

Nama : Muhammad Arafa Yahfazhka

NIM : F1D02310126

Kelas : B

```
class Node {
    int data;
    Node prev;
    Node next;

    Node(int data) {
        this.data = data;
        this.prev = null;
        this.next = null;
    }
}

class DoublyLinkedList {
    Node head;

    public void tambah(int data) {
        Node newNode = new Node(data);
        if (head == null) {
            head = newNode;
            return;
        }
        Node temp = head;
        while (temp.next != null) {
            temp = temp.next;
        }
        temp.next = newNode;
        newNode.prev = temp;
    }

    public void kedepan(int data) {
        Node newNode = new Node(data);
        if (head == null) {
            head = newNode;
            return;
        }
    }
}
```

```

    }
    newNode.next = head;
    head.prev = newNode;
    head = newNode;
}

public void delete(int data) {
    if (head == null) {
        System.out.println("List is empty");
        return;
    }
    Node temp = head;

    if (temp != null && temp.data == data) {
        head = temp.next;
        if (head != null) {
            head.prev = null;
        }
        return;
    }
    while (temp != null && temp.data != data) {
        temp = temp.next;
    }
    if (temp == null) {
        System.out.println(data + " not found in the list.");
        return;
    }
    if (temp.next != null) {
        temp.next.prev = temp.prev;
    }
    if (temp.prev != null) {
        temp.prev.next = temp.next;
    }
}

public void display() {
    Node temp = head;

```

```

        while (temp != null) {
            System.out.print(temp.data + " ");
            temp = temp.next;
        }
        System.out.println();
    }

    public static void main(String[] args) {
        DoublyLinkedList dll = new DoublyLinkedList();
        dll.tambah(10);
        dll.tambah(20);
        dll.tambah(30);
        //=====
        dll.kedepan(5);
        dll.kedepan(1);
        //=====
        System.out.print("Current List: ");
        dll.display(); // Output: 1 5 10 20 30
        dll.delete(10);
        //=====
        System.out.print("After Deleting 10: ");
        dll.display(); // Output: 1 5 20 30
        dll.delete(100); // Output: 100 not found in the list.
    }
}

```

### Penjelasan

Kelas “Node” dalam implementasi ini merepresentasikan elemen individu dalam struktur “Doubly Linked List”. Setiap node menyimpan tiga atribut: “data”, yang merupakan nilai data yang disimpan dalam node, serta “prev” dan “next”, yang berfungsi sebagai referensi ke node sebelumnya dan berikutnya dalam daftar. Konstruktor dari kelas “Node” menerima nilai data dan menetapkan nilai “prev” dan “next” ke “null”, karena pada awalnya node baru tidak terhubung dengan node lain.

Kelas “DoublyLinkedList” bertindak sebagai struktur utama yang memanipulasi dan mengelola daftar berantai ganda. Atribut “head” menyimpan referensi ke node pertama dalam daftar, atau “null” jika daftar kosong. Metode “tambah(int data)” digunakan untuk menambahkan node baru di bagian akhir daftar, sementara “kedepan(int data)” menambahkan node baru di bagian depan

daftar. Keduanya memperbarui referensi “prev” dan “next” sesuai dengan posisi node baru yang ditambahkan. Metode “delete(int data)” bertugas untuk menghapus node dengan nilai data tertentu, baik dari awal, tengah, atau akhir daftar, dengan memperbarui koneksi antar node yang tersisa.

Metode “display()” dalam kelas “DoublyLinkedList” berfungsi untuk menampilkan seluruh elemen dalam daftar secara berurutan dari “head” hingga akhir, mencetak data dari setiap node. Dalam metode “main()”, beberapa operasi dilakukan pada objek “DoublyLinkedList”, termasuk menambah elemen di bagian depan dan belakang, menampilkan daftar, serta menghapus node tertentu. Tes dilakukan untuk memverifikasi bahwa elemen-elemen ditambahkan dan dihapus dengan benar, termasuk penanganan kasus di mana node yang ingin dihapus tidak ditemukan dalam daftar.