

Jobsheet 10

Praktikum Algoritma & Struktur Data



Rafi Ody Prasetyo
(2341720180)

D-IV Teknik Informatika
Politeknik Negeri Malang
Semester 2
2024

Praktikum 1

1. SS Hasil Percobaan

```
PS D:\tugas_kuliah\Semester 2\Praktikum Algoritma dan Struktur Data (smt 2)>
& 'C:\Program Files\Java\jdk-20\bin\java.exe' '-XX:+ShowCodeDetailsInExceptio
nMessages' '-cp' 'C:\Users\Broody\AppData\Roaming\Code\User\workspaceStorage\
253dc6d7825267ac2f487d7d477acf8e\redhat.java\jdt_ws\Praktikum Algoritma dan S
truktur Data (smt 2)_70586cb2\bin' 'DoubleLinkedListMain'
Linked Lists Kosong
Size : 0
=====
7
3
4

berhasil diisis
Size : 3
=====
7
40
3
4

berhasil diisis
Size : 4
=====
Linked Lists Kosong
Size : 0
=====
```

2. Github

[https://github.com/rafiody16/Praktikum-Algoritma-dan-Struktur-Data--smt-2-
/tree/main/Jobsheet%2010/Praktikum1](https://github.com/rafiody16/Praktikum-Algoritma-dan-Struktur-Data--smt-2-/tree/main/Jobsheet%2010/Praktikum1)

Pertanyaan Praktikum 1

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

Jawab:

Jika single tidak dapat mengakses mundur ke node sebelumnya, sedangkan double dapat mengakses maju dan mundur antara node-node.

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

Jawab:

Sebagai pointer penunjuk node.

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

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

Jawab:

Atribut head digunakan untuk node pertama pada linked list, sedangkan size untuk menghitung jumlah node pada linked list.

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 digunakan untuk menentukan node pertama, jadi sebelum node pertama belum ada nilai node yang tersimpan di linked list atau null.

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

Jawab:

Untuk menyimpan nilai node `addFirst` pada posisi sebelum/prev head.

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 menyimpan `Node` baru apabila node selanjutnya bernilai null, dan node baru akan disimpan di posisi tersebut.

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);
    newNode.prev = current.prev;
    newNode.next = current;
    current.prev.next = newNode;
    current.prev = newNode;
}
```

Jawab:

Jika nilai pada pointer sebelum sekarang bernilai null, maka `newNode` akan disimpan diposisi pertama atau node tersebut menjadi head.

Praktikum 2

1. SS Hasil Praktikum

```
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
=====
```

2. Github

<https://github.com/rafiody16/Praktikum-Algoritma-dan-Struktur-Data--smt-2-/tree/main/Jobsheet%2010/Praktikum2>

Pertanyaan Praktikum 2

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

head = head.next;

head.prev = null;

Jawab:

Head = head.next berfungsi untuk mengubah elemen pertama yang sebelumnya menjadi head digantikan ke elemen selanjutnya menjadi head. Sedangkan head.prev = null digunakan untuk memastikan elemen yang kita hapus tersebut merupakan head, sehingga statement tersebut akan mengecek elemen sebelumnya adalah bernilai null.

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

Jawab:

Apabila elemen selanjutnya yang akan kita hapus bernilai null.

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:

Perintah remove biasanya dioperasikan pada elemen tertentu dalam struktur data, dan kemudian elemen tersebut dihapus. Kode yang diberikan tidak memiliki mekanisme untuk menentukan elemen mana yang harus dihapus.

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

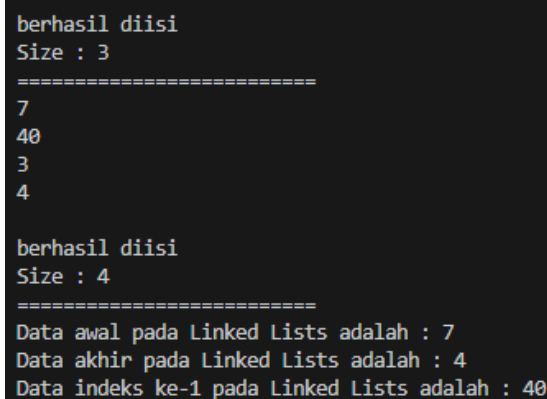
```
current.prev.next = current.next;  
current.next.prev = current.prev;
```

Jawab:

Kedua baris kode tersebut berfungsi untuk menghapus node current dari double linked list dengan menghubungkan node sebelumnya current.prev langsung ke node berikutnya current.next, sehingga node current terlepas dari list.

Praktikum 3

1. SS Hasil Praktikum



```
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
```

2. Github

<https://github.com/rafiody16/Praktikum-Algoritma-dan-Struktur-Data--smt-2-/tree/main/Jobsheet%2010/Praktikum3>

Pertanyaan Praktikum 3

1. Jelaskan method size() pada class DoubleLinkedLists!

Jawab:

Method size() pada class DoubleLinkedLists digunakan untuk mengembalikan jumlah elemen yang ada dalam linked list. Di sini, size adalah variabel yang menyimpan jumlah node dalam linked list.

2. Jelaskan cara mengatur indeks pada double linked lists supaya dapat dimulai dari indeks ke1!

Jawab:

Metode yang mengakses elemen pada linked list harus disesuaikan sehingga indeks 1 mengacu pada elemen pertama, indeks 2 mengacu pada elemen kedua, dan seterusnya.

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

Jawab:

Double linked list dapat diakses maju (next) dan mundur (prev), sehingga operasi seperti penghapusan dan penambahan bisa lebih fleksibel dan efisien di kedua arah. Sedangkan single linked list hanya dapat diakses maju (next), sehingga beberapa operasi bisa menjadi lebih sulit atau memerlukan traversal penuh dari awal.

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)

Jawab:

Logika A berisikan tentang percabangan dengan kondisi apabila size bernilai 0 maka nilai boolean akan mengembalikan true dan else sebaliknya, sedangkan logika B berisikan tentang pengembalian nilai boolean apabila elemen head berisikan null.