

Nama = Mochamad Aleandre Moulidouane

NIM = M0403241105

Insert di awal

```
PS D:\User\Documents\Kode\KULIAH ILKOM\Strukdat\hari3> .\lkp
Masukkan data untuk di-insert di awal: 10
Menu:
1. Insert di Awal
2. Insert di Akhir
3. Hapus di Awal
4. Hapus di Akhir
5. Update Data
6. Cetak Linked List
0. Keluar
Pilih operasi (0-6): 1
Masukkan data untuk di-insert di awal: 11
Menu:
1. Insert di Awal
2. Insert di Akhir
3. Hapus di Awal
4. Hapus di Akhir
5. Update Data
6. Cetak Linked List
0. Keluar
Pilih operasi (0-6): 1
Masukkan data untuk di-insert di awal: 12
Menu:
1. Insert di Awal
2. Insert di Akhir
3. Hapus di Awal
4. Hapus di Akhir
5. Update Data
6. Cetak Linked List
0. Keluar
Pilih operasi (0-6): 6
12 11 10
```

Insert di Akhir

```
PS D:\User\Documents\Kode\KULIAH ILKOM\Strukdat\hari3> .\lkp
Masukkan data untuk di-insert di akhir: 10
Menu:
1. Insert di Awal
2. Insert di Akhir
3. Hapus di Awal
4. Hapus di Akhir
5. Update Data
6. Cetak Linked List
0. Keluar
Pilih operasi (0-6): 2
Masukkan data untuk di-insert di akhir: 11
Menu:
1. Insert di Awal
2. Insert di Akhir
3. Hapus di Awal
4. Hapus di Akhir
5. Update Data
6. Cetak Linked List
0. Keluar
Pilih operasi (0-6): 2
Masukkan data untuk di-insert di akhir: 12
Menu:
1. Insert di Awal
2. Insert di Akhir
3. Hapus di Awal
4. Hapus di Akhir
5. Update Data
6. Cetak Linked List
0. Keluar
Pilih operasi (0-6): 6
10 11 12
```

Hapus di Awal

File Edit Selection View Go ... PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS Strukdat ikp - hari3 + - x

```
PS D:\User\Documents\Kode\KULIAH ILKOM\Strukdat\hari3> .\lkp
Pilih operasi (0-6): 2
Masukkan data untuk di-insert di akhir: 12

Menu:
1. Insert di Awal
2. Insert di Akhir
3. Hapus di Awal
4. Hapus di Akhir
5. Update Data
6. Cetak Linked List
0. Keluar
Pilih operasi (0-6): 6
10 11 12

Menu:
1. Insert di Awal
2. Insert di Akhir
3. Hapus di Awal
4. Hapus di Akhir
5. Update Data
6. Cetak Linked List
0. Keluar
Pilih operasi (0-6): 3

Menu:
1. Insert di Awal
2. Insert di Akhir
3. Hapus di Awal
4. Hapus di Akhir
5. Update Data
6. Cetak Linked List
0. Keluar
Pilih operasi (0-6): 6
11 12
```

main* 0△0 Ln 165, Col 17 Spaces: 4 UTF-8 CRLF {} C++ Go Live Win32

Hapus di Akhir

File Edit Selection View Go ... PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS Strukdat ikp - hari3 + - x

```
PS D:\User\Documents\Kode\KULIAH ILKOM\Strukdat\hari3> .\lkp
Pilih operasi (0-6): 1
Masukkan data untuk di-insert di awal: 10

Menu:
1. Insert di Awal
2. Insert di Akhir
3. Hapus di Awal
4. Hapus di Akhir
5. Update Data
6. Cetak Linked List
0. Keluar
Pilih operasi (0-6): 6
10 11 12

Menu:
1. Insert di Awal
2. Insert di Akhir
3. Hapus di Awal
4. Hapus di Akhir
5. Update Data
6. Cetak Linked List
0. Keluar
Pilih operasi (0-6): 4

Menu:
1. Insert di Awal
2. Insert di Akhir
3. Hapus di Awal
4. Hapus di Akhir
5. Update Data
6. Cetak Linked List
0. Keluar
Pilih operasi (0-6): 6
10 11
```

main* 0△0 Ln 165, Col 17 Spaces: 4 UTF-8 CRLF {} C++ Go Live Win32

Update Data

```
File Edit Selection View Go ... Strukdat
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS D:\User\Documents\Kode\KULIAH ILKOM\Strukdat\hari3> .\ikp
Menu:
1. Insert di Awal
2. Insert di Akhir
3. Hapus di Awal
4. Hapus di Akhir
5. Update Data
6. Cetak Linked List
0. Keluar
Pilih operasi (0-6): 6
10 11 12

Menu:
1. Insert di Awal
2. Insert di Akhir
3. Hapus di Awal
4. Hapus di Akhir
5. Update Data
6. Cetak Linked List
0. Keluar
Pilih operasi (0-6): 5
Masukkan data yang ingin di-update: 12
Masukkan nilai baru: 120
Data berhasil diupdate.

Menu:
1. Insert di Awal
2. Insert di Akhir
3. Hapus di Awal
4. Hapus di Akhir
5. Update Data
6. Cetak Linked List
0. Keluar
Pilih operasi (0-6): 6
10 11 120
```

Kode

```
#include <iostream>

using namespace std;
```

```
// Definisi struktur Node
```

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

```
// Insert node di awal linked list
```

```
void insertAtBeginning(Node*& head, int newData) {
    Node* newNode = new Node;
    newNode->data = newData;
    newNode->next = head;
    head = newNode;
```

```
}
```

```
// Insert node di akhir linked list

void insertAtEnd(Node*& head, int newData) {

    Node* newNode = new Node;
    newNode->data = newData;
    newNode->next = nullptr;

    if (head == nullptr) {
        head = newNode;
        return;
    }

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

// Hapus node di awal linked list

void deleteAtBeginning(Node*& head) {

    if (head == nullptr) {
        cout << "Linked list kosong. Tidak bisa hapus di awal.\n";
        return;
    }
}
```

```
Node* temp = head;
head = head->next;
delete temp;
}

// Hapus node di akhir linked list

void deleteAtEnd(Node*& head) {
    if (head == nullptr) {
        cout << "Linked list kosong. Tidak bisa hapus di akhir.\n";
        return;
    }

    if (head->next == nullptr) {
        delete head;
        head = nullptr;
        return;
    }

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

// Cari data tertentu dan ganti dengan nilai baru
```

```
void updateData(Node* head, int oldData, int newData) {  
    Node* current = head;  
  
    while (current != nullptr) {  
  
        if (current->data == oldData) {  
  
            current->data = newData;  
  
            cout << "Data berhasil diupdate.\n";  
  
            return;  
  
        }  
  
        current = current->next;  
  
    }  
  
    cout << "Data tidak ditemukan di linked list.\n";  
  
}
```

```
// Cetak seluruh isi linked list  
  
void printLinkedList(Node* head) {  
  
    Node* current = head;  
  
    while (current != nullptr) {  
  
        cout << current->data << ' ';  
  
        current = current->next;  
  
    }  
  
    cout << '\n';  
  
}
```

```
// Hapus seluruh linked list  
  
void deleteLinkedList(Node*& head) {  
  
    while (head != nullptr) {  
  
        Node* temp = head;
```

```
    head = head->next;

    delete temp;

}

}

// Tampilkan menu

void printMenu() {

    cout

        << "\nMenu:\n"
        << "1. Insert di Awal\n"
        << "2. Insert di Akhir\n"
        << "3. Hapus di Awal\n"
        << "4. Hapus di Akhir\n"
        << "5. Update Data\n"
        << "6. Cetak Linked List\n"
        << "0. Keluar\n";

}

int main() {

    Node* head = nullptr;
    int choice;

    do {

        printMenu();
        cout << "Pilih operasi (0-6): ";
        cin >> choice;
```

```
switch (choice) {  
    case 1: {  
        int insertData;  
  
        cout << "Masukkan data untuk di-insert di awal: ";  
  
        cin >> insertData;  
  
        insertAtBeginning(head, insertData);  
  
        break;  
    }  
  
    case 2: {  
        int insertDataEnd;  
  
        cout << "Masukkan data untuk di-insert di akhir: ";  
  
        cin >> insertDataEnd;  
  
        insertAtEnd(head, insertDataEnd);  
  
        break;  
    }  
  
    case 3: {  
        deleteAtBeginning(head);  
  
        break;  
    }  
  
    case 4: {  
        deleteAtEnd(head);  
  
        break;  
    }  
  
    case 5: {  
        int oldData, newData;  
  
        cout << "Masukkan data yang ingin di-update: ";
```

```
        cin >> oldData;

        cout << "Masukkan nilai baru: ";

        cin >> newData;

        updateData(head, oldData, newData);

        break;

    }

case 6: {

    printLinkedList(head);

    break;

}

case 0: {

    cout << "Keluar dari program.\n";

    break;

}

default: {

    cout << "Pilihan tidak valid. Coba lagi.\n";

    break;

}

}

}

} while (choice != 0);

// Hapus seluruh linked list sebelum keluar dari program

deleteLinkedList(head);

return 0;

}
```