

LAPORAN TUGAS BESAR

Mata Kuliah: Struktur Data

Telkom University - Semester 3

SISTEM LAYANAN MAGANG (Applicant Tracking System)

Implementasi Multi-Linked List Tipe B

DAFTAR ISI

1. [Pendahuluan](#)
 2. [Analisis Masalah](#)
 3. [Perancangan Struktur Data](#)
 4. [Implementasi Program](#)
 5. [Pengujian dan Validasi](#)
 6. [Kesimpulan](#)
-

1. PENDAHULUAN

1.1 Latar Belakang

Proses pelamaran magang mahasiswa seringkali memerlukan sistem yang dapat mengelola relasi many-to-many antara lowongan pekerjaan dan pelamar. Untuk menyelesaikan permasalahan ini, kami mengimplementasikan **Multi-Linked List (MLL) Tipe B** sebagai struktur data utama yang memungkinkan pengelolaan data yang efisien dan terstruktur.

1.2 Tujuan

1. Mengimplementasikan struktur data Multi-Linked List Tipe B dalam kasus nyata
2. Membangun sistem Applicant Tracking System (ATS) berbasis CLI
3. Menerapkan konsep Role-Based Access Control (RBAC) untuk multi-user
4. Mengintegrasikan fitur otomatis seperti CV Scoring dan notifikasi

1.3 Ruang Lingkup

Sistem ini mencakup:

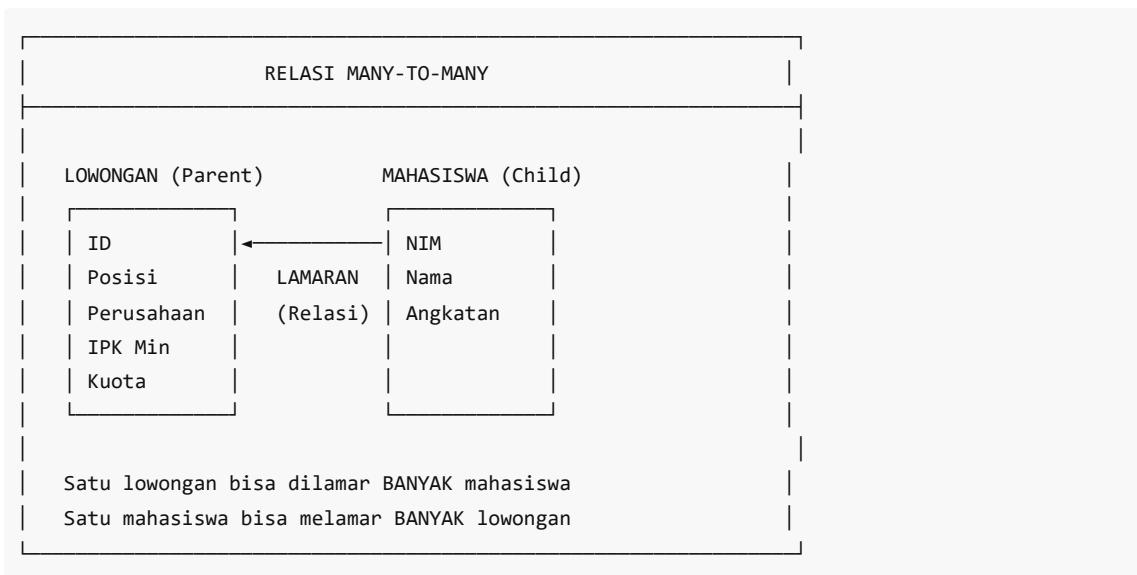
- Pengelolaan data lowongan magang (Parent)
 - Pengelolaan data mahasiswa (Child)
 - Pengelolaan lamaran sebagai relasi (Relasi)
 - Sistem autentikasi multi-role
 - Applicant Tracking System dengan CV Scoring
-

2. ANALISIS MASALAH

2.1 Identifikasi Kebutuhan

Stakeholder	Kebutuhan
Mahasiswa	Melihat lowongan, melamar pekerjaan, upload CV, cek status lamaran
Dosen	Memverifikasi kelayakan mahasiswa untuk magang
Perusahaan	Membuka lowongan, mereview CV, memberikan keputusan
Admin	Mengelola seluruh data sistem dan pengguna

2.2 Relasi Antar Entitas



2.3 Mengapa MLL Tipe B?

MLL Tipe B dipilih karena:

- Setiap Parent (Lowongan) langsung memiliki pointer ke list lamaran miliknya
- Tidak perlu traversal list relasi global untuk mencari lamaran per lowongan
- Akses data lebih cepat untuk operasi yang berpusat pada lowongan
- Cocok untuk kasus dimana query utama adalah "siapa saja pelamar di lowongan X?"

3. PERANCANGAN STRUKTUR DATA

3.1 Definisi Struktur (ADT)

3.1.1 Struktur Parent (Lowongan)

```
struct InfoLowongan {
    int id_lowongan;
    string posisi;
```

```

    string nama_perusahaan;
    float ipk_min;      // IPK Minimum (0.00 - 4.00)
    int kuota;          // Jumlah posisi tersedia
};

struct NodeParent {
    InfoLowongan info;
    address_parent next;
    address_relati first_relati; // Pointer ke lamaran pertama
};

struct ListParent {
    address_parent first;
};

```

3.1.2 Struktur Child (Mahasiswa)

```

struct InfoMahasiswa {
    string nim;
    string nama;
    int angkatan;
};

struct NodeChild {
    InfoMahasiswa info;
    address_child next;
};

struct ListChild {
    address_child first;
};

```

3.1.3 Struktur Relasi (Lamaran)

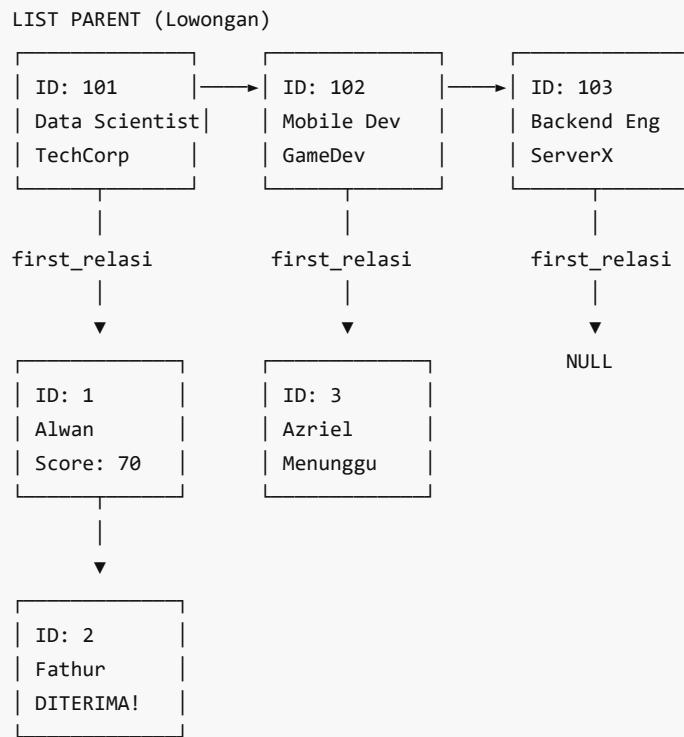
```

struct InfoLamaran {
    int id_lamaran;
    int status_dosen;      // 0: Menunggu, 1: Disetujui
    int status_perusahaan; // 0: Menunggu, 1: Diterima, -1: Ditolak
    string cv_path;
    int cv_score;          // Skor ATS dari analisis CV
};

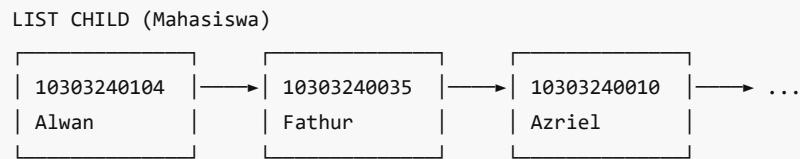
struct NodeRelasi {
    InfoLamaran info;
    address_relati next;
    address_parent ptr_parent; // Pointer ke lowongan
    address_child ptr_child;   // Pointer ke mahasiswa
};

```

3.2 Visualisasi Struktur MLL Tipe B



↓ ptr_child menunjuk ke:



3.3 Penjelasan Pointer

Pointer	Dari	Ke	Fungsi
next	Parent/Child/Relasi	Node sejenis berikutnya	Traversal horizontal
first_relasi	Parent	Relasi pertama	Akses lamaran di lowongan
ptr_parent	Relasi	Parent	Menunjuk lowongan
ptr_child	Relasi	Child	Menunjuk mahasiswa pelamar

4. IMPLEMENTASI PROGRAM

4.1 Arsitektur File

```
tubes-besar/
└── include/
```

```

|   └── MLL_STRUCTS.h      # Definisi struktur data ADT
|   └── lamaran.h          # Header fungsi lamaran
|   └── login.h            # Header fungsi autentikasi
|   └── lowongan.h         # Header fungsi lowongan
|   └── mahasiswa.h        # Header fungsi mahasiswa
|
└── src/
    ├── main.cpp           # Entry point & menu utama
    ├── lamaran.cpp         # Implementasi operasi relasi
    ├── login.cpp           # Implementasi autentikasi
    ├── lowongan.cpp        # Implementasi operasi parent
    └── mahasiswa.cpp       # Implementasi operasi child
|
└── uploads/              # Penyimpanan file CV
└── run.bat               # Script compile & run
└── README.md             # Dokumentasi

```

4.2 Daftar Fungsi Utama

4.2.1 Modul Mahasiswa (`mahasiswa.cpp`)

No	Fungsi	Kategori	Deskripsi
1	<code>createListChild()</code>	Create	Inisialisasi list mahasiswa
2	<code>alokasiChild()</code>	Create	Alokasi node baru mahasiswa
3	<code>insertChild()</code>	Insert	Menambahkan mahasiswa ke list
4	<code>findChildByNIM()</code>	Search	Mencari mahasiswa berdasarkan NIM
5	<code>menuCariMahasiswa()</code>	Search	Menu interaktif pencarian
6	<code>showMahasiswa()</code>	Display	Menampilkan semua mahasiswa
7	<code>handleInputMahasiswa()</code>	Create	Handler input data mahasiswa
8	<code>cleanupRelasiBeforeChildDeletion()</code>	Delete	Cleanup relasi sebelum hapus
9	<code>deleteChildByNIM()</code>	Delete	Hapus mahasiswa & relasinya

4.2.2 Modul Lowongan (`lowongan.cpp`)

No	Fungsi	Kategori	Deskripsi
1	<code>createListParent()</code>	Create	Inisialisasi list lowongan
2	<code>alokasiParent()</code>	Create	Alokasi node baru lowongan
3	<code>insertParent()</code>	Insert	Menambahkan lowongan ke list
4	<code>findParent()</code>	Search	Mencari lowongan berdasarkan ID
5	<code>menuCariLowongan()</code>	Search	Menu interaktif pencarian
6	<code>showLowongan()</code>	Display	Menampilkan semua lowongan
7	<code>menuInsertParent()</code>	Create	Menu input lowongan baru

8	<code>deleteParentByID()</code>	Delete	Hapus lowongan & lamarannya
9	<code>importLowonganFromAPI()</code>	Import	Import data dari file
10	<code>searchJobsOnline()</code>	API	Pencarian lowongan online

4.2.3 Modul Lamaran (`lamaran.cpp`)

No	Fungsi	Kategori	Deskripsi
1	<code>hitungSkorATS()</code>	ATS	Hitung skor CV otomatis
2	<code>uploadFile()</code>	File	Upload CV ke folder lokal
3	<code>insertRelasi()</code>	Insert	Menambahkan lamaran baru
4	<code>deleteRelasi()</code>	Delete	Menghapus/membatalkan lamaran
5	<code>findRelasi()</code>	Search	Cek relasi lowongan-mahasiswa
6	<code>editRelasi()</code>	Update	Mengganti mahasiswa di lamaran
7	<code>editStatusDosen()</code>	Update	Update status verifikasi dosen
8	<code>editStatusPerusahaan()</code>	Update	Update keputusan perusahaan
9	<code>showStatusLamaranMahasiswa()</code>	Display	Status lamaran per mahasiswa
10	<code>showAllChildWithParent()</code>	Display	Tampilkan semua mhs + lowongan
11	<code>showRekapLamaranPerusahaan()</code>	Display	Rekap lamaran dengan ATS Score
12	<code>showRekapLamaranDosen()</code>	Display	Rekap untuk verifikasi dosen
13	<code>countChildPerParent()</code>	Count	Hitung pelamar per lowongan
14	<code>countParentPerChild()</code>	Count	Hitung lowongan per mahasiswa
15	<code>countChildWithoutRelasi()</code>	Count	Hitung mhs belum melamar
16	<code>countParentWithoutRelasi()</code>	Count	Hitung lowongan tanpa pelamar
17	<code>countNotifikasi()</code>	Notification	Hitung jumlah notifikasi
18	<code>showNotifikasi()</code>	Notification	Tampilkan detail notifikasi

4.2.4 Modul Login (`login.cpp`)

No	Fungsi	Kategori	Deskripsi
1	<code>initUsers()</code>	Init	Inisialisasi data pengguna awal
2	<code>AuthenticateUser()</code>	Auth	Validasi username & password
3	<code>Login()</code>	Auth	Proses login dengan UI
4	<code>getNIMByUsername()</code>	Helper	Mendapatkan NIM dari username

5	MainAdmin()	Admin	Menu utama admin
6	KelolaPengguna()	Admin	Kelola data pengguna
7	TambahPenggunaBaru()	Admin	Menambah pengguna baru
8	HapusPengguna()	Admin	Menghapus pengguna
9	UbahDataPengguna()	Admin	Mengubah data pengguna
10	ResetPasswordPengguna()	Admin	Reset password pengguna
11	GantiPasswordAdmin()	Admin	Ganti password admin
12	CatatRiwayatAdmin()	Logging	Mencatat aktivitas admin
13	LihatRiwayatAdmin()	Logging	Melihat log aktivitas

4.3 Fitur Utama

4.3.1 Multi-Role System

Role	Menu Tersedia	Hak Akses
Mahasiswa	7 menu	Input data, lihat lowongan, lamar, cek status, cari, batalkan, notifikasi
Dosen	9 menu	Verifikasi, lihat lowongan, rekap, cari/hapus mahasiswa, statistik
Perusahaan	9 menu	Input lowongan, keputusan, rekap ATS, cari API, cari/hapus, statistik
Admin	12 menu	Full control semua fitur

4.3.2 Applicant Tracking System (ATS)

```
// Algoritma ATS Scoring
int hitungSkorATS(string filePath) {
    // List keyword yang dicari
    string keywords[] = {"C++", "C#", "Java", "Python", "JavaScript",
        "SQL", "HTML", "CSS", "Git", "Linux",
        "Project", "Leadership", "Teamwork", "Communication",
        "Problem-solving", "Research", "Internship"};

    // Baca file (TXT/PDF binary)
    // Untuk setiap keyword ditemukan, tambah skor
    // Return total skor
}
```

4.3.3 Manajemen File CV

- **Upload:** Native Windows File Dialog untuk memilih file
- **Storage:** File disalin ke folder uploads/ dengan nama terstruktur
- **Review:** Perusahaan dapat membuka CV langsung dari aplikasi
- **Cleanup:** Folder uploads dibersihkan saat aplikasi ditutup

4.3.4 Notifikasi Real-time

Mahasiswa mendapat notifikasi saat lamaran diterima:

[!] Anda memiliki 2 pesan baru

Selamat! Lamaran Anda untuk posisi "Data Scientist" di TechCorp telah DITERIMA!

5. PENGUJIAN DAN VALIDASI

5.1 Akun Demo untuk Pengujian

Role	Username	Password
Mahasiswa	muhamadalwansuryadi	0104
Dosen	dosen1	dosen123
Perusahaan	techcorp	123
Admin	admin	admin123

5.2 Skenario Pengujian

Test Case 1: Alur Lengkap Lamaran

Step	Aktor	Aksi	Expected Result	Status
1	Perusahaan	Login & buka lowongan	Lowongan tersimpan	✓
2	Mahasiswa	Login & lihat lowongan	Lowongan tampil	✓
3	Mahasiswa	Upload CV & ajukan lamaran	Lamaran tersimpan, skor ATS dihitung	✓
4	Dosen	Verifikasi lamaran	Status dosen terupdate	✓
5	Perusahaan	Review CV & terima	Status perusahaan terupdate	✓
6	Mahasiswa	Cek notifikasi	Notifikasi diterima muncul	✓

Test Case 2: Operasi CRUD

Operasi	Parent (Lowongan)	Child (Mahasiswa)	Relasi (Lamaran)
Create	✓ insertParent	✓ insertChild	✓ insertRelasi
Read	✓ findParent, showLowongan	✓ findChildByNIM, showMahasiswa	✓ findRelasi, showAllChildWithParent
Update	✓ menuInsertParent (edit)	✓ handleInputMahasiswa	✓ editRelasi, editStatusDosen, editStatusPerusahaan
Delete	✓ deleteParentByID	✓ deleteChildByNIM	✓ deleteRelasi

Test Case 3: Fungsi MLL Tipe B

Fungsi	Deskripsi	Status
countChildPerParent()	Hitung jumlah pelamar per lowongan	✓
countParentPerChild()	Hitung jumlah lowongan per mahasiswa	✓
countChildWithoutRelasi()	Hitung mahasiswa yang belum melamar	✓
countParentWithoutRelasi()	Hitung lowongan tanpa pelamar	✓
showAllChildWithParent()	Tampilkan semua mahasiswa + lowongan yang dilamar	✓

5.3 Hasil Pengujian

Semua test case berhasil dijalankan dengan hasil sesuai ekspektasi.

6. KESIMPULAN

6.1 Hasil yang Dicapai

1. Berhasil mengimplementasikan **MLL Tipe B** dengan struktur Parent-Child-Relasi yang efisien
2. **Sistem multi-role** berfungsi dengan baik untuk 4 tipe pengguna
3. **Fitur ATS** dapat menghitung skor CV secara otomatis
4. Seluruh operasi **CRUD** telah diimplementasikan untuk ketiga entitas
5. **Fungsi statistik dan counting** berjalan sesuai spesifikasi

6.2 Kelebihan Sistem

Aspek	Kelebihan
Struktur Data	MLL Tipe B memungkinkan akses cepat ke relasi per-parent
Modularitas	Kode terpisah per modul (mahasiswa, lowongan, lamaran, login)
UX	Interface CLI yang informatif dengan warna dan formatting
Fitur	ATS scoring, file upload, notifikasi, dan role-based access

6.3 Teknologi yang Digunakan

- **Bahasa Pemrograman:** C++ (Standard Library)
- **Struktur Data:** Multi-Linked List Tipe B
- **Platform:** Windows (menggunakan Windows API untuk file dialog)
- **Compiler:** MinGW / G++

6.4 Cara Menjalankan Program

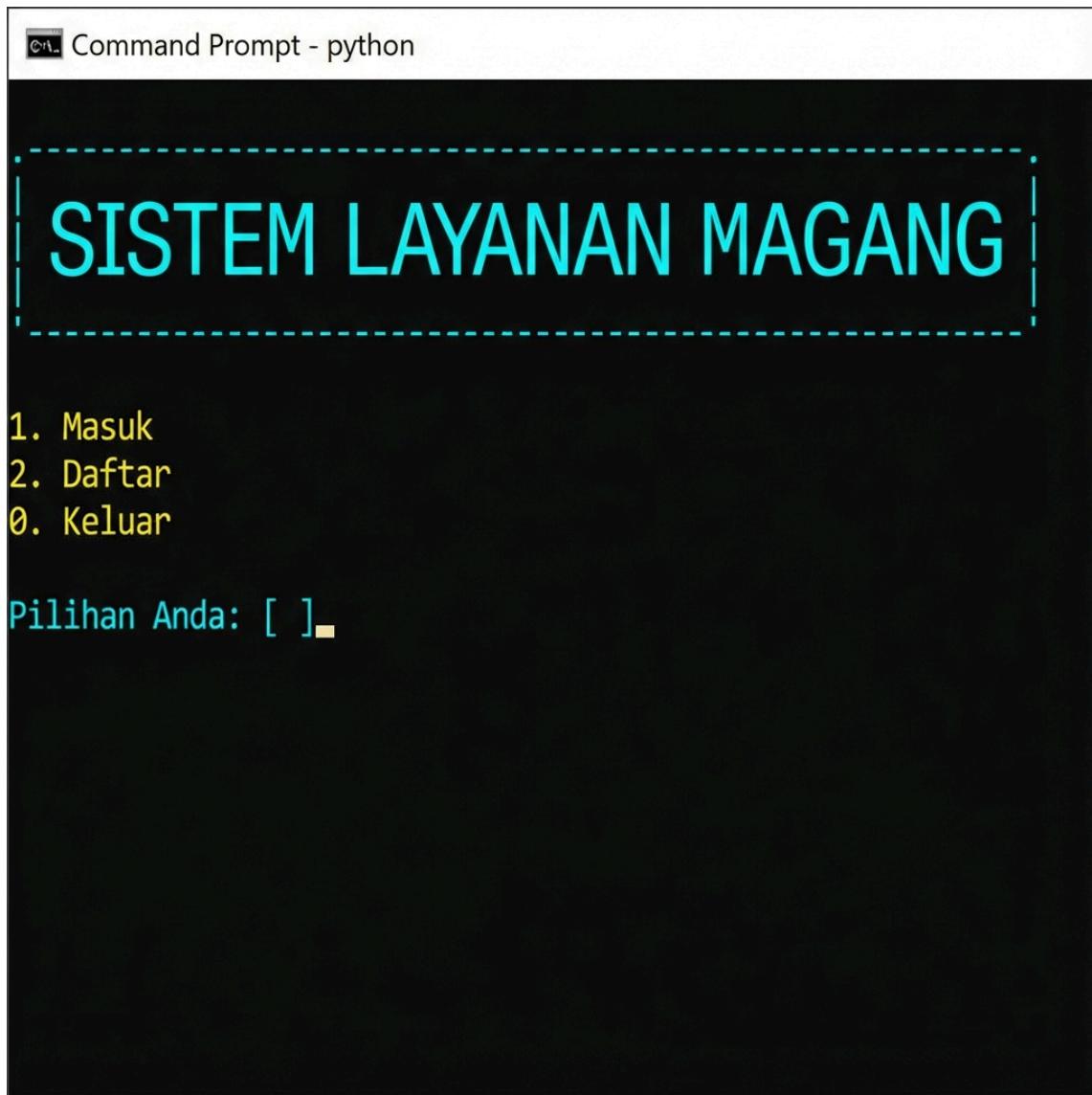
```
# Metode 1: Menggunakan script (Recommended)
./run.bat

# Metode 2: Compile manual
g++ src/*.cpp -Iinclude -o app.exe -lcomdlg32
./app.exe
```

LAMPIRAN

A. Screenshot Tampilan Aplikasi

A.1 Tampilan Login



A.2 Daftar Lowongan

C:\Windows\system32\cmd.exe - Internship CLI

INTERNSHIP JOB LISTINGS MENU

ID	Position	Company	Min GPA	Quota
1	Data Scientist	TechCorp	3.50	5
2	Mobile Dev	GameDev Studio	3.00	3
3	Software Engineer	CloudNet Solutions	3.25	4
4	AI Research Intern	FutureTech Labs	3.75	2

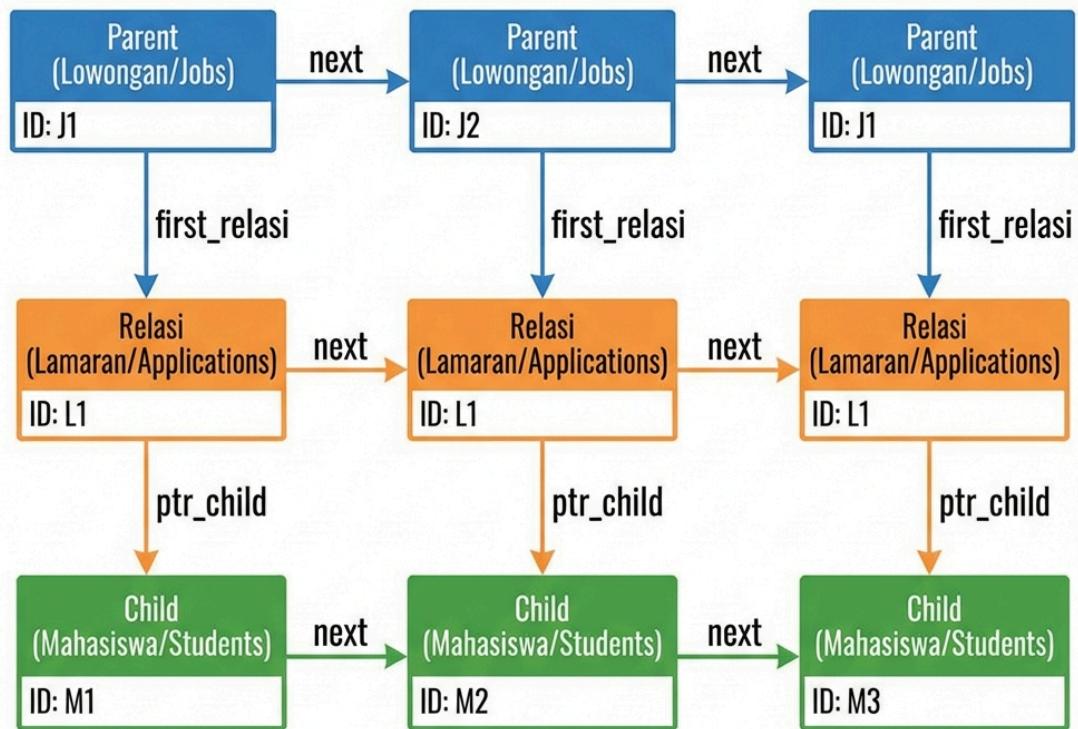
Select a job ID to view details, or type 'exit' to quit: ■

A.3 ATS Score Display

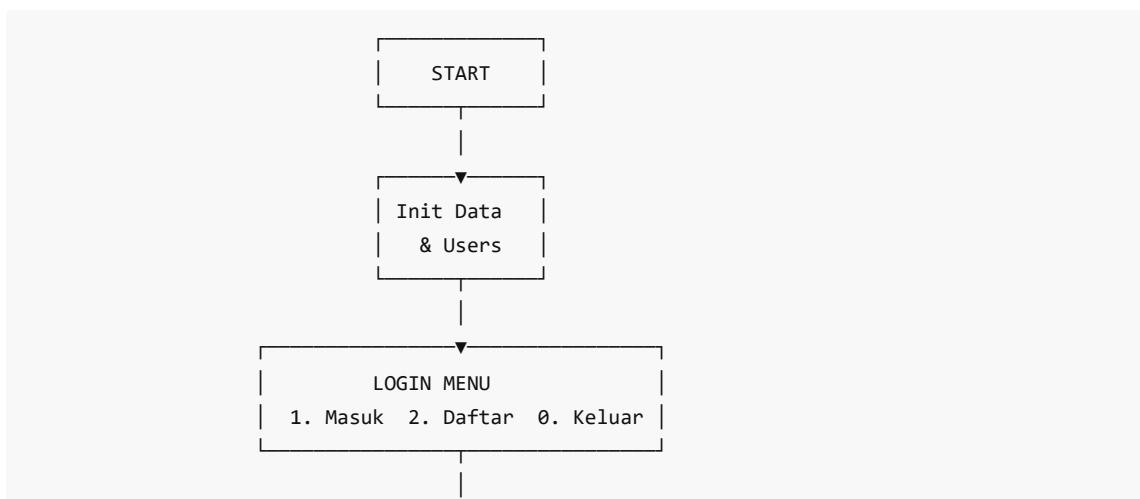


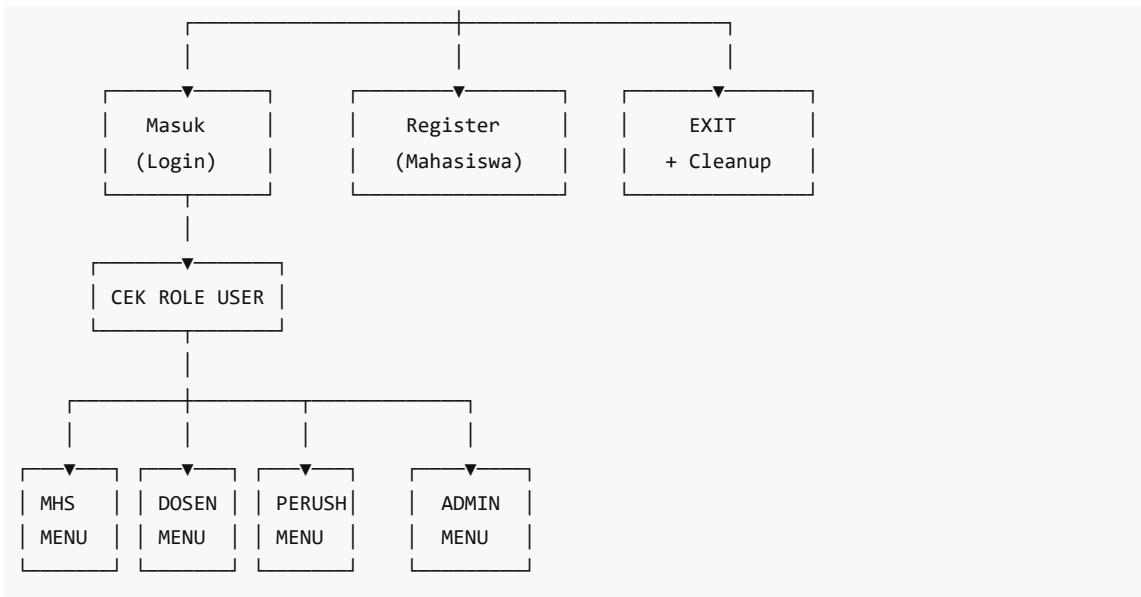
A.4 Diagram Struktur MLL Tipe B

Multi-Linked List Type B



B. Diagram Alur Program





C. Daftar Anggota Kelompok

No	NIM	Nama	Kontribusi
1	103032400010	Azriel Raihan Eldova Hartoto	Implementasi MLL & Lamaran
2	103032400035	Fathurrahman Alfarizi	Implementasi Login & Admin
3	103032400104	Muhamad Alwan Suryadi	Implementasi Lowongan & Mahasiswa

Dibuat untuk Tugas Besar Struktur Data

Telkom University - Desember 2024