

## Aturan Praktikum Struktur Data

1. **Akun GitHub:** Setiap praktikan wajib memiliki akun GitHub yang aktif dan digunakan selama praktikum berlangsung.
2. **Invite Collaborator:** Setiap praktikan diwajibkan untuk menambahkan collaborator di setiap repository
  - a. Asisten Praktikum: AndiniNH
  - b. Asisten Praktikum: 4ldiputra
3. **Repository Praktikum:** Setiap praktikan diwajibkan untuk membuat satu repository di GitHub yang akan digunakan untuk seluruh tugas dan laporan praktikum. Repository ini harus diatur dengan rapi dan sesuai dengan instruksi yang akan diberikan di lampiran.
4. **Penamaan Folder:** Penamaan folder dalam repository akan dibahas secara rinci di lampiran. Praktikan wajib mengikuti aturan penamaan yang telah ditentukan.

| Nomor | Pertemuan                            | Penamaan                       |
|-------|--------------------------------------|--------------------------------|
| 1     | Pengalaman Bahasa C++ Bagian Pertama | 01_Pengenalan_CPP_Bagian_1     |
| 2     | Pengenalan Bahasa C++ Bagian Kedua   | 02_Pengenalan_CPP_Bagian_2     |
| 3     | Abstract Data Type                   | 03_Abstract_Data_Type          |
| 4     | Single Linked List Bagian Pertama    | 04_Single_Linked_List_Bagian_1 |
| 5     | Single Linked List Bagian Kedua      | 05_Single_Linked_List_Bagian_2 |
| 6     | Double Linked List Bagian Pertama    | 06_Double_Linked_List_Bagian_1 |
| 7     | Stack                                | 07_Stack                       |
| 8     | Queue                                | 08_Queue                       |
| 9     | Assessment Bagian Pertama            | 09_Assessment_Bagian_1         |
| 10    | Tree Bagian Pertama                  | 10_Tree_Bagian_1               |
| 11    | Tree Bagian Kedua                    | 11_Tree_Bagian_2               |
| 12    | Asistensi Tugas Besar                | 12_Asistensi_Tugas_Besar       |
| 13    | Multi Linked List                    | 13_Multi_Linked_List           |
| 14    | Graph                                | 14_Graph                       |
| 15    | Assessment Bagian Kedua              | 15_Assessment_Bagian_2         |
| 16    | Tugas Besar                          | 16_Tugas_Besar                 |

#### 5. Jam Praktikum:

- Jam masuk praktikum adalah **1 jam lebih lambat** dari jadwal yang tercantum. Sebagai contoh, jika jadwal praktikum adalah pukul 06.30 - 09.30, maka aturan praktikum akan diatur sebagai berikut:
  - **06.30 - 07.30:** Waktu ini digunakan untuk **Tugas Praktikum dan Laporan Praktikum** yang dilakukan di luar laboratorium.
  - **07.30 - 09.30:** Sesi ini mencakup **tutorial, diskusi, dan kasus problem-solving**. Kegiatan ini berlangsung di dalam laboratorium dengan alokasi waktu sebagai berikut:
    - **60 menit pertama:** Tugas terbimbing.
    - **60 menit kedua:** Tugas mandiri.

6. **Pengumpulan Tugasn Pendahuluan:** Tugas Pendahuluan (TP) wajib dikumpulkan melalui GitHub sesuai dengan format berikut:

**nama\_repo/nama\_pertemuan/TP\_Pertemuan\_Ke.md**

Sebagai contoh:

**STD\_Yudha\_Islalmi\_Sulistya\_XXXXXXXX/01\_Running\_Modul/TP\_01.md**

7. **Pengecekan Tugas Pendahuluan:** Pengumpulan laporan praktikum akan diperiksa **1 hari sebelum praktikum selanjutnya** dimulai. Pastikan tugas telah diunggah tepat waktu untuk menghindari sanksi.

## 8. Struktur Laporan Praktikum

### 1. Cover :

**LAPORAN PRAKTIKUM**  
**Modul 4**  
**“SINGLE LINKED LIST (BAGIAN PERTAMA) ”**



**Disusun Oleh:**  
**Reza Afiansyah Wibowo -2311104062**  
**SE0702**

**Dosen :**  
**Arief Rais Bahtiar**

**PROGRAM STUDI S1 SOFTWARE ENGINEERING**  
**FAKULTAS INFORMATIKA**  
**TELKOM UNIVERSITY**  
**PURWOKERTO**  
**2024**

## **Tujuan**

Memahami penggunaan linked list dengan pointer operator- operator dalam program.  
Memahami operasi-operasi dasar dalam linked list.  
Membuat program dengan menggunakan linked list dengan prototype yang ada

## **Landasan Teori**

1. Linked List adalah cara menyimpan data yang saling berhubungan, seperti rangkaian gerbong kereta api yang disambung-sambungkan. Datanya bisa ditambah atau dikurangi sesuai kebutuhan.

2. Linked List bisa menyimpan:

- Data tunggal (misalnya hanya nama saja)
- Data lengkap (misalnya data mahasiswa yang berisi nama, NIM, dan alamat)

1. Ada beberapa jenis Linked List:

- Single Linked List (sambungan satu arah)
- Double Linked List (sambungan dua arah)
- Circular Linked List (sambungan melingkar)
- Dan beberapa jenis lainnya seperti Stack, Queue, Tree, dan Graph

1. Hal-hal yang bisa dilakukan dengan Linked List:

- Membuat list baru
- Menambah data
- Menghapus data
- Melihat isi data
- Mencari data
- Mengubah data

1. Single Linked List punya ciri khas:

- Hanya punya satu penunjuk arah
- Data hanya bisa dibaca ke depan
- Mudah untuk menambah atau menghapus data di tengah-tengah

## Kesimpulan

- Linked List seperti rangkaian gerbong kereta yang bisa ditambah atau dikurangi sesuai kebutuhan. Ini membuat pengelolaan data lebih mudah dan fleksibel dibanding cara lama yang menggunakan array (seperti kotak-kotak tetap).
- Single Linked List adalah jenis Linked List yang paling sederhana karena hanya punya satu arah sambungan. Cocok digunakan kalau sering menambah atau menghapus data di tengah-tengah.
- Dengan Linked List, kita bisa melakukan banyak hal pada data seperti menambah, menghapus, mencari, dan mengubah data dengan mudah sesuai kebutuhan.
- Struktur ini sangat membantu dalam membuat program komputer, terutama untuk aplikasi yang sering melakukan perubahan data, seperti sistem antrian atau daftar tugas.