

GUIDED

```
#include <iostream>
```

```
using namespace std;
```

```
class Node {
```

```
public:
```

```
    int data;
```

```
    Node* prev;
```

```
    Node* next;
```

```
};
```

```
class DoublyLinkedList {
```

```
public:
```

```
    Node* head;
```

```
    Node* tail;
```

```
    // Constructor untuk inisialisasi head dan tail
```

```
    DoublyLinkedList() {
```

```
        head = nullptr;
```

```
        tail = nullptr;
```

```
    }
```

```
    // Fungsi untuk menambahkan elemen di depan list
```

```
    void insert(int data) {
```

```
        Node* newNode = new Node;
```

```
        newNode->data = data;
```

```
        newNode->prev = nullptr;
```

```
        newNode->next = head;
```

```
        if (head != nullptr) {
```

```
            head->prev = newNode;
```

```
        } else {
```

```
            tail = newNode; // Jika list kosong, tail juga mengarah ke node baru
```

```
        }
```

```
        head = newNode;
```

```
    }
```

```
    // Fungsi untuk menghapus elemen dari depan list
```

```
    void deleteNode() {
```

```
        if (head == nullptr) {
```

```
            return; // Jika list kosong
```

```
        }
```

```
        Node* temp = head;
```

```
        head = head->next;
```

```
        if (head != nullptr) {
```

```
            head->prev = nullptr;
```

```
        } else {
```

```
            tail = nullptr; // Jika hanya satu elemen di list
```

```
        }
```

```
        delete temp; // Hapus elemen
```

```
    }
```

```
    // Fungsi untuk mengupdate data di list
```

```
    bool update(int oldData, int newData) {
```

```
        Node* current = head;
```

```
        while (current != nullptr) {
```

```
            if (current->data == oldData) {
```

```

        current->data = newData;
        return true; // Jika data ditemukan dan diupdate
    }
    current = current->next;
}
return false; // Jika data tidak ditemukan
}

// Fungsi untuk menghapus semua elemen di list
void deleteAll() {
    Node* current = head;
    while (current != nullptr) {
        Node* temp = current;
        current = current->next;
        delete temp;
    }
    head = nullptr;
    tail = nullptr;
}

// Fungsi untuk menampilkan semua elemen di list
void display() {
    Node* current = head;
    while (current != nullptr) {
        cout << current->data << " ";
        current = current->next;
    }
    cout << endl;
}

};

int main() {
    DoublyLinkedList list;
    while (true) {
        cout << "1. Add data" << endl;
        cout << "2. Delete data" << endl;
        cout << "3. Update data" << endl;
        cout << "4. Clear data" << endl;
        cout << "5. Display data" << endl;
        cout << "6. Exit" << endl;

        int choice;
        cout << "Enter your choice: ";
        cin >> choice;

        switch (choice) {
            case 1: {
                int data;
                cout << "Enter data to add: ";
                cin >> data;
                list.insert(data);
                break;
            }
            case 2: {
                list.deleteNode();
                break;
            }
        }
    }
}

```

```

        case 3: {
            int oldData, newData;
            cout << "Enter old data: ";
            cin >> oldData;
            cout << "Enter new data: ";
            cin >> newData;
            bool updated = list.update(oldData, newData);
            if (!updated) {
                cout << "Data not found" << endl;
            }
            break;
        }
        case 4: {
            list.deleteAll();
            break;
        }
        case 5: {
            list.display();
            break;
        }
        case 6: {
            return 0;
        }
        default: {
            cout << "Invalid choice" << endl;
            break;
        }
    }
}
return 0;
}

```

The screenshot shows a Windows command prompt window with the title bar "D:\LAPRAKCPP\Modul 6\Gui". The window contains a menu with six options: 1. Add data, 2. Delete data, 3. Update data, 4. Clear data, 5. Display data, and 6. Exit. Below the menu, the prompt "Enter your choice: |" is displayed, indicating that the user is expected to input a choice.

```

1. Add data
2. Delete data
3. Update data
4. Clear data
5. Display data
6. Exit
Enter your choice: |

```