

TUGAS PENDAHULUAN
STRUKTUR DATA
MODUL 4



Nama :

Fajar Budiawan (2311104039)

Dosen :

Yudha Islami Sulistya S.Kom,
M.Kom


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

1. Code

List.cpp

```
1  #include "list.h"
2  #include <iostream>
3
4  using namespace std;
5
6  void createList(List &L) {
7      first(L) = NULL;
8  }
9
10 address allocate(infotype x) {
11     address p = new elmlist;
12     info(p) = x;
13     next(p) = NULL;
14     return p;
15 }
16
17 void insertFirst(List &L, address P) {
18     next(P) = first(L);
19     first(L) = P;
20 }
21
22 void printInfo(List L) {
23     address p = first(L);
24     while (p != NULL) {
25         cout << info(p) << ", ";
26         p = next(p);
27     }
28     cout << endl;
29 }
```

List.h



```
1  #ifndef LIST_H
2  #define LIST_H
3
4  #include <iostream>
5  #define first(L) L.first
6  #define next(P) P->next
7  #define info(P) P->info
8
9  using namespace std;
10
11 typedef int infotype;
12 typedef struct elmlist *address;
13
14 struct elmlist {
15     infotype info;
16     address next;
17 };
18
19 struct List {
20     address first;
21 };
22
23 void createList(List &L);
24 address allocate(infotype x);
25 void insertFirst(List &L, address P);
26 void printInfo(List L);
27
28 #endif
```

```

1  #include <iostream>
2  #include "list.h"
3  #include "list.cpp"
4
5  using namespace std;
6
7  int main() {
8      List L;
9      createList(L);
10
11     int data;
12     cout << "Masukkan angka pertama yang ingin diinput ke List: ";
13     cin >> data;
14     address P = allocate(data);
15     insertFirst(L, P);
16     cout << "Isi list setelah input pertama: ";
17     printInfo(L);
18
19     cout << "Masukkan angka kedua yang ingin diinput ke List: ";
20     cin >> data;
21     P = allocate(data);
22     insertFirst(L, P);
23     cout << "Isi list setelah input kedua: ";
24     printInfo(L);
25
26     cout << "Masukkan angka ketiga yang ingin diinput ke List: ";
27     cin >> data;
28     P = allocate(data);
29     insertFirst(L, P);
30     cout << "Isi list setelah input ketiga: ";
31     printInfo(L);
32
33     return 0;
34 }

```

Kode yang diberikan adalah implementasi dari struktur data Linked List dalam bahasa C++ yang terdiri dari tiga file utama: `list.h`, `list.cpp`, dan `main.cpp`. Kode ini digunakan untuk menyimpan dan mengelola elemen-elemen integer menggunakan struktur data singly linked list, di mana elemen-elemen tersebut terhubung melalui pointer.

File list.h berisi definisi struktur dan fungsi yang digunakan dalam implementasi linked list. Terdapat struktur `elmList`, yang merepresentasikan satu elemen dalam list, yang berisi `info` (untuk menyimpan data berupa integer) dan `next` (sebuah pointer yang menunjuk ke elemen berikutnya dalam list). Selain itu, ada struktur `List` yang memiliki pointer `first`, menunjuk ke elemen pertama dari list. Beberapa makro didefinisikan untuk mempermudah akses ke elemen-elemen dari list, seperti `first(L)` untuk mengakses elemen pertama, `next(P)` untuk mengakses elemen berikutnya, dan `info(P)` untuk mengakses data dari sebuah elemen. Fungsi-fungsi yang dideklarasikan di header ini antara lain `createList`, `allocate`, `insertFirst`, dan `printInfo`.

File `list.cpp` mengimplementasikan fungsi-fungsi yang dideklarasikan di `list.h`. Fungsi `createList` menginisialisasi list dengan mengatur pointer `first` menjadi `NULL`, menandakan bahwa list kosong. Fungsi `allocate` digunakan untuk membuat elemen baru dengan data yang diberikan, dan mengembalikan pointer ke elemen baru tersebut. Fungsi `insertFirst` berfungsi untuk menambahkan elemen baru di awal list, dengan mengatur pointer `next` dari elemen baru tersebut menunjuk ke elemen pertama list sebelumnya, dan kemudian mengatur elemen baru sebagai elemen pertama. Terakhir, fungsi `printInfo` digunakan untuk mencetak seluruh elemen dalam list.

File `main.cpp` merupakan fungsi utama program. Dalam fungsi `main`, list pertama kali diinisialisasi menggunakan `createList`. Kemudian, pengguna diminta untuk memasukkan beberapa angka yang akan dimasukkan ke dalam list. Setiap angka yang dimasukkan akan dialokasikan menggunakan fungsi `allocate`, lalu ditambahkan ke awal list menggunakan `insertFirst`. Setelah setiap kali elemen dimasukkan, list dicetak dengan menggunakan fungsi `printInfo` untuk menampilkan elemen-elemen yang ada dalam list saat ini.

Output

```
PS D:\Praktikum Struktur Data\04_Pengenalan_CPP_Bagian_4> cd 'd:\Praktikum Struktur Data\04_Pengenalan_CPP_Bagian_4\TP\output'
PS D:\Praktikum Struktur Data\04_Pengenalan_CPP_Bagian_4\TP\output> & .\'main.exe'
Masukkan angka pertama yang ingin diinput ke List: 1
Isi list setelah input pertama: 1,
Masukkan angka kedua yang ingin diinput ke List: 2
Isi list setelah input kedua: 2, 1,
Masukkan angka ketiga yang ingin diinput ke List: 3
Isi list setelah input ketiga: 3, 2, 1,
PS D:\Praktikum Struktur Data\04_Pengenalan_CPP_Bagian_4\TP\output> |
```