LAPORAN PRAKTIKUM
MATA KULIAH ALGORITMA DAN STRUKTUR DATA

Dosen Pengampu : Triana Fatmawati, S.T, M.T

PERTEMUAN - 12 - Double Linked List

Nama : **M. Zidna Billah Faza**

NIM : **2341760030**

Prodi : **D-IV Sistem Informasi Bisnis**

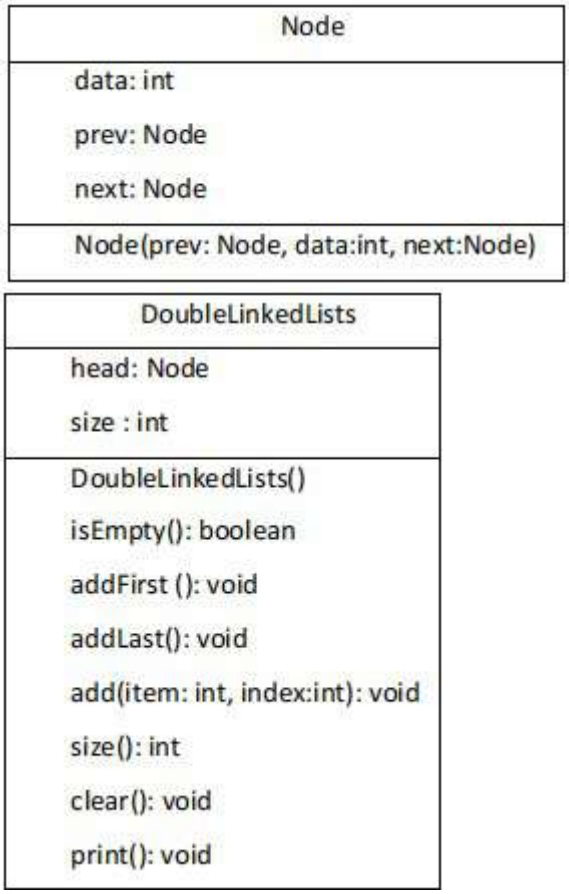
JURUSAN TEKNOLOGI INFORMASI

POLITEKNIK NEGERI MALANG

2024

Percobaan 1

1) Perhatikan diagram class Node dan class DoublelinkedLists di bawah ini! Diagram class ini yang selanjutnya akan dibuat sebagai acuan dalam membuat kode program DoubleLinkedLists.



2) Buat paket baru dengan nama doublelinkedlists
3) Buat class di dalam paket tersebut dengan nama Node

Jobsheet 12 > src > Node_18.java > Node_18

```
1   public class Node_18 {
```

4) Di dalam class tersebut, deklarasikan atribut sesuai dengan diagram class di atas.

```
int data;
Node_18 prev, next;
```

5) Selanjutnya tambahkan konstruktor default pada class Node sesuai diagram di atas.

```
Node_18(Node_18 prev, int data, Node_18 next) {
    this.prev = prev;
    this.data = data;
    this.next = next;
}
```

6) Buatlah sebuah class baru bernama DoubleLinkedLists pada package yang sama dengan node seperti gambar berikut:

7) Pada class DoubleLinkedLists tersebut, deklarasikan atribut sesuai dengan diagram class di atas.

```
Node_18 head;
int size;
```

8) Selajuntnya, buat konstruktor pada class DoubleLinkedLists sesuai gambar berikut

```
public DoubleLinkedLists_18() {
    head = null;
    size = 0;
}
```

9) Buat method isEmpty(). Method ini digunakan untuk memastikan kondisi linked list kosong

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

10) Kemudian, buat method addFirst(). Method ini akan menjalankan penambahan data di bagian depan linked list.

```
public void AddFirst(int item) {

    if (IsEmpty()) {

        head = new Node_18(prev:null, item, next:null);

    } else {

        Node_18 newNode = new Node_18(prev:null, item, head);
        head.prev = newNode;
        head = newNode;

    }

    size++;

}
```

11) Selain itu pembuatan method addLast() akan menambahkan data pada bagian belakang linked list.

```
public void AddLast(int item) {

    if (IsEmpty()) {
        AddFirst(item);
    } else {
        Node_18 cureent = head;
        while (cureent.next != null) {
            cureent = cureent.next;
        }
        Node_18 newNode = new Node_18(cureent, item, next:null);
        cureent.next = newNode;
        size++;
    }

}
```

12) Untuk menambahkan data pada posisi yang telah ditentukan dengan indeks, dapat dibuat dengan method add(int item, int index)

```
public void Add(int item, int index) throws Exception {

    if (IsEmpty()) {

        AddFirst(item);

    } else if (index < 0 || index > size) {

        throw new Exception(message:"Nilai index diluar batas");

    } else {

        Node_18 current = head;
        int i = 0;

        while (i < index) {
            current = current.next;
            i++;
        }

        if (current.prev == null) {

            Node_18 newNode = new Node_18(prev:null, item, current);
            current.prev = newNode;
            head = newNode;

        } else {

            Node_18 newNode = new Node_18(current.prev, item, current);
            newNode.prev = current.prev;
            newNode.next = current;
            current.prev.next = newNode;
            current.prev = newNode;

        }

    }

    size++;

}
```

13) Jumlah data yang ada di dalam linked lists akan diperbarui secara otomatis,sehingga dapat dibuat method size() untuk mendapatkan nilai dari size.

```
public int Size() {
    return size;
}
```

14) Selanjutnya dibuat method clear() untuk menghapus semua isi linked lists, sehingga linked lists dalam kondisi kosong.

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

15) Untuk mencetak isi dari linked lists dibuat method print(). Method ini akan mencetak isi linked lists berapapun size-nya. Jika kosong akan dimunculkan suatu pemberitahuan bahwa linked lists dalam kondisi kosong.

```
public void Print() {
    if (!IsEmpty()) {

        Node_18 tmp = head;

        while (tmp != null) {

            System.out.print(tmp.data + "\t");
            tmp = tmp.next;

        }

        System.out.println(x:"\n=======================================");
        System.out.println(x:"Berhasil diisi");

    } else {

        System.out.println(x:"Linked lists kosong");

    }
}
```

16) Selanjutya dibuat class Main DoubleLinkedListsMain untuk mengeksekusi semua method yang ada pada class DoubleLinkedLists.

```
public class DoubleLinkedListsMain {
```

17) Pada main class pada langkah 16 di atas buatlah object dari class DoubleLinkedLists kemudian eksekusi potongan program berikut ini.

```
DoubleLinkedLists_18 dll = new DoubleLinkedLists_18();
```

18) Verifikasi Hasil Percobaan

```
System.out.println(x:"=======================================");
dll.Print();

System.out.println("Size    : " + dll.size);
System.out.println(x:"\n=======================================");

dll.AddFirst(item:3);
dll.AddLast(item:4);
dll.AddFirst(item:7);
dll.Print();

System.out.println("Size    : " + dll.size);
System.out.println(x:"\n=======================================");

dll.Add(item:40, index:1);
dll.Print();

System.out.println("Size    : " + dll.size);
System.out.println(x:"\n=======================================");

dll.Clear();
dll.Print();

System.out.println("Size    : " + dll.size);
System.out.println(x:"\n=======================================");
```

```
=======================================
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
```

Pertanyaan Percobaan 1

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

Single linked list adalah struktur data di mana setiap node memiliki dua bagian: data yang disimpan dan referensi ke node berikutnya dalam urutan. Artinya, penelusuran hanya bisa dilakukan ke depan.

Sementara itu, double linked list memiliki tiga bagian: data, referensi ke node sebelumnya, dan referensi ke node berikutnya. Dengan demikian, double linked list memungkinkan penelusuran ke depan dan ke belakang. Namun, double linked list memerlukan lebih banyak ruang memori untuk menyimpan referensi tambahan.

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

Atribut next dan prev dalam kelas Node digunakan untuk menunjukkan referensi ke node berikutnya (next) dan node sebelumnya (prev) dalam suatu struktur data yang berurutan

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

```
public DoubleLinkedLists_18() {
    head = null;
    size = 0;
}
```

Inisialisasi atribut head dengan null dan size dengan 0 dalam konstruktor kelas DoubleLinkedLists_18 bertujuan untuk menetapkan kondisi awal dari linked list ganda

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

```
Node_18 newNode = new Node_18(prev:null, item, head);
```

Ini karena node baru akan menjadi node pertama dalam linked list, sehingga tidak memiliki node sebelumnya (prev)

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

Statement tersebut memiliki arti bahwa kita mengatur referensi prev dari node yang sebelumnya menjadi node baru yang ditambahkan

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

```
Node_18 newNode = new Node_18(cureent, item, next:null);
```

Pembuatan objek Node dengan mengisikan parameter prev dengan current dan next dengan null dalam metode addLast() memiliki arti bahwa kita membuat node baru di akhir linked list dan mengatur node sebelumnya (node terakhir sebelum penambahan) sebagai node sebelumnya dari node baru sehingga node baru akan berada di linked list terakhir

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

```
if (current.prev == null) {

    Node_18 newNode = new Node_18(prev:null, item, current);
    current.prev = newNode;
    head = newNode;
```

jelaskan maksud dari bagian tersebut.

Potongan kode tersebut bertujuan untuk menambahkan node baru pada indeks tertentu dalam linked list. Jika indeks yang dimasukkan adalah 0 atau linked list kosong, maka node baru akan ditambahkan di awal linked list. Jika tidak, node baru akan dimasukkan di tengah linked list sesuai dengan index yang kita inputkan

Percobaan 2

1) Buatlah method removeFirst() di dalam class DoubleLinkedLists

```
public void RemoveFirst() throws Exception {

    if (IsEmpty()) {
        throw new Exception(message:"Linked lists masih kosong, tidak dapat dihapus!");
    } else if (size == 1) {
        RemoveLast();
    } else {
        head = head.next;
        head.prev = null;
        size--;
    }

}
```

2) Tambahkan method removeLast() di dalam class DoubleLinkedLists.

```
public void RemoveLast() throws Exception {

    if (IsEmpty()) {
        throw new Exception(message:"Linked lists masih kosong, tidak dapat dihapus!");
    } else if (head.next == null) {
        head = null;
        size--;
        return;
    }

    Node_18 current = head;

    while (current.next.next != null) {
        current = current.next;
    }

    current.next = null;
    size--;

}
```

3) Tambahkan pula method remove(int index) pada class DoubleLinkedLists dan amati hasilnya

```
public void Remove(int index) throws Exception {

    if (IsEmpty() || index >= size) {
        throw new Exception(message:"Linked lists masih kosong, tidak dapat dihapus!");
    } else if (index == 0) {
        RemoveFirst();
    } else {

        Node_18 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--;

    }
}
```

4) Untuk mengeksekusi method yang baru saja dibuat, tambahkan potongan kode program berikut pada main class.

```
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:"\n=======================================");

dll.RemoveFirst();
dll.Print();

System.out.println("Size    : " + dll.size);
System.out.println(x:"\n=======================================");

dll.RemoveLast();
dll.Print();

System.out.println("Size    : " + dll.size);
System.out.println(x:"\n=======================================");

dll.Remove(index:1);
dll.Print();

System.out.println("Size    : " + dll.size);
System.out.println(x:"\n=======================================");
```

5) Verifikasi Hasil Percobaan

```
========================================
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
```

Pertanyaan Percobaan 2

1) Apakah maksud statement berikut pada method removeFirst()?

```
head = head.next;
head.prev = null;
```

Statement tersebut digunakan untuk menggeser "head" list ke elemen berikutnya setelah yang pertama dihapus. Namun, setelah langkah ini, elemen sebelumnya yang merupakan elemen pertama sebelumnya masih memiliki referensi ke elemen yang baru menjadi kepala. Oleh karena itu, pernyataan head.prev = null; dibutuhkan untuk memutuskan referensi mundur dari elemen baru yang menjadi head.

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

Dengan mengecek apakah data setelah head adalah null. Jika setelah head adalah null maka data pada head merupakan data terakhir.

3) Jelaskan alasan potongan kode program di bawah ini tidak cocok untuk perintah remove!

```
Node_18 tmp = head.next;
head.next = tmp.next;
tmp.next.prev = head;
```

Kode tersebut tidak cocok untuk perintah remove karena itu tidak memperhitungkan kemungkinan bahwa elemen yang dihapus adalah elemen terakhir dalam daftar. Ketika Anda mencoba menghapus elemen terakhir, baris kode tmp.next.prev = head; akan menyebabkan NullPointerException karena tmp.next akan berisi null. Itu artinya, tidak ada properti prev yang bisa diakses dari null. Jadi, potongan kode ini hanya akan berhasil jika elemen yang akan dihapus bukan elemen terakhir.

4) Jelaskan fungsi kode program berikut ini pada fungsi remove!

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

Kode tersebut digunakan untuk mengatur kembali pointer (atau referensi) dari elemen sebelum dan sesudah elemen yang dihapus, sehingga elemen yang dihapus dilewati dalam list dan dihapus dari struktur data

Percobaan 3

1) Buatlah method getFirst() di dalam class DoubleLinkedLists untuk mendapatkan data pada awal linked lists.

```
public int GetFirst() throws Exception {
    if (IsEmpty()) {
        throw new Exception(message:"Linked masih kosong");
    }

    return head.data;

}
```

2) Selanjutnya, buatlah method getLast() untuk mendapat data pada akhir linked lists.

```
public int GetLast() throws Exception {
    if (IsEmpty()) {
        throw new Exception(message:"Linked masih kosong");
    }

    Node_18 tmp = head;

    while (tmp.next != null) {
        tmp = tmp.next;
    }

    return tmp.data;

}
```

3) Method get(int index) dibuat untuk mendapatkan data pada indeks tertentu

```
public int Get(int index) throws Exception {
    if (IsEmpty()) {
        throw new Exception(message:"Linked masih kosong");
    }

    Node_18 tmp = head;

    for (int i = 0; i < index; i++) {
        tmp = tmp.next;
    }

    return tmp.data;

}
```

4) Pada main class tambahkan potongan program berikut dan amati hasilnya!

```
dll.Print();

System.out.println("Size    : " + dll.size);
System.out.println(x:"\n=======================================");

dll.AddFirst(item:3);
dll.AddLast(item:4);
dll.AddFirst(item:7);
dll.Print();

System.out.println("Size    : " + dll.size);
System.out.println(x:"\n=======================================");

dll.Add(item:40, index:1);
dll.Print();

System.out.println("Size    : " + dll.size);
System.out.println(x:"\n=======================================");

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 ke-1 pada Linked Lists adalah : " + dll.Get(index:1));
```

```
=======================================
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 ke-1 pada Linked Lists adalah : 40
```

Pertanyaan Percobaan 3

1) Jelaskan method size() pada class DoubleLinkedLists!

Method size() pada kelas DoubleLinkedLists digunakan untuk mengembalikan jumlah elemen yang saat ini ada dalam linked list.

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

Untuk mengatur indeks pada double linked lists agar dimulai dari indeks ke-1, dapat membuat sebuah variabel tambahan yang bertindak sebagai indeks ke-1, seperti `index`, yang diinisialisasi dengan nilai 1. Saat menambahkan elemen pertama ke linked list, set variabel `index` ini ke 1, dan saat menambahkan elemen baru setelahnya, tingkatkan nilai `index` sebesar 1. Ketika mengakses atau menghapus elemen berdasarkan indeks, hitung indeks aktual dalam linked list dengan mengurangi 1 dari nilai indeks yang digunakan di luar linked list. Ini memungkinkan untuk berinteraksi dengan linked list menggunakan indeks yang lebih intuitif dimulai dari 1.

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

- Double Linked Lists, fungsi Add memungkinkan penambahan elemen di depan, di belakang, atau di tengah linked list dengan mudah karena setiap elemen memiliki referensi ke elemen sebelumnya (prev) dan setelahnya (next).

- Single Linked Lists, fungsi Add sebagai penambahan elemen seringkali hanya dimungkinkan di awal (head) atau di akhir (tail) linked list, karena setiap elemen hanya memiliki referensi ke elemen berikutnya. Untuk menambahkan elemen di tengah linked list, Anda harus mencari elemen sebelumnya, yang memerlukan waktu lebih lama karena Anda harus melakukan pencarian dari awal linked list untuk menemukan elemen tersebut.

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

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

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

kode kedua langsung memeriksa status head, sementara kode pertama memeriksa ukuran linked list. Meskipun keduanya memberikan hasil yang sama, pendekatan yang lebih langsung menggunakan head terlihat lebih sederhana dan mudah dipahami.

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)

a) Program (class Antrian)

```
public static class Antrian {

    int noAntrian;
    String nama;
    Antrian prev, next;

    Antrian(Antrian prev, int noAntrian, Antrian next, String nama) {
        this.prev = prev;
        this.noAntrian = noAntrian;
        this.next = next;
        this.nama = nama;
    }

}
```

b) Program (class DoubleLinkedLists)

```java
public static class DoubleLinkedLists {

        Antrian head;
        int size;

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

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

        public void Add(int noAntrian, String nama) {

            if (IsEmpty()) {

                head = new Antrian(null, noAntrian, null, nama);

            } else {

                Antrian newNode = new Antrian(null, noAntrian, head, nama);
                head.prev = newNode;
                head = newNode;

            }

            size++;

        }

        public void Remove() throws Exception {

            if (IsEmpty()) {
                throw new Exception("Linked lists masih kosong, tidak dapat dihapus!");
            } else if (head.next == null) {
                System.out.println("Antrian nomor " + head.noAntrian + " dengan nama " + head.nama + " telah divaksinasi");
                head = null;
                size--;
                return;
            }

            Antrian current = head;

            while (current.next.next != null) {
                current = current.next;
            }

            System.out.println("Antrian nomor " + current.next.noAntrian + " dengan nama " + current.next.nama + " telah divaksinasi");
            current.next = null;
            size--;
        }


        public int Size() {
            return size;
        }

        public void Print() {
            if (!IsEmpty()) {

                Antrian tmp = head;
                System.out.println("=======================================");
                System.out.println("            DAFTAR ANTRIAN");
                System.out.println("=======================================");
                System.out.println("|No Antrian\t|Nama\t\t|");
                System.out.println("---------------------------------------");

                while (tmp != null) {

                    System.out.println("|" + tmp.noAntrian + "\t\t" + "|" + tmp.nama + "\t\t|");
                    tmp = tmp.next;

                }

                System.out.println("\n=======================================");
                System.out.println("Berhasil menambahkan antrian");
                System.out.println("Sisa Antrian    : " + size);

            } else {

                System.out.println("Antrian masih kosong");

            }
        }

    }
```

c) Program (class main)

```
public static void main(String[] args) throws Exception {

        DoubleLinkedLists dll = new DoubleLinkedLists();

        int pilihan = 0;
        int noAntrian;
        String nama;

        do {

            System.out.println("========================================");
            System.out.println("     PENGANTRI VAKSIN EXTRAVAGANZA");
            System.out.println("========================================");
            System.out.println("1. Tambah antrian");
            System.out.println("2. Hapus antrian");
            System.out.println("3. Daftar antrian");
            System.out.println("4. Keluar");
            System.out.println("========================================");
            System.out.print("Masukkan pilihan  : ");
            pilihan = input18.nextInt();

            switch (pilihan) {
                case 1:

                    System.out.println("========================================");
                    System.out.println("            TAMBAH ANTRIAN");
                    System.out.println("========================================");

                    System.out.print("No Antrian  : ");
                    noAntrian = input18.nextInt();
                    System.out.print("Nama Pasien : ");
                    nama = input18String.nextLine();

                    dll.Add(noAntrian, nama);

                    break;

                case 2:

                    dll.Remove();

                    break;

                case 3:

                    dll.Print();

                    break;

                default:

                    System.out.println("========================================");
                    System.out.println("      MASUKKAN PILIHAN YANG BENAR");
                    System.out.println("========================================");

                    break;
            }

        } while (pilihan != 4);

    }
```

d) Output Tambah Antrian

```
=======================================
            DAFTAR ANTRIAN
=======================================
|No Antrian     |Nama           |
---------------------------------------
|3              |Surya          |
|2              |Septa          |
|1              |Satria         |

=======================================
Berhasil menambahkan antrian
Sisa Antrian    : 3
=======================================
     PENGANTRI VAKSIN EXTRAVAGANZA
=======================================
1. Tambah antrian
2. Hapus antrian
3. Daftar antrian
4. Keluar
=======================================
Masukkan pilihan  : 1
=======================================
            TAMBAH ANTRIAN
=======================================
No Antrian  : 4
Nama Pasien : Afiq
=======================================
     PENGANTRI VAKSIN EXTRAVAGANZA
=======================================
1. Tambah antrian
2. Hapus antrian
3. Daftar antrian
4. Keluar
=======================================
Masukkan pilihan  : 3
=======================================
            DAFTAR ANTRIAN
=======================================
|No Antrian     |Nama           |
---------------------------------------
|4              |Afiq           |
|3              |Surya          |
|2              |Septa          |
|1              |Satria         |

=======================================
Berhasil menambahkan antrian
Sisa Antrian    : 4
=======================================
```

e) Output Hapus Antrian

```
========================================
     PENGANTRI VAKSIN EXTRAVAGANZA
========================================
1. Tambah antrian
2. Hapus antrian
3. Daftar antrian
4. Keluar
========================================
Masukkan pilihan  : 2
Antrian nomor 1 dengan nama Satria telah divaksinasi
========================================
     PENGANTRI VAKSIN EXTRAVAGANZA
========================================
1. Tambah antrian
2. Hapus antrian
3. Daftar antrian
4. Keluar
========================================
Masukkan pilihan  : 3
========================================
            DAFTAR ANTRIAN
========================================
|No Antrian     |Nama           |
----------------------------------------
|4              |Afiq           |
|3              |Surya          |
|2              |Septa          |

========================================
Berhasil menambahkan antrian
Sisa Antrian    : 3
========================================
```

Pada antrian selanjutnya berganti menjadi septa sesuai dengan urutan

f) Output Daftar Antrian

Tugas Praktikum 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.

a) Program (class Film)

```
public static class Film {

    int idFilm;
    String judul;
    double rating;
    Film prev, next;

    Film(int idFilm, String judul, double rating, Film prev, Film next) {
        this.idFilm = idFilm;
        this.judul = judul;
        this.rating = rating;
        this.prev = prev;
        this.next = next;
    }

}
```

b) Program (class DoubleLinkedLists dan class main)

Mohon maaf ibu Triana karena kode saya pada class ini banyak sehingga tidak dapat saya tampilkan disini. Ibu bisa melihat hasil outputnya atau melihat repository saya yang akan saya sertakan dibawah

c) Output Tambah Film Awal

```
============================================================
                        DAFTAR FILM
============================================================
| No Film | Judul Film                          | Rating |
------------------------------------------------------------
| 7       | The Avengers                        | 8.80   |
| 6       | Thor                                | 8.70   |
| 4       | Iron Man 2                          | 8.00   |
| 3       | Iron Man                            | 8.40   |
| 2       | Captain Marvel                      | 8.20   |
============================================================
============================================================
                    DATA FILM LAYAR LEBAR
============================================================
1. Tambah Film Awal
2. Tambah Film Akhir
3. Tambah Film Index Tertentu
4. Hapus Film Pertama
5. Hapus Film Terakhir
6. Hapus Film Index Tertentu
7. Tampilkan Daftar Film
8. Cari Film Berdasarkan Id Film
9. Urutkan Daftar Film Berdasarkan Rating (Desc)
10. Cari Film Berdasarkan Judul
11. Hapus Film Berdasarkan Judul
12. Keluar
------------------------------------------------------------
Masukkan Pilihan    : 1
------------------------------------------------------------
                    DATA FILM LAYAR LEBAR
------------------------------------------------------------
ID Film     : 8
Judul Film  : Iron Man 3
Rating Film : 9.0
============================================================
                    DATA FILM LAYAR LEBAR
============================================================
1. Tambah Film Awal
2. Tambah Film Akhir
3. Tambah Film Index Tertentu
4. Hapus Film Pertama
5. Hapus Film Terakhir
6. Hapus Film Index Tertentu
7. Tampilkan Daftar Film
8. Cari Film Berdasarkan Id Film
9. Urutkan Daftar Film Berdasarkan Rating (Desc)
10. Cari Film Berdasarkan Judul
11. Hapus Film Berdasarkan Judul
12. Keluar
============================================================
Masukkan Pilihan    : 7
============================================================
                        DAFTAR FILM
============================================================
| No Film | Judul Film                          | Rating |
------------------------------------------------------------
| 8       | Iron Man 3                          | 9.00   |
| 7       | The Avengers                        | 8.80   |
| 6       | Thor                                | 8.70   |
| 4       | Iron Man 2                          | 8.00   |
| 3       | Iron Man                            | 8.40   |
| 2       | Captain Marvel                      | 8.20   |
============================================================
```

d) Output Tambah Film Akhir

```
==================================================================
Masukkan Pilihan    : 2
==================================================================
                    DATA FILM LAYAR LEBAR
==================================================================
ID Film     : 1
Judul Film  : Captain America The First Avenger
Rating Film : 8.8
==================================================================
                    DATA FILM LAYAR LEBAR
==================================================================
1. Tambah Film Awal
2. Tambah Film Akhir
3. Tambah Film Index Tertentu
4. Hapus Film Pertama
5. Hapus Film Terakhir
6. Hapus Film Index Tertentu
7. Tampilkan Daftar Film
8. Cari Film Berdasarkan Id Film
9. Urutkan Daftar Film Berdasarkan Rating (Desc)
10. Cari Film Berdasarkan Judul
11. Hapus Film Berdasarkan Judul
12. Keluar
==================================================================
Masukkan Pilihan    : 7
==================================================================
                        DAFTAR FILM
==================================================================
| No Film | Judul Film                          | Rating |
------------------------------------------------------------------
| 8       | Iron Man 3                          | 9.00   |
| 7       | The Avengers                        | 8.80   |
| 6       | Thor                                | 8.70   |
| 4       | Iron Man 2                          | 8.00   |
| 3       | Iron Man                            | 8.40   |
| 2       | Captain Marvel                      | 8.20   |
| 1       | Captain America The First Avenger   | 8.80   |
==================================================================
```

e) Output Tambah Film Index Tertentu

```
==================================================================
Masukkan Pilihan    : 3
==================================================================
                    DATA FILM LAYAR LEBAR
==================================================================
ID Film     : 5
Judul Film  : The Incredible Hulk
Rating Film : 7.5
Indeks      : 3
==================================================================
                    DATA FILM LAYAR LEBAR
==================================================================
1. Tambah Film Awal
2. Tambah Film Akhir
3. Tambah Film Index Tertentu
4. Hapus Film Pertama
5. Hapus Film Terakhir
6. Hapus Film Index Tertentu
7. Tampilkan Daftar Film
8. Cari Film Berdasarkan Id Film
9. Urutkan Daftar Film Berdasarkan Rating (Desc)
10. Cari Film Berdasarkan Judul
11. Hapus Film Berdasarkan Judul
12. Keluar
==================================================================
Masukkan Pilihan    : 7
==================================================================
                        DAFTAR FILM
==================================================================
| No Film | Judul Film                          | Rating |
------------------------------------------------------------------
| 8       | Iron Man 3                          | 9.00   |
| 7       | The Avengers                        | 8.80   |
| 6       | Thor                                | 8.70   |
| 5       | The Incredible Hulk                 | 7.50   |
| 4       | Iron Man 2                          | 8.00   |
| 3       | Iron Man                            | 8.40   |
| 2       | Captain Marvel                      | 8.20   |
| 1       | Captain America The First Avenger   | 8.80   |
==================================================================
```

f) Output Hapus Film Pertama

```
============================================================
Masukkan Pilihan    : 4
============================================================
                      FILM YANG DIHAPUS
============================================================
| No Film | Judul Film                              | Rating |
------------------------------------------------------------
| 8       | Iron Man 3                              | 9.00   |
============================================================
============================================================
                    DATA FILM LAYAR LEBAR
============================================================
1. Tambah Film Awal
2. Tambah Film Akhir
3. Tambah Film Index Tertentu
4. Hapus Film Pertama
5. Hapus Film Terakhir
6. Hapus Film Index Tertentu
7. Tampilkan Daftar Film
8. Cari Film Berdasarkan Id Film
9. Urutkan Daftar Film Berdasarkan Rating (Desc)
10. Cari Film Berdasarkan Judul
11. Hapus Film Berdasarkan Judul
12. Keluar
============================================================
Masukkan Pilihan    : 7
============================================================
                        DAFTAR FILM
============================================================
| No Film | Judul Film                              | Rating |
------------------------------------------------------------
| 7       | The Avengers                            | 8.80   |
| 6       | Thor                                    | 8.70   |
| 5       | The Incredible Hulk                     | 7.50   |
| 4       | Iron Man 2                              | 8.00   |
| 3       | Iron Man                                | 8.40   |
| 2       | Captain Marvel                          | 8.20   |
| 1       | Captain America The First Avenger       | 8.80   |
============================================================
```

g) Output Hapus Film Terakhir

```
============================================================
Masukkan Pilihan    : 5
============================================================
                      FILM YANG DIHAPUS
============================================================
| No Film | Judul Film                          | Rating |
------------------------------------------------------------
| 1       | Captain America The First Avenger   | 8.80   |
============================================================
============================================================
                    DATA FILM LAYAR LEBAR
============================================================
1. Tambah Film Awal
2. Tambah Film Akhir
3. Tambah Film Index Tertentu
4. Hapus Film Pertama
5. Hapus Film Terakhir
6. Hapus Film Index Tertentu
7. Tampilkan Daftar Film
8. Cari Film Berdasarkan Id Film
9. Urutkan Daftar Film Berdasarkan Rating (Desc)
10. Cari Film Berdasarkan Judul
11. Hapus Film Berdasarkan Judul
12. Keluar
============================================================
Masukkan Pilihan    : 7
============================================================
                        DAFTAR FILM
============================================================
| No Film | Judul Film                          | Rating |
------------------------------------------------------------
| 7       | The Avengers                        | 8.80   |
| 6       | Thor                                | 8.70   |
| 5       | The Incredible Hulk                 | 7.50   |
| 4       | Iron Man 2                          | 8.00   |
| 3       | Iron Man                            | 8.40   |
| 2       | Captain Marvel                      | 8.20   |
============================================================
```

h) Output Hapus Film Index Tertentu

```
==================================================================
Masukkan Pilihan    : 6
==================================================================
                    HAPUS FILM LAYAR LEBAR
==================================================================
Masukkan indeks film yang akan dihapus    : 1
==================================================================
                        FILM YANG DIHAPUS
==================================================================
| No Film | Judul Film                              | Rating |
------------------------------------------------------------------
| 6       | Thor                                    | 8.70   |
==================================================================
==================================================================
                    DATA FILM LAYAR LEBAR
==================================================================
1. Tambah Film Awal
2. Tambah Film Akhir
3. Tambah Film Index Tertentu
4. Hapus Film Pertama
5. Hapus Film Terakhir
6. Hapus Film Index Tertentu
7. Tampilkan Daftar Film
8. Cari Film Berdasarkan Id Film
9. Urutkan Daftar Film Berdasarkan Rating (Desc)
10. Cari Film Berdasarkan Judul
11. Hapus Film Berdasarkan Judul
12. Keluar
==================================================================
Masukkan Pilihan    : 7
==================================================================
                        DAFTAR FILM
==================================================================
| No Film | Judul Film                              | Rating |
------------------------------------------------------------------
| 7       | The Avengers                            | 8.80   |
| 5       | The Incredible Hulk                     | 7.50   |
| 4       | Iron Man 2                              | 8.00   |
| 3       | Iron Man                                | 8.40   |
| 2       | Captain Marvel                          | 8.20   |
==================================================================
```

i) Output Tampilkan Daftar Film

```
==========================================================================
Masukkan Pilihan    : 7
==========================================================================
                          DAFTAR FILM
==========================================================================
| No Film | Judul Film                                    | Rating |
--------------------------------------------------------------------------
| 7       | The Avengers                                  | 8.80   |
| 5       | The Incredible Hulk                           | 7.50   |
| 4       | Iron Man 2                                    | 8.00   |
| 3       | Iron Man                                      | 8.40   |
| 2       | Captain Marvel                                | 8.20   |
==========================================================================
```

j) Output Cari Film Berdasarkan Id Film

```
==========================================================================
Masukkan Pilihan    : 8
==========================================================================
                       CARI FILM LAYAR LEBAR
==========================================================================
Masukkan ID Film yang akan dicari          : 4
==========================================================================
                       HASIL PENCARIAN FILM
==========================================================================
| No Film | Judul Film                                    | Rating |
--------------------------------------------------------------------------
| 4       | Iron Man 2                                    | 8.00   |
==========================================================================
```

k) Output Urutkan Daftar Film Berdasarkan Rating (Desc)

```
==========================================================================
Masukkan Pilihan    : 9
==========================================================================
            DAFTAR FILM DESCENDING BERDASARKAN RATING
==========================================================================
| No Film | Judul Film                                    | Rating |
--------------------------------------------------------------------------
| 7       | The Avengers                                  | 8.80   |
| 3       | Iron Man                                      | 8.40   |
| 2       | Captain Marvel                                | 8.20   |
| 4       | Iron Man 2                                    | 8.00   |
| 5       | The Incredible Hulk                           | 7.50   |
==========================================================================
```

l) Output Cari Film Berdasarkan Judul (Tambahan)

```
============================================================
Masukkan Pilihan    : 10
============================================================
                    CARI FILM LAYAR LEBAR
============================================================
Masukkan ID Film yang akan dicari         : Iron Man
============================================================
                    HASIL PENCARIAN FILM
============================================================
| No Film | Judul Film                              | Rating |
------------------------------------------------------------
| 3       | Iron Man                                | 8.40   |
============================================================
```

m) Output Hapus Film Berdasarkan Judul (Tambahan)

```
============================================================
Masukkan Pilihan    : 11
============================================================
              HAPUS FILM BERDASARKAN JUDUL
============================================================
Masukkan judul film yang akan dihapus    : The Incredible Hulk
============================================================
                    FILM YANG DIHAPUS
============================================================
| No Film | Judul Film                              | Rating |
------------------------------------------------------------
| 5       | The Incredible Hulk                     | 7.50   |
============================================================
```

```
Masukkan Pilihan    : 7
============================================================
                    DAFTAR FILM
============================================================
| No Film | Judul Film                              | Rating |
------------------------------------------------------------
| 7       | The Avengers                            | 8.80   |
| 4       | Iron Man 2                              | 8.00   |
| 3       | Iron Man                                | 8.40   |
| 2       | Captain Marvel                          | 8.20   |
============================================================
```

n) Output Keluar

```
==================================================================
                    DATA FILM LAYAR LEBAR
==================================================================
1. Tambah Film Awal
2. Tambah Film Akhir
3. Tambah Film Index Tertentu
4. Hapus Film Pertama
5. Hapus Film Terakhir
6. Hapus Film Index Tertentu
7. Tampilkan Daftar Film
8. Cari Film Berdasarkan Id Film
9. Urutkan Daftar Film Berdasarkan Rating (Desc)
10. Cari Film Berdasarkan Judul
11. Hapus Film Berdasarkan Judul
12. Keluar
==================================================================
Masukkan Pilihan    : 12
==================================================================
                        TERIMA KASIH
==================================================================
```

Repository : https://github.com/zidnafaz/Praktikum-Algoritma-Struktur-Data