**LAPORAN TUGAS 10A**

**MATA KULIAH DATA STRUKTUR**

Dosen : Dwi Novia Prasetyanti,S.Kom, M.Cs.



**Dibuat Oleh :**

**Muhammad Hadist Rifannan (240202107)**

**Kelas TI-1D**

**PROGRAM STUDI TEKNIK INFORMATIKA**

**POLITEKNIK NEGERI CILACAP**

**TAHUN AKADEMIK 2024/2025**

**Pendahuluan**

Struktur data merupakan salah satu aspek fundamental dalam ilmu komputer yang berperan penting dalam efisiensi pengolahan data dan pemrograman. Dengan pemahaman yang baik terhadap struktur data, seorang programmer dapat memilih metode penyimpanan dan manipulasi data yang paling sesuai untuk menyelesaikan suatu permasalahan secara optimal.

Dalam praktikum Struktur Data, mahasiswa diajak untuk mengimplementasikan berbagai jenis struktur data seperti array, linked list, stack, queue, tree, dan graph melalui bahasa pemrograman. Praktikum ini bertujuan agar mahasiswa tidak hanya memahami konsep teoritis, tetapi juga mampu mengaplikasikan konsep tersebut dalam bentuk kode program yang efisien dan fungsional.

Melalui laporan ini, akan dibahas secara rinci hasil praktikum yang mencakup tujuan percobaan, langkah-langkah pelaksanaan, hasil pengujian program, serta analisis terhadap kinerja dan keefektifan dari struktur data yang digunakan.

**Isi Laporan**

- 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.

5. Langkah Praktikum



a. Inisialisasi Struktur Node:

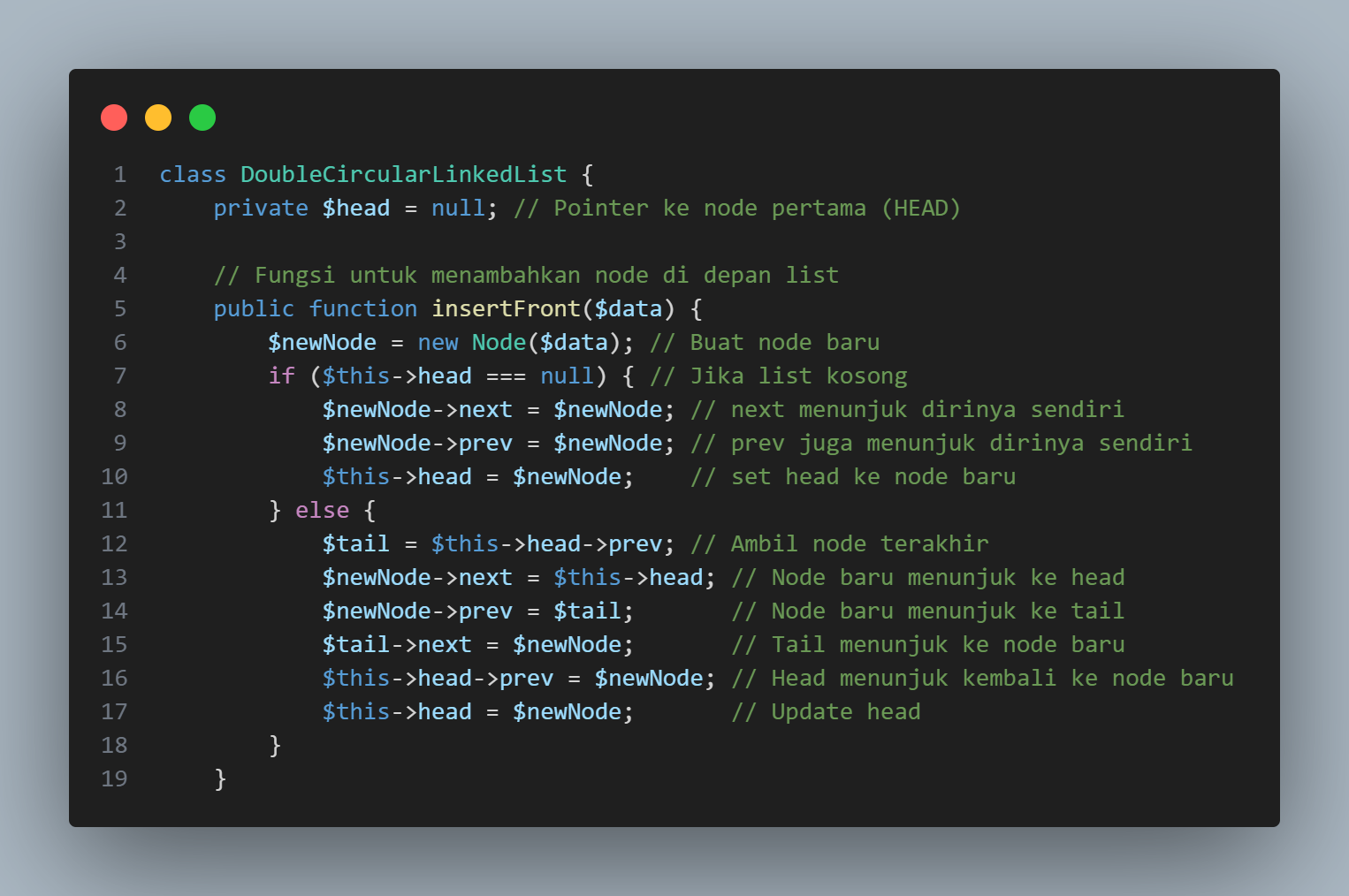
struct Node {

int data;

Node\* next;

Node\* prev;

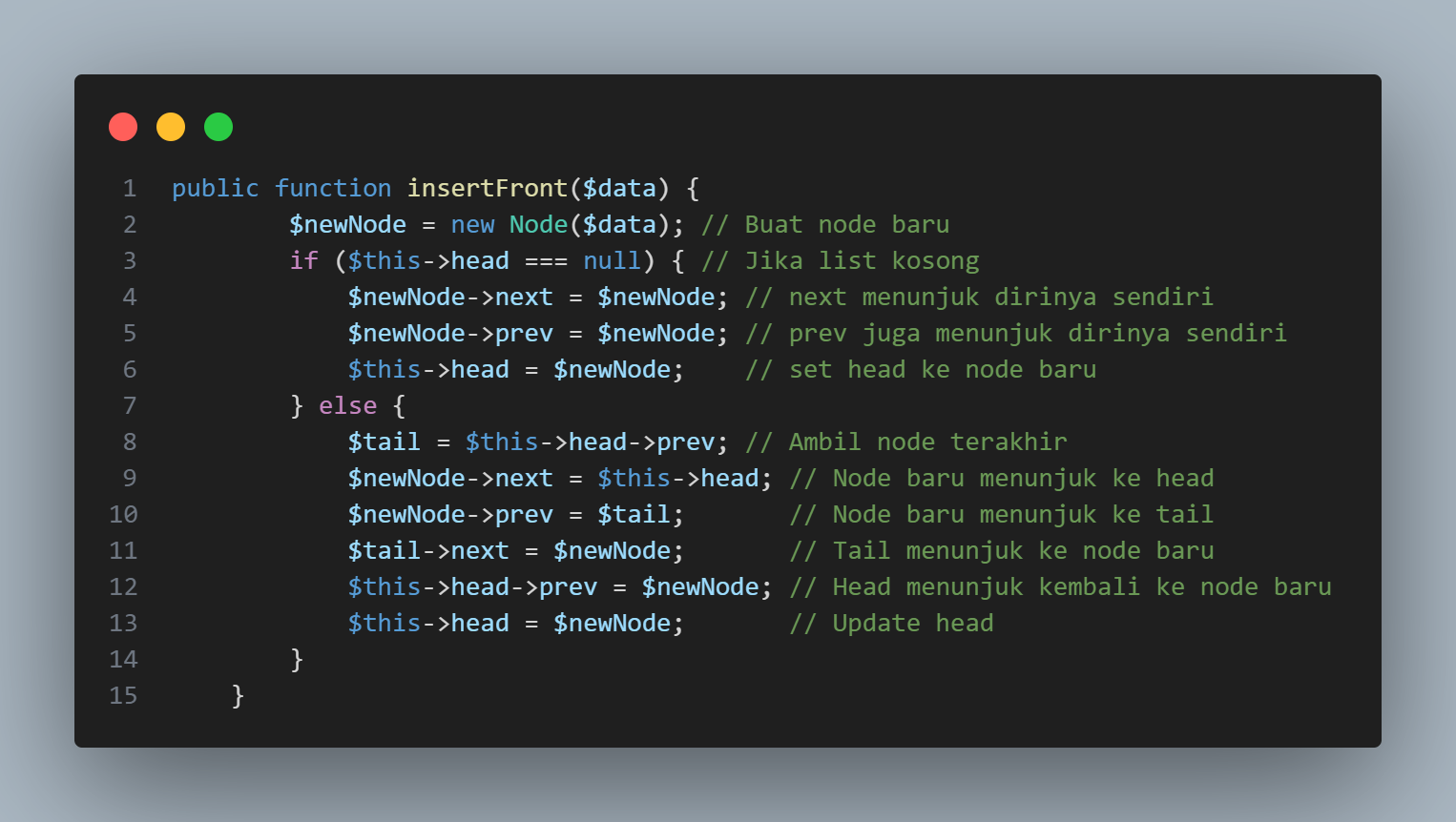
};



b. Inisialisasi Head:

Node\* head = NULL;

c. Menambah Data di Depan:

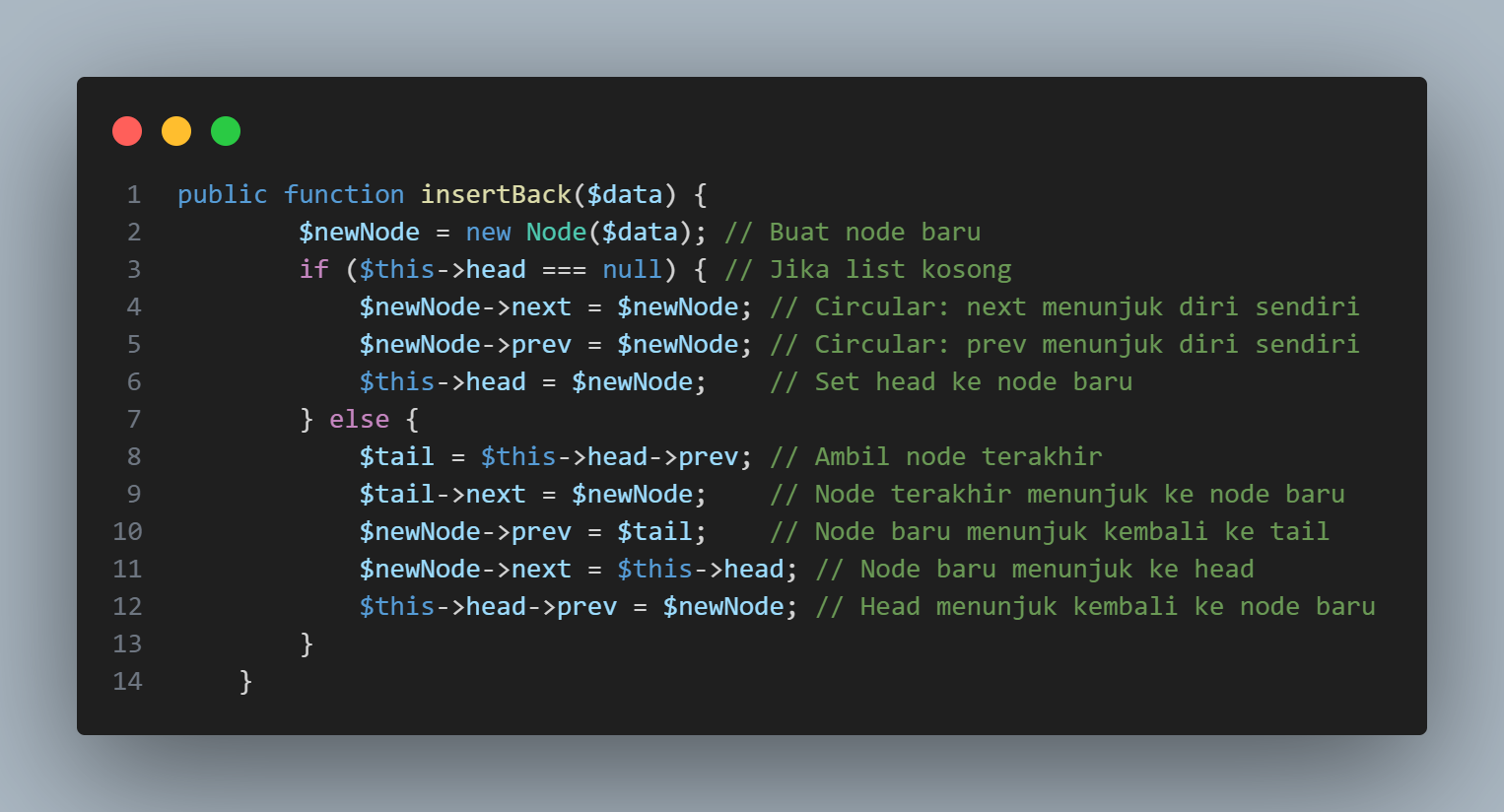


- Buat node baru

- Hubungkan next dan prev ke head dan node sebelumnya

- Update head jika list masih kosong

d. Menambah Data di Belakang:

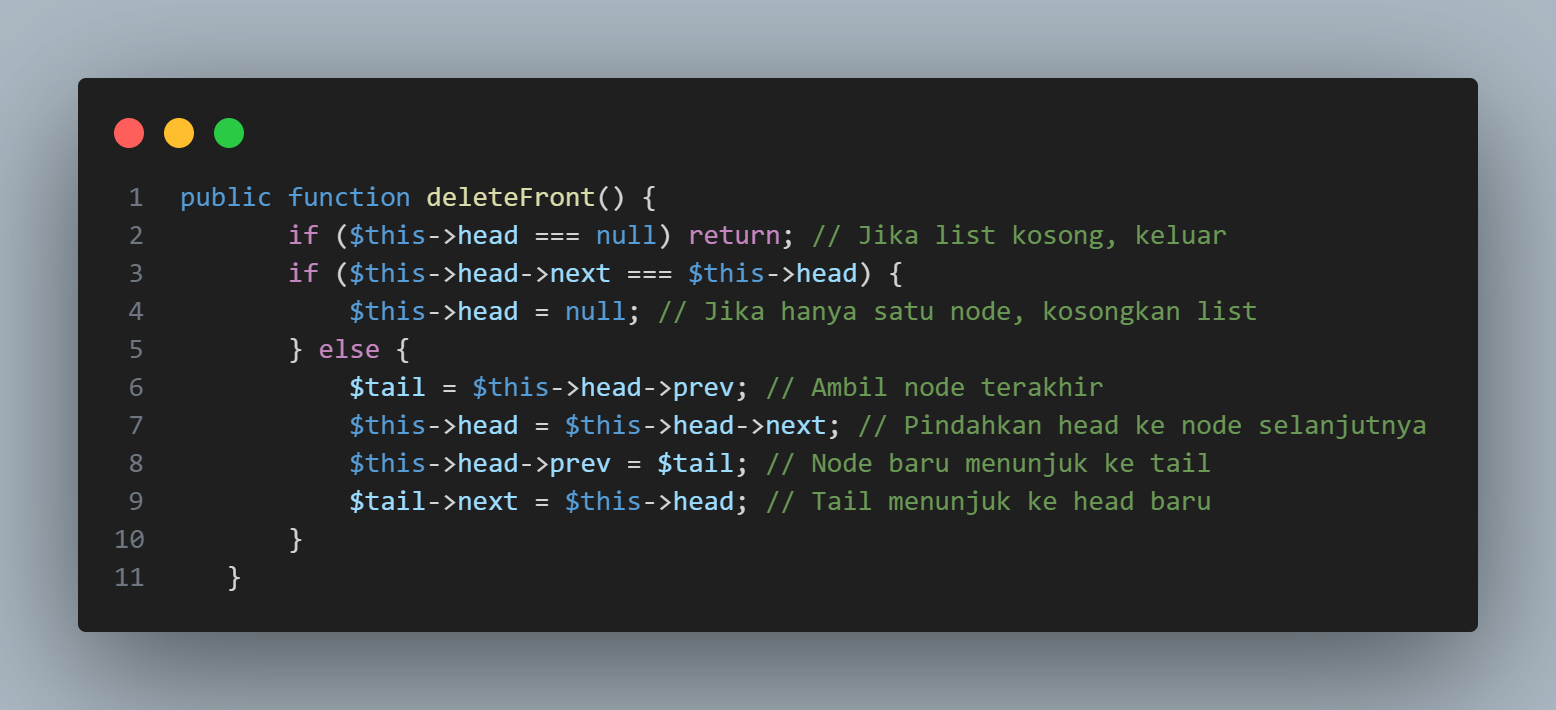


- Traverse hingga node terakhir

- Tambahkan node baru

- Sambungkan kembali circular pointer

e. Menghapus Data di Depan:

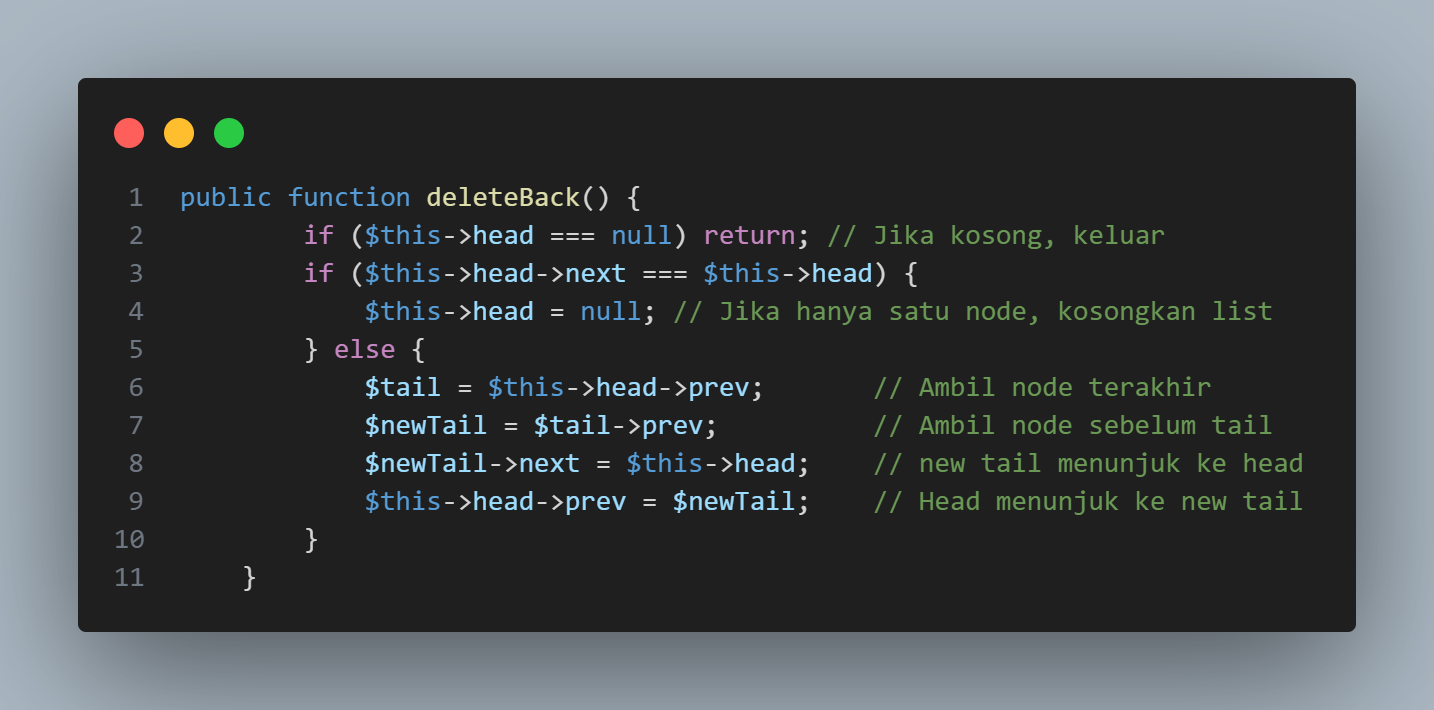


- Pindahkan head ke node selanjutnya

- Hapus node lama

- Update pointer circular

f. Menghapus Data di Belakang:



- Traverse hingga node terakhir

- Putuskan koneksi

- Hapus node terakhir

g. Menampilkan Data:

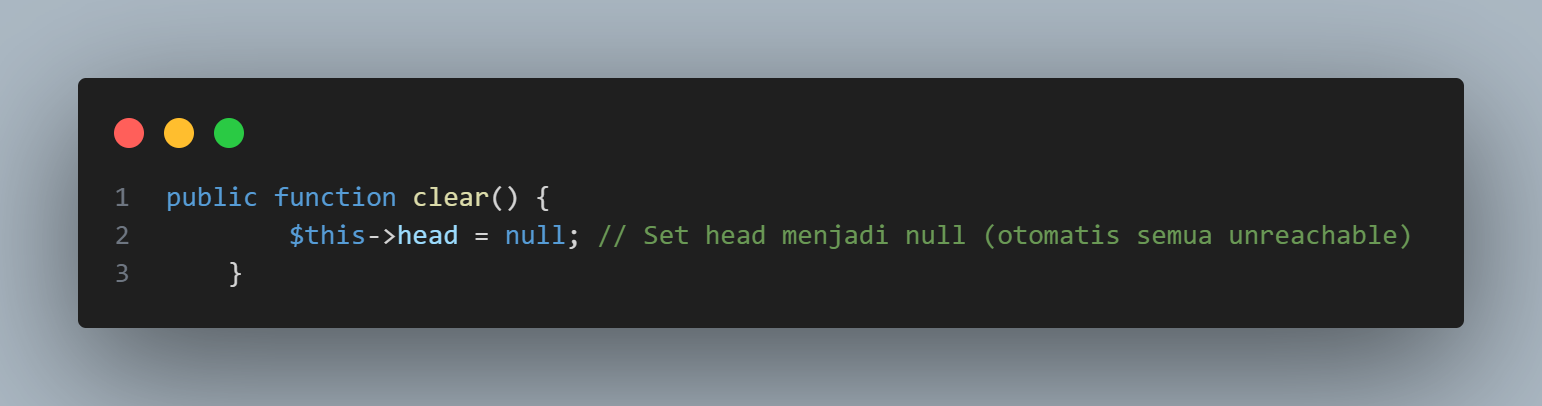
- Traverse dari head hingga kembali ke head

- Tampilkan setiap nilai



h. Menghapus Semua Data:

- Traverse sambil menghapus node satu per satu



6. Program dan Output

Contoh Fungsi yang Diimplementasikan:

insertDepan(value);

insertBelakang(int value);

hapusDepan();

hapusBelakang();

tampil();

hapusSemua();

Pemanggilan Fungsi:

insertDepan(10);

insertBelakang(20);

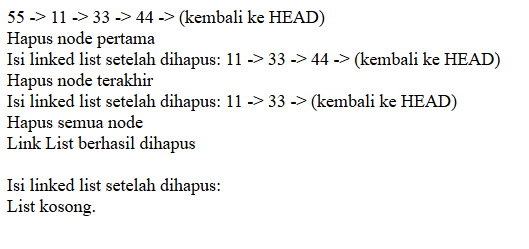
tampil();

hapusDepan();

hapusBelakang();

tampil();

Output:



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.