

Nama : Helmi Efendi Lubis

NIM : 1301223338

DLL.h

```
#ifndef DLL_H_INCLUDED
#define DLL_H_INCLUDED

#include <iostream>
using namespace std;

// typedef int infotype;
struct infotype {
    string Band;
    string judul;
};

typedef struct elmList *address;
struct elmList {
    infotype info;
    address prev;
    address next;
};

struct List {
    address first;
    address last;
};

bool isEmpty_1301223338(List L);
address createNewElmt_1301223338(infotype X, address P);
void insertFirst_1301223338(List &L, address P);
void insertAfter_1301223338(List &L, address &Prec, address &P);
void insertLast_1301223338(List &L, address P);
void deleteFirst_1301223338(List &L, address &P);
void deleteAfter_1301223338(List &L, address Prec, address &P);
void deleteLast_1301223338(List &L, address &P);
void concat_1301223338(List L1, List L2, List &L3);
address findLagu_1301223338(string judul, List L);
void removeLagu_1301223338(string judul, List &L);
void selectMenu_1301223338();
void show_1301223338(List L);

#endif // DLL_H_INCLUDED
```

DLL.cpp

```
#include "DLL.h"
```

```

bool isEmpty_1301223338(List L){
    return L.first == NULL;
}

address createNewElmt_1301223338(infotype X, address P) {
    P = new elmList;
    P->info = X;
    P->next = NULL;
    P->prev = NULL;
    return P;
}

void insertFirst_1301223338(List &L, address P) {
    if(isEmpty_1301223338(L)){
        L.first = P;
        L.last = P;
    }else {
        P->next = L.first;
        L.first->prev = P;
        L.first = P;
    }
}

void insertAfter_1301223338(List &L, address &Prec, address &P) {
    if(Prec == L.last) {
        insertLast_1301223338(L, P);
    }else {
        P->next = Prec->next;
        P->prev = Prec;
        Prec->next->prev = P;
        Prec->next = P;
    }
}

void insertLast_1301223338(List &L, address P) {
    if(isEmpty_1301223338(L)){
        L.first = P;
        L.last = P;
    }else {
        P->prev = L.last;
        L.last->next = P;
        L.last = P;
    }
}

void deleteFirst_1301223338(List &L, address &P) {
    P = L.first;

```

```

    L.first = P->next;
    P->next = NULL;
    L.first->prev = NULL;
}

void deleteAfter_1301223338(List &L, address Prec, address &P) {
    if(Prec->next == L.last){
        deleteLast_1301223338(L, P);
    }else {
        P = Prec->next;
        Prec->next = P->next;
        P->next->prev = Prec;
        P->next = NULL;
        P->prev = NULL;
    }
}

void deleteLast_1301223338(List &L, address &P) {
    P = L.last;
    L.last = P->prev;
    P->prev = NULL;
    L.last->next = NULL;
}

void concat_1301223338(List L1, List L2, List &L3) {
    L3.first = NULL; L3.last = NULL;
    address p = L1.first;
    address q = L2.first;
    while(p != NULL){
        insertLast_1301223338(L3, p);
        p = p->next;
    }

    while(q != NULL){
        insertLast_1301223338(L3, q);
        q = q->next;
    }
}

address findLagu_1301223338(string judul, List L){
    address p, q, find;
    if (isEmpty_1301223338(L)){
        find = NULL;
    }
    else{
        p = L.first;
        q = L.last;
    }
}

```

```

        while(p->info.judul != judul && q->info.judul != judul && p != q && p->next !=
q){
            p = p->next;
            q = q->next;
        }

        if(p->info.judul == judul) {
            find = p;
        }else if(q->info.judul == judul){
            find = q;
        }else{
            find = NULL;
        }
    }
    return find;
}

void removeLagu_1301223338(string judul, List &L) {
    address check = findLagu_1301223338(judul, L);
    address temp;
    if (isEmpty_1301223338(L))
    {
        cout << "List Kosong" << endl;
    }
    else
    {
        if(L.first == check && L.last == check){
            L.first = NULL;
            L.last = NULL;
        }else {
            if(L.first == check) {
                deleteFirst_1301223338(L, check);
            }else if(L.last == check){
                deleteLast_1301223338(L, check);
            }else {
                deleteAfter_1301223338(L, check, temp);
            }
        }
    }
}

void selectMenu_1301223338() {
    cout << "\n===== MENU =====" << endl;
    cout << "1. Menambah N data baru" << endl;
    cout << "2. Menampilkan semua data" << endl;
    cout << "3. Menggabungkan list1 dan list2" << endl;
    cout << "4. Mencari Lagu di dalam List" << endl;
}

```

```

        cout << "5. Menghapus lagu di dalam list" << endl;
        cout << "0. Exit" << endl;
        cout << "Pilihan menu: ";
    }

void show_1301223338(List L){
    address p = L.first;
    if(isEmpty_1301223338(L)){
        cout << "List Kosong" << endl;
    }else {
        while(p != NULL){
            cout << "Nama Band : " << p->info.Band << " "
                << "Judul Lagu : " << p->info.judul << endl;
            p = p->next;
        }
    }
}

```

Main.cpp

```

#include <iostream>
#include "DLL.h"
#include "DLL.cpp"

using namespace std;

int main()
{
    List L1, L2, L3;
    int n, inpList, inpLoc, pilihan, i = 1;
    address p,q,r;
    infotype x;
    L1.first = NULL; L1.last = NULL;
    L2.first = NULL; L2.last = NULL;

    selectMenu_1301223338();
    cin >> pilihan;
    while (pilihan != 0)
    {
        switch (pilihan)
        {
            case 1:
                cout << "\nMau masukin berapa lagu : ";
                cin >> n;
                while (i <= n)
                {
                    cout << "\nMau input ke-List 1 atau List-2: ";
                    cin >> inpList;

```

```

        cout << "Input ke-" << i << endl;
        cout << "Masukkan nama band : ";
        cin >> x.Band;
        cout << "Masukkan judul lagu : ";
        cin >> x.judul;
        q = createNewElmt_1301223338(x, p);
        cout << "\nInput di:\n 1.Awal \n 2.Tengah \n 3.Akhir";
        cout << "\nSilahkan pilih: ";
        cin >> inpLoc;
        if (inpList == 1)
        {
            if(inpLoc == 1){
                insertFirst_1301223338(L1, q);
            }else if(inpLoc == 2){
                show_1301223338(L1);
                cout << "Mau di-input setelah lagu apa: ";
                cin >> x.judul;
                r = findLagu_1301223338(x.judul, L1);
                insertAfter_1301223338(L1, r, q);
            }else {
                insertLast_1301223338(L1, q);
            }
        }else if (inpList == 2){
            if(inpLoc == 1){
                insertFirst_1301223338(L2, q);
            }else if(inpLoc == 2){
                show_1301223338(L2);
                cout << "Mau di-input setelah lagu apa: ";
                cin >> x.judul;
                r = findLagu_1301223338(x.judul, L2);
                insertAfter_1301223338(L2, r, q);
            }else {
                insertLast_1301223338(L2, q);
            }
        }
        i = i + 1;
    }
    i = 1;
    break;
case 2:
    cout << "\nIsi List 1 : " << endl;
    show_1301223338(L1);
    cout << "\nIsi List 2 : " << endl;
    show_1301223338(L2);
    cout << "\nIsi List 3 : " << endl;
    show_1301223338(L3);
    break;
case 3:

```

```

        concat_1301223338(L1, L2, L3);
        cout << "\nIsi List 3 : " << endl;
        show_1301223338(L3);
        break;
    case 4:
        cout << "\nMau cari lagu di: \n1.List-1 \n2.List-2 \n3.List-3";
        cout << "\nSilahkan pilih: ";
        cin >> inpList;
        cout << "\nMasukkan judul lagu : ";
        cin >> x.judul;
        if(inpList == 1){
            p = findLagu_1301223338(x.judul, L1);
        }else if(inpList == 2){
            p = findLagu_1301223338(x.judul, L2);
        }else {
            p = findLagu_1301223338(x.judul, L3);
        }
        if (p != NULL) {
            cout << "Nama Band : " << p->info.Band << " "
                << "Judul Lagu : " << p->info.judul << endl;
        }else {
            cout << "Lagu tidak ditemukan" << endl;
        }
        break;
    case 5:
        cout << "\nMau hapus lagu di: \n1.List-1 \n2.List-2 \n3.List-3";
        cout << "\nSilahkan pilih: ";
        cin >> inpList;
        cout << "\nMasukkan judul lagu : ";
        cin >> x.judul;
        if(inpList == 1){
            removeLagu_1301223338(x.judul, L1);
        }else if(inpList == 2){
            removeLagu_1301223338(x.judul, L2);
        }else {
            removeLagu_1301223338(x.judul, L3);
        }
        break;
    default:
        cout << "\nInput tidak valid" << endl;
        break;
    }
    selectMenu_1301223338();
    cin >> pilihan;
}
cout << "\nTerima kasih" << endl;

```

```
//  
  
}
```

Hasil:

1. Coba procedure insert.

```
Last login: Sat Oct 21 13:13:11 on ttys008  
/Users/mac/Documents/COOLYEAH/KULIAH\ SEMESTER\ 3/Struktur\ Data/Praktikum/TP\ MODUL\ 6/tp6/test ; exit;  
mac@192 ~ % /Users/mac/Documents/COOLYEAH/KULIAH\ SEMESTER\ 3/Struktur\ Data/Praktikum/TP\ MODUL\ 6/tp6/test ; exit;  
  
===== MENU =====  
1. Menambah N data baru  
2. Menampilkan semua data  
3. Menggabungkan list1 dan list2  
4. Mencari Lagu di dalam List  
5. Menghapus lagu di dalam list  
0. Exit  
Pilihan menu: 1  
  
Mau masukin berapa lagu : 5  
  
Mau input ke-List 1 atau List-2: 1  
Input ke-1  
Masukkan nama band : so7  
Masukkan judul lagu : dan  
  
Input di:  
1.Awal  
2.Tengah  
3.Akhir  
Silahkan pilih: 1  
  
Mau input ke-List 1 atau List-2: 1  
Input ke-2  
Masukkan nama band : so7  
Masukkan judul lagu : kita  
  
Input di:  
1.Awal  
2.Tengah  
3.Akhir  
Silahkan pilih: 2  
Nama Band : so7 Judul Lagu : dan  
Mau di-input setelah lagu apa: dan  
  
Mau input ke-List 1 atau List-2: 1  
Input ke-3  
Masukkan nama band : wali  
Masukkan judul lagu : yank
```


Input di:

- 1.Awal
- 2.Tengah
- 3.Akhir

Silahkan pilih: 3

Mau input ke-List 1 atau List-2: 2

Input ke-4

Masukkan nama band : wali

Masukkan judul lagu : dik

Input di:

- 1.Awal
- 2.Tengah
- 3.Akhir

Silahkan pilih: 1

Mau input ke-List 1 atau List-2: 2

Input ke-5

Masukkan nama band : marooon5

Masukkan judul lagu : sugar

Input di:

- 1.Awal
- 2.Tengah
- 3.Akhir

Silahkan pilih: 2

Nama Band : wali Judul Lagu : dik

Mau di-input setelah lagu apa: dik

```
===== MENU =====
1. Menambah N data baru
2. Menampilkan semua data
3. Menggabungkan list1 dan list2
4. Mencari Lagu di dalam List
5. Menghapus lagu di dalam list
0. Exit
Pilihan menu: 2

Isi List 1 :
Nama Band : so7 Judul Lagu : dan
Nama Band : so7 Judul Lagu : kita
Nama Band : wali Judul Lagu : yank

Isi List 2 :
Nama Band : wali Judul Lagu : dik
Nama Band : maroon5 Judul Lagu : sugar

Isi List 3 :
List Kosong

===== MENU =====
1. Menambah N data baru
2. Menampilkan semua data
3. Menggabungkan list1 dan list2
4. Mencari Lagu di dalam List
5. Menghapus lagu di dalam list
0. Exit
Pilihan menu: █
```

2. Coba procedure delete

===== MENU =====

1. Menambah N data baru
2. Menampilkan semua data
3. Menggabungkan list1 dan list2
4. Mencari Lagu di dalam List
5. Menghapus lagu di dalam list
0. Exit

Pilihan menu: 5

Mau hapus lagu di:

- 1.List-1
- 2.List-2
- 3.List-3

Silahkan pilih: 1

Masukkan judul lagu : kita

===== MENU =====

1. Menambah N data baru
2. Menampilkan semua data
3. Menggabungkan list1 dan list2
4. Mencari Lagu di dalam List
5. Menghapus lagu di dalam list
0. Exit

Pilihan menu: 5

Mau hapus lagu di:

- 1.List-1
- 2.List-2
- 3.List-3

Silahkan pilih: 1

Masukkan judul lagu : dan

===== MENU =====

1. Menambah N data baru
2. Menampilkan semua data
3. Menggabungkan list1 dan list2
4. Mencari Lagu di dalam List
5. Menghapus lagu di dalam list
0. Exit

Pilihan menu: 5

```
Mau hapus lagu di:
1.List-1
2.List-2
3.List-3
Silahkan pilih: 2

Masukkan judul lagu : sugar

===== MENU =====
1. Menambah N data baru
2. Menampilkan semua data
3. Menggabungkan list1 dan list2
4. Mencari Lagu di dalam List
5. Menghapus lagu di dalam list
0. Exit
Pilihan menu: 2

Isi List 1 :
Nama Band : so7 Judul Lagu : kita

Isi List 2 :
Nama Band : wali Judul Lagu : dik

Isi List 3 :
List Kosong

===== MENU =====
1. Menambah N data baru
2. Menampilkan semua data
3. Menggabungkan list1 dan list2
4. Mencari Lagu di dalam List
5. Menghapus lagu di dalam list
0. Exit
Pilihan menu: █
```

3. Coba function find

===== MENU =====

1. Menambah N data baru
2. Menampilkan semua data
3. Menggabungkan list1 dan list2
4. Mencari Lagu di dalam List
5. Menghapus lagu di dalam list
0. Exit

Pilihan menu: 4

Mau cari lagu di:

- 1.List-1
- 2.List-2
- 3.List-3

Silahkan pilih: 1

Masukkan judul lagu : kita

Nama Band : so7 Judul Lagu : kita

===== MENU =====

1. Menambah N data baru
2. Menampilkan semua data
3. Menggabungkan list1 dan list2
4. Mencari Lagu di dalam List
5. Menghapus lagu di dalam list
0. Exit

Pilihan menu: 4

Mau cari lagu di:

- 1.List-1
- 2.List-2
- 3.List-3

Silahkan pilih: 2

Masukkan judul lagu : dik

Nama Band : wali Judul Lagu : dik

===== MENU =====

1. Menambah N data baru
2. Menampilkan semua data
3. Menggabungkan list1 dan list2
4. Mencari Lagu di dalam List
5. Menghapus lagu di dalam list
0. Exit

Pilihan menu: █

4. Coba procedure concat

```
===== MENU =====
1. Menambah N data baru
2. Menampilkan semua data
3. Menggabungkan list1 dan list2
4. Mencari Lagu di dalam List
5. Menghapus lagu di dalam list
0. Exit
Pilihan menu: 3

Isi List 3 :
Nama Band : so7 Judul Lagu : kita
Nama Band : wali Judul Lagu : dik

===== MENU =====
1. Menambah N data baru
2. Menampilkan semua data
3. Menggabungkan list1 dan list2
4. Mencari Lagu di dalam List
5. Menghapus lagu di dalam list
0. Exit
Pilihan menu: █
```