

PRAKTIKUM PEMROGRAMAN 1

TUGAS PERTEMUAN 6

Nama : Muhammad Faris Fathur Rohman

NRP : 223040126

Kelas : D

Link github : https://github.com/Riss27/PP1_223040126_Pertemuan6.git

Latihan 1

```
public boolean find(int x)
{
    Node curNode = HEAD;
    boolean ketemu = false;

    while (curNode != null && !ketemu)
    {
        if (curNode.getData() == x)
        {
            ketemu = true;
        }
        else
        {
            curNode = curNode.getNext();
        }
    }
    return ketemu;
}

/*
 * mencari nilai x dalam linked list, dimulai dari curNode awal (HEAD).
 * Selama curNode tidak kosong dan nilai x belum ditemukan,
 * program akan terus mencari. Jika nilai x ditemukan,
 * fungsi mengembalikan true, jika tidak, mengembalikan false.
 */
```

Latihan 2

```
1  v public class StrukturListTest {  
    Run | Debug | Codeium: Refactor | Explain | Generate Javadoc  
2  v  public static void main(String[] args) {  
3      StrukturList list = new StrukturList();  
4  
5      list.addHead(data:6);  
6      list.addHead(data:4);  
7      list.addHead(data:5);  
8  
9      list.displayElement();  
10     System.out.println("\n" +list.find(x:6));  
11 }  
12 }
```

[Running] cd "d:\UNPAS\SEMESTER 4\PRAKTIKUM PEMROGRAMAN 1\PP1_223040126_Pertemuan6\"

5 4 6

true

Latihan 3

```
// algoritma fungsi size
Codeium: Refactor | Explain

public int size()
{
    Node curNode = HEAD;
    int jumlah = 0;
    while (curNode != null)
    {
        jumlah++;
        curNode = curNode.getNext();
    }
    return jumlah;
}

/*
 * Setiap curnode akan menambahkan satu pada totalnya, sehingga di akhir perhitungan,
 * totaljumlahnya akan sama dengan jumlah simpul yang ada.
 */
```

Latihan 4

```
1 public class SturkturListTest2 {
2     public static void main(String[] args)
3     {
4         StrukturList list = new StrukturList();
5
6         list.addHead(data:3);
7         list.addHead(data:2);
8         list.addHead(data:4);
9         list.addHead(data:6);
10        list.addHead(data:7);
11
12        list.displayElement();
13        System.out.println("\n" + list.size());
14    }
15 }
```

```
[Running] cd "d:\UNPAS\SEMESTER 4\PRAKTIKUM PEMROGRAMAN 1\PP1_223040126_Pertemuan6\"
7 6 4 2 3
5
```

Tugas

```
35 // algoritma fungsi removeAll
    Codeium: Refactor | Explain
36 void removeAll() {
37     HEAD = null;
38 }
```

```
1 public class StrukturListTest3 {
    Run | Debug | Codeium: Refactor | Explain | Generate Javadoc
2     public static void main(String[] args) {
3         StrukturList list = new StrukturList();
4
5         list.addHead(data:9);
6         list.addHead(data:1);
7         list.addHead(data:8);
8         list.addHead(data:7);
9         list.addHead(data:3);
10        list.addHead(data:4);
11
12        list.displayElement();
13        list.removeAll();
14        list.displayElement();
15    }
16 }
```

StrukturListTest3.java && java StrukturListTest3

4 3 7 8 1 9

List Kosong