;
            }
        }
        if (konfirmasi2 == 0) {
            System.out.println("Data Kode
NIM Tidak Ditemukan");
        }
    }while (ketemu);
    System.out.println("\nDAFTAR MATA
KULIAH");

    System.out.println("*****
*****");
    System.out.println("Kode\tMata
Kuliah\t\tSKS");
    for (mataKuliah matakuliah : mk) {

        System.out.println(matakuliah.kode+"\t" +
matakuliah.matakuliah + "\t" + matakuliah.sks);
    }
    boolean ketemu2 = true;
    String pilihmatakuliah; int konfirmasi =
0;
    do {
        System.out.print("Pilih MK by Kode :
");
        pilihmatakuliah = sc.next();
        for (mataKuliah mklh : mk) {
            if
(pilihmatakuliah.equals(mklh.kode)) {
                ketemu2 = false;
                konfirmasi++;
            }
        }
        if (konfirmasi == 0) {
            System.out.println("Data Kode
Matakuliah Tidak Ditemukan");
        }
    }while (ketemu2);

```

```

        nli.add(new Nilai(kode, nilai,
pilihatakuliah, pilihmahasiswa));
    }
    //membuat menu ke dua
    void menudua() {
        System.out.println("DAFTAR NILAI
MAHASISWA");

System.out.println("*****
*****");
        System.out.println("NIM\tNama
Mahasiswa\t\tMata Kuliah\t\tSKS\tNilai");
        nli.forEach(nilaisatuan -> {

System.out.print(nilaisatuan.nimmhs+"\t");
            for (Mahasiswa mahasiswa : mhs)
{

if(mahasiswa.nim.equals(nilaisatuan.nimmhs)){

System.out.print(mahasiswa.nama+"\t\t\t");
                }
            }
            for (mataKuliah matkul : mk) {
                if
(matkul.kode.equals(nilaisatuan.kodematakuliah)) {

System.out.print(matkul.matakuliah + "\t" +
matkul.sks+"\t");
                    }
                }

System.out.println(nilaisatuan.nilai);
            });
        }
        //membuat menu ke tiga
        void menutiga() {
            menudua();
            System.out.print("Masukkan Data
Mahasiswa[nim] : ");
            String caridata = sc.next();
            int konfirmasi3 = 0;
            for (Nilai carinilai : nli) {
                if
(carinilai.nimmhs.equals(caridata)) {

System.out.print(carinilai.nimmhs+"\t");

```

```

                                for (Mahasiswa mahasiswa :
mhs) {

if(mahasiswa.nim.equals(carinilai.nimmhs)){

System.out.print(mahasiswa.nama+"\t");
                                }
                                }
                                for (mataKuliah matkul : mk)
{
                                if
(matkul.kode.equals(carinilai.kodematakuliah)) {

System.out.print(matkul.matakuliah + "\t" +
matkul.sks+"\t");
                                }
                                }

System.out.println(carinilai.nilai);
                                konfirmasi3++;
                                }
                                }
                                if (konfirmasi3 == 0) {
                                    System.out.println("Data Dengan
NIM " + caridata + " Tidak Ditemukan");
                                }
                                }
                                //membuat menu ke empat
                                void menuempat(){
                                    for (int i = 0; i < nli.size()-1;
i++) {
                                        for (int j = 0; j < nli.size()-
i-1; j++) {
                                            if (nli.get(j).nilai >
nli.get(j+1).nilai){
                                                float swap =
nli.get(j).nilai;
                                                nli.get(j).nilai =
nli.get(j+1).nilai;
                                                nli.get(j+1).nilai =
swap;
                                            }
                                        }
                                    }
                                }
                                menudua();
                                }
}

```

Hasil:

```
Output - minggu16 (run-single) X
*****
SISTEM PENGOLAHAN DATA NILAI MAHASISWA SEMESTER
*****
1. Input Nilai
2. Tampil Nilai
3. Mencari Nilai Mahasiswa
4. Urut Data Nilai
5. Keluar
*****
Pilih : 1
Masukkan Data
Kode : 0001
Nilai : 80.75

DAFTAR MAHASISWA
*****
NIM    Nama      Telp
20001  Thalbah   021xxx
20002  Zubair     021xxx
20003  Abdur-Rahman 021xxx
20004  Sa'ad       021xxx
20005  Sa'id       021xxx
20006  Ubaidah     021xxx
Pilih Mahasiswa by NIM :20001

DAFTAR MATA KULIAH
*****
Kode   Mata Kuliah      SKS
00001  Internet of Things    3
00002  Algoritma dan Struktur Data  2
00003  Algoritma dan Pemrograman  2
00004  Praktikum Algoritma dan Struktur Data 3
00005  Praktikum Algoritma dan Pemrograman  3
Pilih MK by Kode :00001

Output - minggu16 (run-single) X
*****
SISTEM PENGOLAHAN DATA NILAI MAHASISWA SEMESTER
*****
1. Input Nilai
2. Tampil Nilai
3. Mencari Nilai Mahasiswa
4. Urut Data Nilai
5. Keluar
*****
Pilih : 1
Masukkan Data
Kode : 0002
Nilai : 91.25

DAFTAR MAHASISWA
*****
NIM    Nama      Telp
20001  Thalbah   021xxx
20002  Zubair     021xxx
20003  Abdur-Rahman 021xxx
20004  Sa'ad       021xxx
20005  Sa'id       021xxx
20006  Ubaidah     021xxx
Pilih Mahasiswa by NIM :20002

Output - minggu16 (run-single) X
*****
DAFTAR MATA KULIAH
*****
Kode   Mata Kuliah      SKS
00001  Internet of Things    3
00002  Algoritma dan Struktur Data  2
00003  Algoritma dan Pemrograman  2
00004  Praktikum Algoritma dan Struktur Data 3
00005  Praktikum Algoritma dan Pemrograman  3
Pilih MK by Kode :00002

Output - minggu16 (run-single) X
*****
SISTEM PENGOLAHAN DATA NILAI MAHASISWA SEMESTER
*****
1. Input Nilai
2. Tampil Nilai
3. Mencari Nilai Mahasiswa
4. Urut Data Nilai
5. Keluar
*****
Pilih : 1
Masukkan Data
Kode : 0003
Nilai : 95.00

DAFTAR MAHASISWA
*****
NIM    Nama      Telp
20001  Thalbah   021xxx
20002  Zubair     021xxx
20003  Abdur-Rahman 021xxx
20004  Sa'ad       021xxx
20005  Sa'id       021xxx
20006  Ubaidah     021xxx
Pilih Mahasiswa by NIM :20004
```

```

DAFTAR MATA KULIAH
*****
Kode    Mata Kuliah      SKS
00001   Internet of Things  3
00002   Algoritma dan Struktur Data  2
00003   Algoritma dan Pemrograman  2
00004   Praktikum Algoritma dan Struktur Data  3
00005   Praktikum Algoritma dan Pemrograman  3
Pilih MK by Kode :00005

```

```

n Tools Window Help
380.8/590.0MB
Output - minggu16 (run-single) X
SISTEM PENGOLAHAN DATA NILAI MAHASISWA SEMESTER
*****
1. Input Nilai
2. Tampil Nilai
3. Mencari Nilai Mahasiswa
4. Urut Data Nilai
5. Keluar
*****
Pilih : 2
DAFTAR NILAI MAHASISWA
*****
NIM    Nama Mahasiswa      Mata Kuliah      SKS    Nilai
20001   Thalbah             Internet of Things  3      80.75
20002   Zubair                 Algoritma dan Struktur Data  2      91.25
20004   Sa'ad                  Praktikum Algoritma dan Pemrograman  3      95.0

*****
SISTEM PENGOLAHAN DATA NILAI MAHASISWA SEMESTER
*****
1. Input Nilai
2. Tampil Nilai
3. Mencari Nilai Mahasiswa
4. Urut Data Nilai
5. Keluar
*****
Pilih : 3
DAFTAR NILAI MAHASISWA
*****
NIM    Nama Mahasiswa      Mata Kuliah      SKS    Nilai
20001   Thalbah             Internet of Things  3      80.75
20002   Zubair                 Algoritma dan Struktur Data  2      91.25
20004   Sa'ad                  Praktikum Algoritma dan Pemrograman  3      95.0
Masukkan Data Mahasiswa[nim] 20004
20004   Sa'ad                  Praktikum Algoritma dan Pemrograman  3      95.0

*****
SISTEM PENGOLAHAN DATA NILAI MAHASISWA SEMESTER
*****
1. Input Nilai
2. Tampil Nilai
3. Mencari Nilai Mahasiswa
4. Urut Data Nilai
5. Keluar
*****
Pilih : 4
DAFTAR NILAI MAHASISWA
*****
NIM    Nama Mahasiswa      Mata Kuliah      SKS    Nilai
20001   Thalbah             Internet of Things  3      80.75
20002   Zubair                 Algoritma dan Struktur Data  2      91.25
20004   Sa'ad                  Praktikum Algoritma dan Pemrograman  3      95.0

*****
SISTEM PENGOLAHAN DATA NILAI MAHASISWA SEMESTER
*****
1. Input Nilai
2. Tampil Nilai
3. Mencari Nilai Mahasiswa
4. Urut Data Nilai
5. Keluar
*****
Pilih : 5
anda telah keluar dari program

```