

LAPORAN PRAKTIKUM
STRUKTUR DATA
DOUBLE LINKED LIST BAGIAN 1



Nama :

Reza Irawan (2311104035)

Dosen :

Yudha Islami Sulistya, S.Kom., M.Cs.

PROGRAM STUDI S1 REKAYASA PERANGKAT LUNAK
FAKULTAS INFORMATIKA
TELKOM UNIVERSITY PURWOKERTO
2024

A. Soal TP

1. Soal_01

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

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

Node* createNode_2311104035(int data) {
    Node* newNode = new Node();
    newNode->data = data;
    newNode->prev = nullptr;
    newNode->next = nullptr;
    return newNode;
}

void insertFirst_2311104035(Node*& head, int data) {
    Node* newNode = createNode_2311104035(data);
    if (head == nullptr) {
        head = newNode;
    } else {
        newNode->next = head;
        head->prev = newNode;
        head = newNode;
    }
}

void insertLast_2311104035(Node*& head, int data) {
    Node* newNode = createNode_2311104035(data);
    if (head == nullptr) {
        head = newNode;
    } else {
        Node* temp = head;
        while (temp->next != nullptr) {
            temp = temp->next;
        }
        temp->next = newNode;
        newNode->prev = temp;
    }
}

void display_2311104035(Node* head) {
    Node* temp = head;
    while (temp != nullptr) {
        cout << temp->data;
        if (temp->next != nullptr)
            cout << " <-> ";
        temp = temp->next;
    }
    cout << endl;
}

int main() {
    Node* head = nullptr;
    insertFirst_2311104035(head, 10);
    insertFirst_2311104035(head, 5);
    insertLast_2311104035(head, 20);

    cout << "DAFTAR ANGGOTA LIST: ";
    display_2311104035(head);

    return 0;
}
```

Output Kodingan diatas :

```
PS D:\STRUKTUR DATA\06_Double_Lingked_List_Bagian_1> cd "d:\STRUKTUR DATA\06_Double_Lingked_List_Bagian_1"
pp -o Soal01 } ; if ($?) { .\Soal01 }
DAFTAR ANGGOTA LIST: 5 <-> 10 <-> 20
PS D:\STRUKTUR DATA\06_Double_Lingked_List_Bagian_1\TP>
```

2. Soal_02

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

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

Node* createNode_2311104035(int data) {
    Node* newNode = new Node();
    newNode->data = data;
    newNode->prev = nullptr;
    newNode->next = nullptr;
    return newNode;
}

void insertLast_2311104035(Node*& head, int data) {
    Node* newNode = createNode_2311104035(data);
    if (head == nullptr) {
        head = newNode;
    } else {
        Node* temp = head;
        while (temp->next != nullptr) {
            temp = temp->next;
        }
        temp->next = newNode;
        newNode->prev = temp;
    }
}

void deleteFirst_2311104035(Node*& head) {
    if (head == nullptr) return;
    Node* temp = head;
    if (head->next != nullptr) {
        head = head->next;
        head->prev = nullptr;
    } else {
        head = nullptr;
    }
    delete temp;
}

void deleteLast_2311104035(Node*& head) {
    if (head == nullptr) return;
    Node* temp = head;
    if (temp->next == nullptr) {
        head = nullptr;
        delete temp;
    } else {
        while (temp->next != nullptr) {
            temp = temp->next;
        }
        temp->prev->next = nullptr;
        delete temp;
    }
}

void display_2311104035(Node* head) {
    Node* temp = head;
    while (temp != nullptr) {
        cout << temp->data;
        if (temp->next != nullptr)
            cout << " <-> ";
        temp = temp->next;
    }
    cout << endl;
}

int main() {
    Node* head = nullptr;
    insertLast_2311104035(head, 10);
    insertLast_2311104035(head, 15);
    insertLast_2311104035(head, 20);

    deleteFirst_2311104035(head);
    deleteLast_2311104035(head);

    cout << "DAFTAR ANGGOTA LIST SETELAH PENGHAPUSAN: ";
    display_2311104035(head);

    return 0;
}
```

Output Kodingan diatas :

```
PS D:\STRUKTUR DATA\06_Double_Lingked_List_Bagian_1> cd "d:\STRUKTUR DATA\06_
pp -o Soal02 } ; if ($?) { .\Soal02 }
DAFTAR ANGGOTA LIST SETELAH PENGHAPUSAN: 15
PS D:\STRUKTUR DATA\06_Double_Lingked_List_Bagian_1\TP>
```

3. Soal_03

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

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

Node* createNode_2311104035(int data) {
    Node* newNode = new Node();
    newNode->data = data;
    newNode->prev = nullptr;
    newNode->next = nullptr;
    return newNode;
}

void insertLast_2311104035(Node*& head, int data) {
    Node* newNode = createNode_2311104035(data);
    if (head == nullptr) {
        head = newNode;
    } else {
        Node* temp = head;
        while (temp->next != nullptr) {
            temp = temp->next;
        }
        temp->next = newNode;
        newNode->prev = temp;
    }
}

void displayForward_2311104035(Node* head) {
    Node* temp = head;
    while (temp != nullptr) {
        cout << temp->data;
        if (temp->next != nullptr)
            cout << " <-> ";
        temp = temp->next;
    }
    cout << endl;
}

void displayBackward_2311104035(Node* head) {
    if (head == nullptr) return;

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

    while (temp != nullptr) {
        cout << temp->data;
        if (temp->prev != nullptr)
            cout << " <-> ";
        temp = temp->prev;
    }
    cout << endl;
}

int main() {
    Node* head = nullptr;
    insertLast_2311104035(head, 1);
    insertLast_2311104035(head, 2);
    insertLast_2311104035(head, 3);
    insertLast_2311104035(head, 4);

    cout << "Daftar elemen dari depan ke belakang: ";
    displayForward_2311104035(head);

    cout << "Daftar elemen dari belakang ke depan: ";
    displayBackward_2311104035(head);

    return 0;
}
```

Output Kodingan diatas :

```
PS D:\STRUKTUR DATA\06_Double_Lingked_List_Bagian_1> cd "d:\STRUKTUR DATA\06_Double_Lingked_List_Bagian_1"
pp -o Soal03 } ; if ($?) { .\Soal03 }
Daftar elemen dari depan ke belakang: 1 <-> 2 <-> 3 <-> 4
Daftar elemen dari belakang ke depan: 4 <-> 3 <-> 2 <-> 1
PS D:\STRUKTUR DATA\06_Double_Lingked_List_Bagian_1\TP>
```


