

LAPORAN PRAKTIKUM
MODUL 4
SINGLE LINKED LIST (BAGIAN PERTAMA)



Nama :

Novita Syahwa Tri Hapsari (2311104007)

Dosen :

Yudha Islami Sulistya, S.Kom, M.Cs

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

1. Jika dilayar muncul 3 digit nim Anda yang muncul secara terurut TERBALIK dari urutan input.

Jawab:

File list.cpp

```
#include <iostream>
#include "list.h"

using namespace std;

void createList(List &L) {
    L.first = NULL;
}

address allocate(infotype x) {
    address P = new elmlist;
    info(P) = x;
    next(P) = NULL;
    return P;
}

void insertFirst(List &L, address P) {
    next(P) = first(L);
    first(L) = P;
}

void insertLast(List &L, address P) {
    if (first(L) == NULL) {
        first(L) = P;
    } else {
        address Q = first(L);
        while (next(Q) != NULL) {
            Q = next(Q);
        }
        next(Q) = P;
    }
}

void printInfo(List L) {
    address P = first(L);
    while (P != NULL) {
        cout << info(P) << endl;
        P = next(P);
    }

    cout<<endl;
}
```

File list.h

```
#include <iostream>
#define first(L) L.first
#define next(P) P->next
#define info(P) P->info

using namespace std;
typedef int infotype;
typedef struct elmlist *address;

struct elmlist {
    infotype info;
    address next;
};

struct List{
    address first;
};

void createList(List &L);
void insertFirst(List &L, address P);
void insertLast(List &L, address P);
void printInfo(List L);
address allocate(infotype x);
```

Main.cpp

```
#include <iostream>
#include "list.h"

using namespace std;

int main() {
    List L;

    // 1. Panggil Create List
    createList(L);

    // 2. Minta input angka pertama dari user
    int angka1;
    cout << "Masukkan angka pertama: ";
    cin >> angka1;

    // 3. Panggil fungsi allocate
    address P1 = allocate(angka1);

    // 4. Panggil fungsi insertFirst
    insertFirst(L, P1);

    // 5. Panggil fungsi printInfo
    printInfo(L);

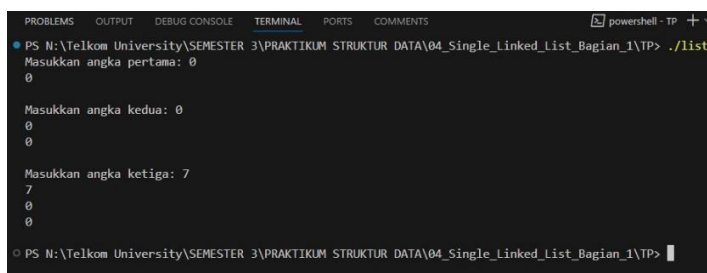
    // 6. Minta input angka kedua dari user
    int angka2;
    cout << "Masukkan angka kedua: ";
    cin >> angka2;

    // Panggil allocate, insertFirst, dan printInfo untuk
    // angka kedua
    address P2 = allocate(angka2);
    insertFirst(L, P2);
    printInfo(L);

    // 7. Minta input angka ketiga dari user
    int angka3;
    cout << "Masukkan angka ketiga: ";
    cin >> angka3;

    // Panggil allocate, insertFirst, dan printInfo untuk
    // angka ketiga
    address P3 = allocate(angka3);
    insertFirst(L, P3);
    printInfo(L);

    return 0;
}
```



```
PS N:\Telkom University\SEMESTER 3\PRAKTIKUM STRUKTUR DATA\04_Single_Linked_List_Bagian_1\TP> ./list
Masukkan angka pertama: 0
0

Masukkan angka kedua: 0
0
0

Masukkan angka ketiga: 7
7
0
0

PS N:\Telkom University\SEMESTER 3\PRAKTIKUM STRUKTUR DATA\04_Single_Linked_List_Bagian_1\TP>
```

Mulai dari list.h, file yang mendefinisikan struktur data daftar terkait dan fungsi-fungsi yang akan digunakan. Fungsi-fungsi ini dapat digunakan dalam file list.cpp, seperti createList untuk memulai list, allocate untuk membuat node baru, insertFirst untuk menambahkan node di awal list, dan printInfo untuk mencetak isi list. File main.cpp kemudian memanggil fungsi-fungsi ini untuk membuat list, menerima input pengguna, menambahkan elemen ke dalam list, dan menampilkan hasilnya. Menjadikan outputnya akan terbalik.

Havefun

```
#include <iostream>
#include "list.h"
using namespace std;

int main() {
    List L;
    createList(L);

    int digit;
    address P;

    // Loop untuk input 10 digit NIM
    for (int i = 1; i ≤ 10; i++) {
        int digit;
        cout << "Digit " << i << " : ";
        cin >> digit;
        address P = allocate(digit);
        insertLast(L, P);
    }

    // Tampilkan isi list (NIM lengkap)
    cout << "Isi list : ";
    printInfo(L);

    return 0;
}
```

```
PS N:\Telkom University\SEMESTER 3\PRAKTIKUM STRUKTUR DATA\04_Single_Linked_List_Bagian_1\TP> ./have
fun
Digit 1 : 1
Digit 2 : 2
Digit 3 : 4
Digit 4 : 4
Digit 5 : 6
Digit 6 : 3
Digit 7 : 9
Digit 8 : 7
Digit 9 : 5
Digit 10 : 6
Isi list : 1
2
4
4
6
3
9
7
5
6
```

Kode di atas adalah program yang meminta pengguna untuk memasukkan 10 digit, yang mewakili NIM, dan menyimpannya dalam daftar yang terkait. Program dimulai dengan membuat daftar kosong menggunakan `createList`, dan kemudian meminta pengguna untuk memasukkan 10 digit satu per satu selama loop. Program menggunakan `printInfo` untuk mencetak isi NIM setelah semua digit yang dimasukkan dimasukkan ke akhir list dengan `insertLast`