LAPORAN PRAKTIKUM  
MODUL X-A: Double Linked List Circular dengan HEAD

# 1. Judul Praktikum

Double Linked List Circular dengan HEAD

# 2. Tujuan

Mahasiswa memahami dan mampu mengimplementasikan konsep Double Linked List Circular (DLLC) dengan HEAD dalam menyelesaikan permasalahan pemrograman struktural.

# 3. Dasar Teori

- Double Linked List Circular (DLLC) adalah struktur data berbentuk list ganda yang node-nya terhubung secara melingkar.

- Setiap node memiliki dua pointer: next ke node setelahnya, dan prev ke node sebelumnya.

- Head digunakan sebagai penunjuk awal list, serta sebagai anchor node untuk sirkularitas.

# 4. Alat dan Bahan

- Bahasa Pemrograman: C++

- IDE: Code::Blocks / DevC++ / Visual Studio

- Komputer/laptop dengan compiler C++

# 5. Langkah Praktikum

a. Inisialisasi Struktur Node:  
 struct Node {  
 int data;  
 Node\* next;  
 Node\* prev;  
 };

b. Inisialisasi Head:  
 Node\* head = NULL;

c. Menambah Data di Depan:  
- Buat node baru  
- Hubungkan next dan prev ke head dan node sebelumnya  
- Update head jika list masih kosong

d. Menambah Data di Belakang:  
- Traverse hingga node terakhir  
- Tambahkan node baru  
- Sambungkan kembali circular pointer

e. Menghapus Data di Depan:  
- Pindahkan head ke node selanjutnya  
- Hapus node lama  
- Update pointer circular

f. Menghapus Data di Belakang:  
- Traverse hingga node terakhir  
- Putuskan koneksi  
- Hapus node terakhir

g. Menampilkan Data:  
- Traverse dari head hingga kembali ke head  
- Tampilkan setiap nilai

h. Menghapus Semua Data:  
- Traverse sambil menghapus node satu per satu

# 6. Program dan Output

Contoh Fungsi yang Diimplementasikan:  
void insertDepan(int value);  
void insertBelakang(int value);  
void hapusDepan();  
void hapusBelakang();  
void tampil();  
void hapusSemua();

Pemanggilan Fungsi:  
insertDepan(10);  
insertBelakang(20);  
tampil();  
hapusDepan();  
hapusBelakang();  
tampil();

Output:  
10 -> 20  
List setelah penghapusan:

# 7. Kesimpulan

- DLLC memungkinkan traversal dua arah dan tidak memiliki node null.

- Penggunaan head sebagai penunjuk awal memudahkan operasi circular.

- Cocok untuk aplikasi antrian melingkar atau permainan berbasis giliran.

# 8. Daftar Pustaka

1. Goodrich, M.T., Tamassia, R., Mount, D. (2004). Data Structures and Algorithms in C++. Wiley.

2. Hartono, J. (1992). Konsep Dasar Pemrograman Bahasa C. Yogyakarta: Andi.

3. Wahyudi, B. (2004). Pengantar Struktur Data & Algoritma. Yogyakarta: Andi.

4. Yatini, B., Indra & Nasution, E. (2005). Algoritma & Struktur Data dengan C++. Graha Ilmu.