

# **LAPORAN PRAKTIKUM**

## **Modul 6**

### **Double Linked List**



**Disusun Oleh:**

**Eduardo Bagus Prima Julian**

**2311104025 S1SE-07-01**

**Dosen :**

**Yudha Islami Sulistya, S.KOM., M.Cs.**

**PROGRAM STUDI S1 SOFTWARE ENGINEERING**

**FAKULTAS INFORMATIKA**

**TELKOM UNIVERSITY PURWOKERTO**

**2024**

## I. Code

1. insertFirst digunakan untuk menambahkan elemen baru di awal list, sedangkan insertLast untuk menambahkan elemen baru di akhir list. displayList menampilkan seluruh list, dari awal hingga akhir.

```
#include <iostream>
using namespace std;

struct Node {
    int data;
    Node* next;
    Node* prev;
};

void insertFirst_2311104013(Node*& head, int value) {
    Node* node = new Node();
    node->data = value;
    node->next = head;
    node->prev = NULL;

    if (head != NULL) {
        head->prev = node;
    }
    head = node;
}

void insertLast_2311104013(Node*& head, int value) {
    Node* node = new Node();
    node->data = value;
    node->next = NULL;
    node->prev = NULL;

    if (head == NULL) {
        head = node;
        return;
    }

    Node* temp = head;
    while (temp->next != NULL) {
        temp = temp->next;
    }

    temp->next = node;
    node->prev = temp;
}

void displayList_2311104013(Node* head) {
    Node* temp = head;
    cout << "Daftar Anggota List: ";
    while (temp != NULL) {
        cout << temp->data;
        if (temp->next != NULL) {
            cout << " ↔ ";
        }
        temp = temp->next;
    }
    cout << endl;
}

int main() {
    Node* head = NULL;
    int value;

    cout << "Masukan elemen pertama: ";
    cin >> value;
    insertFirst_2311104013(head, value);

    cout << "Masukan elemen kedua di awal: ";
    cin >> value;
    insertFirst_2311104013(head, value);

    cout << "Masukan elemen ketiga di akhir: ";
    cin >> value;
    insertLast_2311104013(head, value);

    displayList_2311104013(head);

    return 0;
}
```

2. insertLast untuk menambahkan elemen baru pada akhir list. Selain itu, terdapat deleteFirst untuk menghapus elemen pertama yang ada di dalam list. Kemudian, deleteLast untuk menghapus elemen yang berada di akhir list. displayList menampilkan seluruh elemen setelah penghapusan.

```

#include <iostream>
using namespace std;

struct Node {
    int data;
    Node* next;
    Node* prev;
};

void insertLast_2311104013(Node*& head, int value) {
    Node* node = new Node();
    node->data = value;
    node->next = NULL;

    if (head == NULL) {
        node->prev = NULL;
        head = node;
        return;
    }

    Node* temp = head;
    while (temp->next != NULL) {
        temp = temp->next;
    }

    temp->next = node;
    node->prev = temp;
}

void deleteFirst_2311104013(Node*& head) {
    Node* temp = head;
    head = head->next;

    if (head != NULL) {
        head->prev = NULL;
    }

    delete temp;
}

void deleteLast_2311104013(Node*& head) {
    Node* temp = head;
    if (temp->next == NULL) {
        delete temp;
        head = NULL;
        return;
    }

    while (temp->next != NULL) {
        temp = temp->next;
    }

    temp->prev->next = NULL;
    delete temp;
}

void displayList_2311104013(Node* head) {
    Node* temp = head;
    cout << "Daftar Anggota List setelah dihapus: ";
    while (temp != NULL) {
        cout << temp->data;
        if (temp->next != NULL) {
            cout << " ↔ ";
        }
        temp = temp->next;
    }
    cout << endl;
}

int main() {
    Node* head = NULL;
    int value;

    cout << "Masukan elemen pertama: ";
    cin >> value;
    insertLast_2311104013(head, value);

    cout << "Masukan elemen kedua di akhir: ";
    cin >> value;
    insertLast_2311104013(head, value);

    cout << "Masukan elemen ketiga di akhir: ";
    cin >> value;
    insertLast_2311104013(head, value);

    deleteFirst_2311104013(head);
    deleteLast_2311104013(head);

    displayList_2311104013(head);

    return 0;
}

```

3. insertLast menambahkan elemen baru pada akhir list. displayForward menampilkan elemen dari node pertama hingga akhir, sebaliknya dengan displayBackward untuk menampilkan elemen dari node akhir hingga pertama.

```
#include <iostream>
using namespace std;

struct Node {
    int data;
    Node* next;
    Node* prev;
};

void insertLast_2311104013(Node*& head, int value) {
    Node* node = new Node();
    node->data = value;
    node->next = NULL;

    if (head == NULL) {
        node->prev = NULL;
        head = node;
        return;
    }

    Node* temp = head;
    while (temp->next != NULL) {
        temp = temp->next;
    }

    temp->next = node;
    node->prev = temp;
}

void displayForward_2311104013(Node* head) {
    Node* temp = head;
    cout << "Daftar elemen dari depan ke belakang: ";
    while (temp != NULL) {
        cout << temp->data;
        if (temp->next != NULL) {
            cout << " ↔ ";
        }
        temp = temp->next;
    }
    cout << endl;
}

void displayBackward_2311104013(Node* head) {
    Node* temp = head;
    while (temp->next != NULL) {
        temp = temp->next;
    }

    cout << "Daftar elemen dari belakang ke depan: ";
    while (temp != NULL) {
        cout << temp->data;
        if (temp->prev != NULL) {
            cout << " ↔ ";
        }
        temp = temp->prev;
    }
    cout << endl;
}

int main() {
    Node* head = NULL;
    int value;

    for (int i = 0; i < 4; i++) {
        cout << "Masukan elemen ke-" << i+1 << ": ";
        cin >> value;
        insertLast_2311104013(head, value);
    }

    displayForward_2311104013(head);
    displayBackward_2311104013(head);

    return 0;
}
```