



NAMA : Fatikah Salsabilla
NIM : 2341720003
NO ABSEN : 12
KELAS : 1C
MATERI : Double Linked List (Jobsheet 12)

LAPORAN PRAKTIKUM ALGORITMA DAN STRUKTUR DATA

12.1 Kegiatan Praktikum 1

Node12.java

```
package P12.DoubleLinkedLists;

public class Node12 {
    int data;
    Node12 prev, next;

    Node12(Node12 prev, int data, Node12 next){
        this.prev = prev;
        this.data = data;
        this.next = next;
    }
}
```

DoubleLinkedLists12.java

```
package P12.DoubleLinkedLists;

public class DoubleLinkedLists12 {
    Node12 head;
    int size;

    public DoubleLinkedLists12() {
        head = null;
        size = 0;
    }

    public boolean isEmpty() {
        return head == null;
    }

    public void addFirst(int item) {
        if (isEmpty()) {
            head = new Node12(prev:null, item, next:null);
        } else {
            Node12 newNode = new Node12(prev:null, item, head);
            head.prev = newNode;
            head = newNode;
        }
        size++;
    }

    public void addLast(int item) {
        if (isEmpty()) {
            addFirst(item);
        } else {
            Node12 current = head;
            while (current.next != null) {
                current = current.next;
            }
            Node12 newNode = new Node12(current, item, next:null);
            current.next = newNode;
            size++;
        }
    }

    public void add(int item, int index) throws Exception {
        if (isEmpty()) {
            addFirst(item);
        } else if (index < 0 || index > size) {
            throw new Exception(message:"Nilai indeks di luar batas");
        } else {
            Node12 current = head;
            int i = 0;
            while (i < index) {
                current = current.next;
                i++;
            }
            if (current.prev == null) {
                Node12 newNode = new Node12(prev:null, item, current);
                current.prev = newNode;
                head = newNode;
            } else {
                Node12 newNode = new Node12(current.prev, item, current);
                newNode.prev = current.prev;
                newNode.next = current;
                current.prev.next = newNode;
                current.prev = newNode;
            }
            size++;
        }
    }

    public int size() {
        return size;
    }

    public void clear() {
        head = null;
        size = 0;
    }

    public void print() {
        if (isEmpty()) {
            Node12 tmp = head;
            while (tmp != null) {
                System.out.print(tmp.data + "\t");
                tmp = tmp.next;
            }
            System.out.println(x:"\nberhasil diisi");
        } else {
            System.out.println(x:"Linked Lists Kosong");
        }
    }
}
```



NAMA : Fatikah Salsabilla
NIM : 2341720003
NO ABSEN : 12
KELAS : 1C
MATERI : Double Linked List (Jobsheet 12)

DoubleLinkedListsMain12.java

```
package P12.DoubleLinkedLists;

public class DoubleLinkedListMain12 {
    Run | Debug
    public static void main(String[] args) throws Exception {

        DoubleLinkedLists12 dll = new DoubleLinkedLists12();

        dll.print();
        System.out.println("Size : " + dll.size());
        System.out.println(x:"=====");
        dll.addFirst(item:3);
        dll.addLast(item:4);
        dll.addFirst(item:7);
        dll.print();
        System.out.println("Size : " + dll.size());
        System.out.println(x:"=====");
        dll.add(item:40, index:1);
        dll.print();
        System.out.println("Size : " + dll.size());
        System.out.println(x:"=====");
        dll.clear();
        dll.print();
        System.out.println("Size : " + dll.size());
    }
}
```

Hasil running

```
Linked Lists Kosong
Size : 0
=====
7      3      4
berhasil diisi
Size : 3
=====
7      40     3      4
berhasil diisi
Size : 4
=====
Linked Lists Kosong
Size : 0
PS C:\kuliah\SM 2\PrakASD_1C_12>
```

Question :

1. Jelaskan perbedaan antara single linked list dengan double linked lists!

Jawab :

- Double linked list mempunyai 2 pointer yaitu next dan prev, jika Single linked list hanya memiliki 1 pointer yaitu next.
- Double linked list bisa menunjuk ke 2 arah "next" untuk ke node berikutnya dan "prev" untuk ke node sebelumnya, jika Single linked list hanya bisa menunjuk ke node berikutnya

2. Perhatikan class Node, di dalamnya terdapat atribut next dan prev. Untuk apakah atribut tersebut?

Jawab :

Pointer next menunjuk pada node setelahnya dan pointer prev menunjuk pada node sebelumnya

3. Perhatikan konstruktor pada class DoubleLinkedLists. Apa kegunaan inisialisasi atribut head dan size seperti pada gambar berikut ini?



NAMA : Fatikah Salsabilla
NIM : 2341720003
NO ABSEN : 12
KELAS : 1C
MATERI : Double Linked List (Jobsheet 12)

```
public DoubleLinkedLists() {  
    head = null;  
    size = 0;  
}
```

Jawab :

Untuk memastikan bahwa double linked list dimulai dengan data kosong

4. Pada method addFirst(), kenapa dalam pembuatan object dari konstruktor class Node prev dianggap sama dengan null?

Node newNode = new Node(null, item, head);

Jawab :

Karena node yang akan ditambahkan pada double linked list tidak memiliki node sebelumnya

5. Perhatikan pada method addFirst(). Apakah arti statement head.prev = newNode ?

Jawab :

Untuk menghubungkan node yang berada di posisi head dengan node baru yang ditambahkan didepan dan untuk memastikan bahwa node saat ini yang menjadi head memiliki acuan ke node baru sebagai node sebelumnya

6. Perhatikan isi method addLast(), apa arti dari pembuatan object Node dengan mengisi parameter prev dengan current, dan next dengan null?

Node newNode = new Node(current, item, null);

Jawab :

Untuk memastikan bahwa node baru yang ditambahkan akan dihubungkan ke node terakhir dan akan menjadi node terakhir dalam urutan

7. Pada method add(), terdapat potongan kode program sebagai berikut:

```
while (i < index) {  
    current = current.next;  
    i++;  
}  
  
if (current.prev == null) {  
    Node newNode = new Node(null, item, current);  
    current.prev = newNode;  
    head = newNode;  
}  
else {  
    Node newNode = new Node(current.prev, item, current);  
    current.prev.next = newNode;  
    current.prev = newNode;  
}
```

jelaskan maksud dari bagian yang ditandai dengan kotak kuning.

Jawab :

untuk menambahkan node baru sebelum node pertama, yang pertama yaitu mengecek apakah current.prev == null jika iya, prev dari node pertama akan menunjuk ke node baru dan memperbarui head untuk menunjuk ke node baru sebagai node pertama



NAMA : Fatikah Salsabilla
NIM : 2341720003
NO ABSEN : 12
KELAS : 1C
MATERI : Double Linked List (Jobsheet 12)

12.2 Kegiatan Praktikum 2

DoubleLinkedLists12.java

```
public void removeFirst() throws Exception {
    if (isEmpty()) {
        throw new Exception(message:"Linked List masih kosong, tidak dapat dihapus!");
    } else if (size == 1) {
        removeLast();
    } else {
        head = head.next;
        head.prev = null;
        size--;
    }
}

public void removeLast() throws Exception {
    if (isEmpty()) {
        throw new Exception(message:"Linked List masih kosong, tidak dapat dihapus!");
    } else if (head.next == null) {
        head = null;
        size --;
        return;
    }
    Node12 current = head;
    while (current.next.next != null){
        current = current.next;
    }
    current.next = null;
    size--;
}

public void remove(int index) throws Exception {
    if (isEmpty()) {
        throw new Exception(message:"Linked List masih kosong, tidak dapat dihapus!");
    } else if (index == 0){
        removeFirst();
    } else {
        Node12 current = head;
        int i = 0;
        while (i < index) {
            current = current.next;
            i++;
        }
        if (current.next == null) {
            current.prev.next = null;
        } else if (current.prev == null) {
            current = current.next;
            current.prev = null;
            head = current;
        } else {
            current.prev.next = current.next;
            current.next.prev = current.prev;
        }
        size--;
    }
}
```



NAMA : Fatikah Salsabilla
NIM : 2341720003
NO ABSEN : 12
KELAS : 1C
MATERI : Double Linked List (Jobsheet 12)

DoubleLinkedListsMain12.java

```
dll.addLast(item:50);  
dll.addLast(item:40);  
dll.addLast(item:10);  
dll.addLast(item:20);  
dll.print ();  
System.out.println("Size : "+dll.size());  
System.out.println(x:"=====");  
dll.removeFirst ();  
dll.print();  
System.out.println("Size : "+dll.size());  
System.out.println(x:"=====");  
dll.removeLast ();  
dll.print ();  
System.out.println("Size : "+dll.size());  
System.out.println(x:"=====");  
dll.remove(index:1) ;  
dll.print();  
System.out.println("Size : "+dll.size());
```

Hasil running

```
50    40    10    20  
berhasil diisi  
Size : 4  
=====  
40    10    20  
berhasil diisi  
Size : 3  
=====  
40    10  
berhasil diisi  
Size : 2  
=====  
40  
berhasil diisi  
Size : 1  
PS C:\kuliah\SM 2\PrakASD_1C_12>
```

Question :

1. Apakah maksud statement berikut pada method removeFirst)?
head = head.next;
head.prev = null;

Jawab :

Head pada node pertama di geser ke posisi node selanjutnya kemudian head pada node pertama akan menjadi null

2. Bagaimana cara mendeteksi posisi data ada pada bagian akhir pada method removeLast()?

Jawab :

- 1) Pertama yaitu mengecek apakah linked list kosong, jika iya, maka tidak ada node yang bisa dihapus
- 2) Jika node hanya 1 node maka head akan akan dihapus menjadi null kemudian size akan berkurang
- 3) Jika memiliki lebih dari 1 node maka akan melakukan perulangan sampai sebelum node terakhir



NAMA : Fatikah Salsabilla
NIM : 2341720003
NO ABSEN : 12
KELAS : 1C
MATERI : Double Linked List (Jobsheet 12)

- 4) Jika sudah menemukan node sebelum terakhir kemudian menggunakan `current.next` menjadi null untuk menghapus node terakhir kemudian seze nya dikurangi
3. Jelaskan alasan potongan kode program di bawah ini tidak cocok untuk perintah `remove`!

```
Node tmp = head.next;  
head.next=tmp.next;  
tmp.next.prev=head;
```

Jawab :

Karena kode diatas menghapus elemen yang berada di posisi ke dua

4. Jelaskan fungsi kode program berikut ini pada fungsi `remove`!

```
current.prev.next = current.next;  
current.next.prev = current.prev;
```

Jawab :

`current.prev.next = current.next;` digunakan untuk menghubungkan node sebelum current ke node setelah current.

`current.next.prev = current.prev;` digunakan untuk menghubungkan node setelah current ke node sebelum current

12.3 Kegiatan Praktikum 3

DoubleLinkedLists12.java

```
public int getFirst() throws Exception{  
    if (isEmpty()) {  
        throw new Exception(message:"Linked list kosong");  
    }  
    return head.data;  
}  
  
public int getLast() throws Exception{  
    if (isEmpty()) {  
        throw new Exception(message:"Linked list kosong");  
    }  
    Node12 tmp = head;  
    while (tmp.next != null) {  
        tmp = tmp.next;  
    }  
    return tmp.data;  
}  
  
public int get(int index) throws Exception{  
    if (isEmpty() || index >= size) {  
        throw new Exception(message:"Nilai indeks di luar batas");  
    }  
    Node12 tmp = head;  
    for (int i = 0; i < index; i++) {  
        tmp = tmp.next;  
    }  
    return tmp.data;  
}
```



NAMA : Fatikah Salsabilla
NIM : 2341720003
NO ABSEN : 12
KELAS : 1C
MATERI : Double Linked List (Jobsheet 12)

DoubleLinkedListsMain12.java

```
dll.print();
System.out.println("Size: " + dll.size());
System.out.println(x:"=====");
dll.addFirst(item:3);
dll.addLast(item:4);
dll.addFirst(item:7);
dll.print();
System.out.println("Size: " + dll.size());
System.out.println(x:"=====");
dll.add(item:40, index:1);
dll.print();
System.out.println("Size: " + dll.size());
System.out.println(x:"=====");
System.out.println ("Data awal pada Linked Lists adalah: " + dll.getFirst());
System.out.println ("Data akhir pada Linked Lists adalah: " + dll.getLast ());
System.out.println ("Data indeks ke-1 pada Linked Lists adalah: " + dll.get(index:1));
```

Hasil running

```
Linked Lists Kosong
Size: 0
=====
7      3      4
berhasil diisi
Size: 3
=====
7      40     3      4
berhasil diisi
Size: 4
=====
Data awal pada Linked Lists adalah: 7
Data akhir pada Linked Lists adalah: 4
Data indeks ke-1 pada Linked Lists adalah: 40
PS C:\kuliaah\SM 2\PrakASD 1C 12> |
```

Question :

1. Jelaskan method size() pada class DoubleLinkedLists!

Jawab :

Untuk menghitung dan mengembalikan jumlah elemen dalam linked list

2. Jelaskan cara mengatur indeks pada double linked lists supaya dapat dimulai dari indeks ke-1!

Jawab :

3. Jelaskan perbedaan karakteristik fungsi Add pada Double Linked Lists dan Single Linked Lists!

Jawab :

Double Linked Lists lebih fleksibel dalam hal penambahan pada posisi tertentu karena kemampuannya untuk akses maju dan mundur, sementara Single Linked Lists lebih sederhana dan efisien dalam penambahan di awal atau di akhir linked list

4. Jelaskan perbedaan logika dari kedua kode program di bawah ini!

```
public boolean isEmpty(){
    if(size == 0){
        return true;
    } else{
        return false;
    }
}
```

(a)

```
public boolean isEmpty(){
    return head == null;
}
```

(b)



NAMA : Fatikah Salsabilla
NIM : 2341720003
NO ABSEN : 12
KELAS : 1C
MATERI : Double Linked List (Jobsheet 12)

Jawab :

- (a) Menggunakan variabel size untuk menentukan apakah list kosong atau tidak, size menyimpan jumlah elemen. Jika size bernilai 0 metode isEmpty akan mengembalikan true (list kosong) dan jika size tidak bernilai 0, metode isEmpty akan mengembalikan false (list tidak kosong)
- (b) Menggunakan variabel head untuk menentukan apakah list kosong atau tidak, Jika list kosong, head akan bernilai null, sehingga isEmpty mengembalikan true, jika head tidak bernilai null, isEmpty akan mengembalikan false (tidak kosong)

Tugas Praktikum

1. Buat program antrian vaksinasi menggunakan queue berbasis double linked list sesuai ilustrasi dan menu di bawah ini! (**counter jumlah antrian tersisa di menu cetak(3) dan data orang yang telah divaksinasi di menu Hapus Data(2) harus ada**)

Jawab :

Node12.java

```
package P12.TugasPraktikum.Tugas1;

public class Node12 {
    String noAntrian;
    String nama;
    Node12 prev, next;

    Node12(Node12 prev, String noAntrian, String nama, Node12 next) {
        this.prev = prev;
        this.noAntrian = noAntrian;
        this.nama = nama;
        this.next = next;
    }
}
```




NAMA : Fatikah Salsabilla
NIM : 2341720003
NO ABSEN : 12
KELAS : 1C
MATERI : Double Linked List (Jobsheet 12)

Queue12.java

```
package P12.TugasPraktikum.Tugas1;

public class Queue12 {
    Node12 head;
    int size;

    public Queue12() {
        head = null;
        size = 0;
    }

    public boolean isEmpty() {
        return head == null;
    }

    public void enqueue(String noAntrian, String nama) {
        if (isEmpty()) {
            head = new Node12(prev:null, noAntrian, nama, next:null);
        } else {
            Node12 current = head;
            while (current.next != null) {
                current = current.next;
            }
            Node12 newNode = new Node12(current, noAntrian, nama, next:null);
            current.next = newNode;
            size++;
        }
    }

    public void dequeue() throws Exception {
        if (isEmpty()) {
            throw new Exception(message:"Antrian masih kosong, tidak dapat dihapus!");
        } else if (size == 1) {
            System.out.println(head.nama + " telah di vaksinasi.");
            head = null;
        } else {
            System.out.println(head.nama + " telah di vaksinasi.");
            head = head.next;
            head.prev = null;
        }
        size--;
    }

    public String peek() throws Exception {
        if (isEmpty()) {
            throw new Exception(message:"Antrian kosong");
        }
        return "Nomor Antrian: " + head.noAntrian + ", Nama: " + head.nama;
    }

    public void printQueue() {
        if (!isEmpty()) {
            Node12 tmp = head;
            System.out.println(x:"+++++++");
            System.out.println(x:"Data Pengantri Vaksin");
            System.out.println(x:"+++++++");
            System.out.printf(format:"| %-5s | %-5s |\n", ...args:"No.", "Nama");
            while (tmp != null) {
                System.out.printf(format:"| %-5s | %-5s |\n", tmp.noAntrian, tmp.nama);
                tmp = tmp.next;
            }
            System.out.println("Sisa Antrian : " + size);
        } else {
            System.out.println(x:"Antrian Kosong");
        }
    }

    public int size() {
        return size;
    }

    public void clear() {
        head = null;
        size = 0;
    }
}
```



NAMA : Fatikah Salsabilla
NIM : 2341720003
NO ABSEN : 12
KELAS : 1C
MATERI : Double Linked List (Jobsheet 12)

DLLMain12.java

```
1 package P12.TugasPraktikum.Tugas1;  
2  
3 import java.util.Scanner;  
4  
5 public class DLLMain12 {  
6     public static void main(String[] args) throws Exception {  
7         Queue12 queue = new Queue12();  
8         Scanner sc12 = new Scanner(System.in);  
9         int pilihan;  
10        boolean isTrue = false;  
11  
12        String noAntrian, nama;  
13  
14        while (true) {  
15            System.out.println("+++++++");  
16            System.out.println("PENGANTRI VAKSIN EXTRAVAGANZA");  
17            System.out.println("+++++++");  
18            System.out.println("1. Tambah Data Penerima Vaksin");  
19            System.out.println("2. Hapus Data Pengantri Vaksin");  
20            System.out.println("3. Daftar Penerima Vaksin");  
21            System.out.println("4. Keluar ");  
22            System.out.println("+++++++");  
23            pilihan = sc12.nextInt();  
24            sc12.nextLine();  
25  
26            switch (pilihan) {  
27                case 1:  
28                    System.out.println("-----");  
29                    System.out.println("Masukkan Data Penerima Vaksin");  
30                    System.out.println("-----");  
31                    System.out.println("Nomor Antrian:");  
32                    noAntrian = sc12.nextLine();  
33                    System.out.println("Nama Penerima:");  
34                    nama = sc12.nextLine();  
35                    queue.enqueue(noAntrian, nama);  
36                    break;  
37  
38                case 2:  
39                    if (!queue.isEmpty()) {  
40                        queue.dequeue();  
41                        queue.printQueue();  
42                    } else {  
43                        System.out.println("Antrian masih kosong, tidak dapat dihapus!");  
44                    }  
45                    break;  
46  
47                case 3:  
48                    queue.printQueue();  
49                    break;  
50  
51                case 4:  
52                    System.exit(0);  
53                    break;  
54            }  
55        }  
56    }  
57 }  
58 }  
59 }  
60 }
```

Hasil Running

- Menu awal

```
+++++++  
PENGANTRI VAKSIN EXTRAVAGANZA  
+++++++  
1. Tambah Data Penerima Vaksin  
2. Hapus Data Pengantri Vaksin  
3. Daftar Penerima Vaksin  
4. Keluar  
+++++++
```



NAMA : Fatikah Salsabilla
NIM : 2341720003
NO ABSEN : 12
KELAS : 1C
MATERI : Double Linked List (Jobsheet 12)

- Cetak Data

```
+++++
PENGANTRI VAKSIN EXTRAVAGANZA
+++++
1. Tambah Data Penerima Vaksin
2. Hapus Data Pengantri Vaksin
3. Daftar Penerima Vaksin
4. Keluar
+++++
1
-----
Masukkan Data Penerima Vaksin
-----
Nomor Antrian:
123
Nama Penerima:
Joko
+++++
PENGANTRI VAKSIN EXTRAVAGANZA
+++++
1. Tambah Data Penerima Vaksin
2. Hapus Data Pengantri Vaksin
3. Daftar Penerima Vaksin
4. Keluar
+++++
1
-----
Masukkan Data Penerima Vaksin
-----
Nomor Antrian:
124
Nama Penerima:
Mely
+++++
PENGANTRI VAKSIN EXTRAVAGANZA
+++++
1. Tambah Data Penerima Vaksin
2. Hapus Data Pengantri Vaksin
3. Daftar Penerima Vaksin
4. Keluar
+++++
1
-----
Masukkan Data Penerima Vaksin
-----
Nomor Antrian:
135
Nama Penerima:
Johan
```

```
+++++
PENGANTRI VAKSIN EXTRAVAGANZA
+++++
1. Tambah Data Penerima Vaksin
2. Hapus Data Pengantri Vaksin
3. Daftar Penerima Vaksin
4. Keluar
+++++
3
-----
Data Pengantri Vaksin
+++++
| No. | Nama |
| 123 | Joko |
| 124 | Mely |
| 135 | Johan |
| 146 | Rosi |
Sisa Antrian : 4
```

```
+++++
PENGANTRI VAKSIN EXTRAVAGANZA
+++++
1. Tambah Data Penerima Vaksin
2. Hapus Data Pengantri Vaksin
3. Daftar Penerima Vaksin
4. Keluar
+++++
1
-----
Masukkan Data Penerima Vaksin
-----
Nomor Antrian:
146
Nama Penerima:
Rosi
```

- Hapus Data

```
+++++
PENGANTRI VAKSIN EXTRAVAGANZA
+++++
1. Tambah Data Penerima Vaksin
2. Hapus Data Pengantri Vaksin
3. Daftar Penerima Vaksin
4. Keluar
+++++
2
Joko telah di vaksinasi.
+++++
Data Pengantri Vaksin
+++++
| No. | Nama |
| 124 | Mely |
| 135 | Johan |
| 146 | Rosi |
Sisa Antrian : 3
```



NAMA : Fatikah Salsabilla
NIM : 2341720003
NO ABSEN : 12
KELAS : 1C
MATERI : Double Linked List (Jobsheet 12)

2. Buatlah program daftar film yang terdiri dari id, judul dan rating menggunakan double linked lists, bentuk program memiliki fitur pencarian melalui ID Film dan pengurutan Rating secara descending. Class Film wajib diimplementasikan dalam soal ini.

Jawab :

Film12.java

```
package P12.TugasPraktikum.Tugas2;

public class Film12 {
    String id;
    String judulFilm;
    double rating;
    Film12 prev, next;

    Film12(Film12 prev, String id, String judulFilm, double rating, Film12 next) {
        this.prev = prev;
        this.id = id;
        this.judulFilm = judulFilm;
        this.rating = rating;
        this.next = next;
    }
}
```

FilmDLL12.java

```
package P12.TugasPraktikum.Tugas2;

public class FilmDLL12 {
    Film12 head;
    int size;

    public FilmDLL12() {
        head = null;
        size = 0;
    }

    public boolean isEmpty() {
        return head == null;
    }

    public void addFirst(String id, String judulFilm, double rating) {
        if (isEmpty()) {
            head = new Film12(prev:null, id, judulFilm, rating, next:null);
        } else {
            Film12 newNode = new Film12(prev:null, id, judulFilm, rating, head);
            head.prev = newNode;
            head = newNode;
        }
        size++;
    }

    public void addLast(String id, String judulFilm, double rating) {
        if (isEmpty()) {
            addFirst(id, judulFilm, rating);
        } else {
            Film12 current = head;
            while (current.next != null) {
                current = current.next;
            }
            Film12 newNode = new Film12(current, id, judulFilm, rating, next:null);
            current.next = newNode;
            size++;
        }
    }

    public void add(String id, String judul, double rating, int index) {
        if (index < 0 || index > size) {
            System.out.println("Indeks di luar batas.");
            return;
        }

        if (index == 0) {
            addFirst(id, judul, rating);
        } else if (index == size) {
            addLast(id, judul, rating);
        } else {
            Film12 current = head;
            for (int i = 0; i < index - 1; i++) {
                current = current.next;
            }

            Film12 newNode = new Film12(current, id, judul, rating, current.next);
            current.next.prev = newNode;
            current.next = newNode;
            size++;
        }
    }

    public int getIndex(Film12 node) {
        Film12 current = head;
        int index = 0;
        while (current != null) {
            if (current == node) {
                return index;
            }
            current = current.next;
            index++;
        }
        return -1;
    }

    public int size() {
        return size;
    }

    public void clear() {
        head = null;
        size = 0;
    }
}
```



NAMA : Fatikah Salsabilla
NIM : 2341720003
NO ABSEN : 12
KELAS : 1C
MATERI : Double Linked List (Jobsheet 12)

```
public void print() {  
    if (isEmpty()) {  
        Film12 tmp = head;  
        while (tmp != null) {  
            System.out.println("ID: " + tmp.id);  
            System.out.println("Judul Film : " + tmp.judulFilm);  
            System.out.println("ipk : " + tmp.rating);  
            tmp = tmp.next;  
        }  
    } else {  
        System.out.println("Daftar Film Kosong");  
    }  
}  
  
public void removeFirst() throws Exception {  
    if (isEmpty()) {  
        throw new Exception(message: "Daftar Film Kosong");  
    } else if (size == 1) {  
        removeLast();  
    } else {  
        System.out.println("Data film " + head.judulFilm + " berhasil dihapus");  
        head = head.next;  
        head.prev = null;  
        size--;  
    }  
}  
  
public void removeLast() throws Exception {  
    if (isEmpty()) {  
        throw new Exception(message: "Daftar Film Kosong");  
    } else if (head.next == null) {  
        System.out.println("Data film " + head.judulFilm + " berhasil dihapus");  
        head = null;  
        size--;  
        return;  
    }  
  
    Film12 current = head;  
  
    while (current.next.next != null) {  
        current = current.next;  
    }  
    System.out.println("Data film " + current.next.judulFilm + " berhasil dihapus");  
    current.next = null;  
    size--;  
}
```

```
public void remove(int index) throws Exception {  
    if (isEmpty()) {  
        throw new Exception(message: "Daftar Film Kosong");  
    } else if (index == 0) {  
        removeFirst();  
    } else {  
        Film12 current = head;  
        int i = 0;  
        while (i < index) {  
            current = current.next;  
            i++;  
        }  
        System.out.println("Data film " + current.judulFilm + " berhasil dihapus");  
        if (current.next == null) {  
            current.prev.next = null;  
        } else {  
            current.prev.next = current.next;  
            current.next.prev = current.prev;  
        }  
        size--;  
    }  
}
```

```
public void bubbleSortDesc() {  
    if (size > 1) {  
        boolean swap;  
        do {  
            Film12 current = head;  
            Film12 prev = null;  
            Film12 next = head.next;  
            swap = false;  
  
            while (next != null) {  
                if (current.rating < next.rating) {  
                    swap = true;  
                    if (prev != null) {  
                        Film12 temp = next.next;  
                        prev.next = next;  
                        next.next = current;  
                        current.next = temp;  
                    } else {  
                        Film12 temp = next.next;  
                        head = next;  
                        next.next = current;  
                        current.next = temp;  
                    }  
                    prev = next;  
                    next = current.next;  
                } else {  
                    prev = current;  
                    current = next;  
                    next = next.next;  
                }  
            }  
        } while (swap);  
    }  
}
```

```
public int findSeqSearchIdFilm(String id) {  
    Film12 current = head;  
    int index = 0;  
    while (current != null) {  
        if (current.id.equals(id)) {  
            return index;  
        }  
        current = current.next;  
        index++;  
    }  
    return -1;  
}
```

```
public void printSearch(String id) {  
    Film12 current = head;  
    while (current != null) {  
        if (current.id.equals(id)) {  
            System.out.println("ID: " + current.id);  
            System.out.println("Judul Film: " + current.judulFilm);  
            System.out.println("IMDB Rating: " + current.rating);  
            return;  
        }  
        current = current.next;  
    }  
    System.out.println("Film dengan ID " + id + " tidak ditemukan.");  
}
```



NAMA : Fatikah Salsabilla
NIM : 2341720003
NO ABSEN : 12
KELAS : 1C
MATERI : Double Linked List (Jobsheet 12)

FilmDLLMain12.java

```
package P12.TugasPraktikum.Tugas2;

import java.util.Scanner;

public class FilmDLLMain12 {
    Run/Debug
    public static void main(String[] args) throws Exception {
        FilmDLL12 dataFilm = new FilmDLL12();
        Scanner sc12 = new Scanner(System.in);
        int pilihan;
        int index;
        boolean isTrue = false;
        String id, judulFilm;
        double rating;

        while (true) {
            System.out.println(x:"=====");
            System.out.println(x:"DATA FILM LAYAR LEBAR");
            System.out.println(x:"=====");
            System.out.println(x:"1. Tambah Data Awal");
            System.out.println(x:"2. Tambah Data Akhir");
            System.out.println(x:"3. Tambah Data Index Tertentu");
            System.out.println(x:"4. Hapus Data Pertama");
            System.out.println(x:"5. Hapus Data Terakhir");
            System.out.println(x:"6. Hapus Data Tertentu");
            System.out.println(x:"7. Cetak");
            System.out.println(x:"8. Cari ID Film");
            System.out.println(x:"9. Urut Data Rating Film-DESC");
            System.out.println(x:"10. Keluar ");
            System.out.println(x:"=====");
            pilihan = sc12.nextInt();
            sc12.nextLine();

            switch (pilihan) {
                case 1:
                    System.out.println(x:"Masukkan Data Film Posisi Awal");
                    System.out.println(x:"ID Film:");
                    id = sc12.nextLine();
                    System.out.println(x:"Judul Film:");
                    judulFilm = sc12.nextLine();
                    System.out.println(x:"Rating Film:");
                    rating = sc12.nextDouble();
                    dataFilm.addFirst(id, judulFilm, rating);
                    break;

                    case 2:
                        System.out.println(x:"Masukkan Data Film Posisi Akhir");
                        System.out.println(x:"ID Film:");
                        id = sc12.nextLine();
                        System.out.println(x:"Judul Film:");
                        judulFilm = sc12.nextLine();
                        System.out.println(x:"Rating Film:");
                        rating = sc12.nextDouble();
                        dataFilm.addLast(id, judulFilm, rating);
                        break;

                    case 3:
                        System.out.println(x:"Masukkan Data Film \nUrutan ke-");
                        System.out.println(x:"ID Film:");
                        id = sc12.nextLine();
                        System.out.println(x:"Judul Film:");
                        judulFilm = sc12.nextLine();
                        System.out.println(x:"Rating Film:");
                        rating = sc12.nextDouble();
                        index = 2;
                        System.out.println("Data Film ini akan masuk di urutan ke- " + (index + 1));
                        dataFilm.add(id, judulFilm, rating, index);
                        break;

                    case 4:
                        dataFilm.removeFirst();
                        break;

                    case 5:
                        dataFilm.removeLast();
                        break;

                    case 6:
                        System.out.println(x:"Hapus data pada index tertentu");
                        index = sc12.nextInt();
                        dataFilm.remove(index);
                        break;

                    case 7:
                        System.out.println(x:"Cetak Data");
                        dataFilm.print();
                        break;

                    case 8:
                        System.out.println(x:"Cari Data");
                        System.out.println(x:"Masukkan ID Film yang dicari: ");
                        id = sc12.nextLine();
                        index = dataFilm.findSeqSearchIdFilm(id);
                        if (index != -1) {
                            System.out.println("Data Id Film: " + id + " berada di node ke- " + (index + 1));
                            System.out.println(x:"IDENTITAS:");
                            dataFilm.printSearch(id);
                        } else {
                            System.out.println(x:"Data Id Film Tidak ditemukan ");
                        }
                        break;

                    case 9:
                        dataFilm.bubbleSortDesc();
                        dataFilm.print();
                        break;

                    case 10:
                        System.exit(status:0);
                        break;
            }
        }
    }
}
```

Hasil Running :

- Menu Awal

```
=====
DATA FILM LAYAR LEBAR
=====
1. Tambah Data Awal
2. Tambah Data Akhir
3. Tambah Data Index Tertentu
4. Hapus Data Pertama
5. Hapus Data Terakhir
6. Hapus Data Tertentu
7. Cetak
8. Cari ID Film
9. Urut Data Rating Film-DESC
10. Keluar
=====
```



NAMA : Fatikah Salsabilla
NIM : 2341720003
NO ABSEN : 12
KELAS : 1C
MATERI : Double Linked List (Jobsheet 12)

1. Tambah data awal

```
=====
DATA FILM LAYAR LEBAR
=====
1. Tambah Data Awal
2. Tambah Data Akhir
3. Tambah Data Index Tertentu
4. Hapus Data Pertama
5. Hapus Data Terakhir
6. Hapus Data Tertentu
7. Cetak
8. Cari ID Film
9. Urut Data Rating Film-DESC
10. Keluar
=====
1
Masukkan Data Film Posisi Awal
ID Film:
1222
Judul Film:
Spider-Man: No Way Home
Rating Film:
8.7
```

2. Tambah data akhir

```
=====
DATA FILM LAYAR LEBAR
=====
1. Tambah Data Awal
2. Tambah Data Akhir
3. Tambah Data Index Tertentu
4. Hapus Data Pertama
5. Hapus Data Terakhir
6. Hapus Data Tertentu
7. Cetak
8. Cari ID Film
9. Urut Data Rating Film-DESC
10. Keluar
=====
2
Masukkan Data Film Posisi Akhir
ID Film:
1346
Judul Film:
Uncharted
Rating Film:
6.7
```

3. Tambah data index tertentu

```
=====
DATA FILM LAYAR LEBAR
=====
1. Tambah Data Awal
2. Tambah Data Akhir
3. Tambah Data Index Tertentu
4. Hapus Data Pertama
5. Hapus Data Terakhir
6. Hapus Data Tertentu
7. Cetak
8. Cari ID Film
9. Urut Data Rating Film-DESC
10. Keluar
=====
3
Masukkan Data Film
Urutan ke-
ID Film:
1234
Judul Film:
Death on the Nile
Rating Film:
6.6
Data Film ini akan masuk di urutan ke- 3
```



NAMA : Fatikah Salsabilla
NIM : 2341720003
NO ABSEN : 12
KELAS : 1C
MATERI : Double Linked List (Jobsheet 12)

4. Hapus data pertama

```
=====
DATA FILM LAYAR LEBAR
=====
1. Tambah Data Awal
2. Tambah Data Akhir
3. Tambah Data Index Tertentu
4. Hapus Data Pertama
5. Hapus Data Terakhir
6. Hapus Data Tertentu
7. Cetak
8. Cari ID Film
9. Urut Data Rating Film-DESC
10. Keluar
=====
4
Data film Spider-Man: No Way Home berhasil dihapus
=====
DATA FILM LAYAR LEBAR
=====
1. Tambah Data Awal
2. Tambah Data Akhir
3. Tambah Data Index Tertentu
4. Hapus Data Pertama
5. Hapus Data Terakhir
6. Hapus Data Tertentu
7. Cetak
8. Cari ID Film
9. Urut Data Rating Film-DESC
10. Keluar
=====
7
Cetak Data
ID: 1346
Judul Film : Uncharted
ipk : 6.7
ID: 1234
Judul Film : Death on the Nile
ipk : 6.6
```

5. Hapus data terakhir

```
=====
DATA FILM LAYAR LEBAR
=====
1. Tambah Data Awal
2. Tambah Data Akhir
3. Tambah Data Index Tertentu
4. Hapus Data Pertama
5. Hapus Data Terakhir
6. Hapus Data Tertentu
7. Cetak
8. Cari ID Film
9. Urut Data Rating Film-DESC
10. Keluar
=====
5
Data film Death on the Nile berhasil dihapus
=====
DATA FILM LAYAR LEBAR
=====
1. Tambah Data Awal
2. Tambah Data Akhir
3. Tambah Data Index Tertentu
4. Hapus Data Pertama
5. Hapus Data Terakhir
6. Hapus Data Tertentu
7. Cetak
8. Cari ID Film
9. Urut Data Rating Film-DESC
10. Keluar
=====
7
Cetak Data
ID: 1346
Judul Film : Uncharted
ipk : 6.7
```

6. Hapus data tertentu

```
=====
DATA FILM LAYAR LEBAR
=====
1. Tambah Data Awal
2. Tambah Data Akhir
3. Tambah Data Index Tertentu
4. Hapus Data Pertama
5. Hapus Data Terakhir
6. Hapus Data Tertentu
7. Cetak
8. Cari ID Film
9. Urut Data Rating Film-DESC
10. Keluar
=====
6
Hapus data pada index tertentu
0
Data film Uncharted berhasil dihapus
=====
DATA FILM LAYAR LEBAR
=====
1. Tambah Data Awal
2. Tambah Data Akhir
3. Tambah Data Index Tertentu
4. Hapus Data Pertama
5. Hapus Data Terakhir
6. Hapus Data Tertentu
7. Cetak
8. Cari ID Film
9. Urut Data Rating Film-DESC
10. Keluar
=====
7
Cetak Data
Daftar Film Kosong
```




NAMA : Fatikah Salsabilla
NIM : 2341720003
NO ABSEN : 12
KELAS : 1C
MATERI : Double Linked List (Jobsheet 12)

7. Cetak

```
=====
DATA FILM LAYAR LEBAR
=====
1. Tambah Data Awal
2. Tambah Data Akhir
3. Tambah Data Index Tertentu
4. Hapus Data Pertama
5. Hapus Data Terakhir
6. Hapus Data Tertentu
7. Cetak
8. Cari ID Film
9. Urut Data Rating Film-DESC
10. Keluar
=====
7
Cetak Data
ID: 1222
Judul Film : Spider-Man: No Way Home
ipk : 8.7
ID: 1765
Judul Film : Skyfall
ipk : 7.8
ID: 1567
Judul Film : The Dark Knight Rises
ipk : 8.4
ID: 1234
Judul Film : Death on The Nile
ipk : 6.6
ID: 1346
Judul Film : Uncharted
ipk : 6.7
```

8. Cari ID Film

```
=====
DATA FILM LAYAR LEBAR
=====
1. Tambah Data Awal
2. Tambah Data Akhir
3. Tambah Data Index Tertentu
4. Hapus Data Pertama
5. Hapus Data Terakhir
6. Hapus Data Tertentu
7. Cetak
8. Cari ID Film
9. Urut Data Rating Film-DESC
10. Keluar
=====
8
Cari Data
Masukkan ID Film yang dicari:
1567
Data Id Film: 1567 berada di node ke- 3
IDENTITAS:
ID: 1567
Judul Film: The Dark Knight Rises
IMDB Rating: 8.4
```

9. Urut data rating film-DESC

```
=====
DATA FILM LAYAR LEBAR
=====
1. Tambah Data Awal
2. Tambah Data Akhir
3. Tambah Data Index Tertentu
4. Hapus Data Pertama
5. Hapus Data Terakhir
6. Hapus Data Tertentu
7. Cetak
8. Cari ID Film
9. Urut Data Rating Film-DESC
10. Keluar
=====
9
ID: 1222
Judul Film : Spider-Man: No Way Home
ipk : 8.7
ID: 1567
Judul Film : The Dark Knight Rises
ipk : 8.4
ID: 1765
Judul Film : Skyfall
ipk : 7.8
ID: 1346
Judul Film : Uncharted
ipk : 6.7
ID: 1234
Judul Film : Death on The Nile
ipk : 6.6
```



NAMA : Fatikah Salsabilla
NIM : 2341720003
NO ABSEN : 12
KELAS : 1C
MATERI : Double Linked List (Jobsheet 12)

10. Keluar

```
=====
DATA FILM LAYAR LEBAR
=====
1. Tambah Data Awal
2. Tambah Data Akhir
3. Tambah Data Index Tertentu
4. Hapus Data Pertama
5. Hapus Data Terakhir
6. Hapus Data Tertentu
7. Cetak
8. Cari ID Film
9. Urut Data Rating Film-DESC
10. Keluar
=====
10
PS C:\kuliah\SM 2\PrakASD_1C_12>
```