

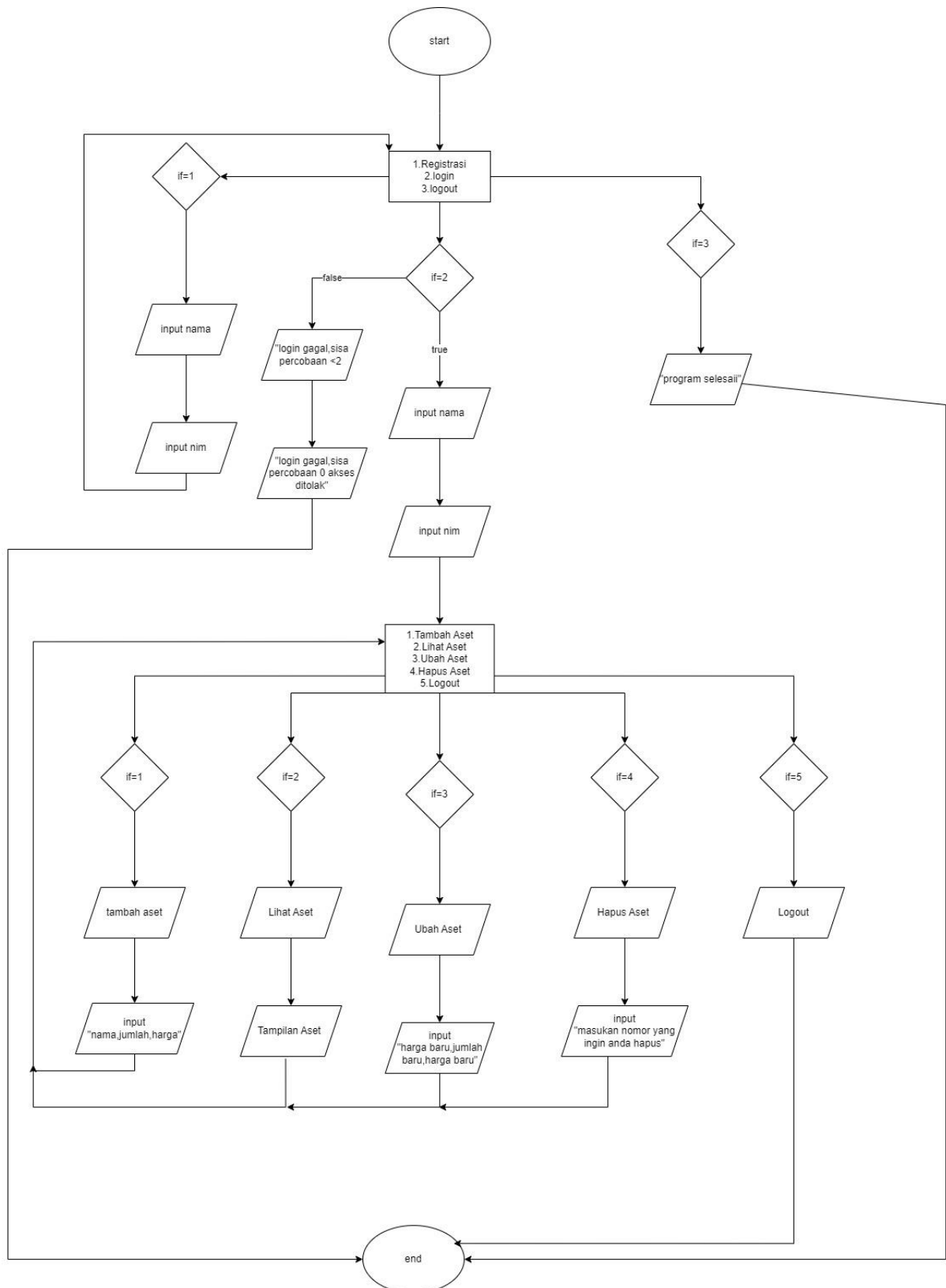
LAPORAN PRAKTIKUM
POSTTEST 3
ALGORITMA PEMROGRAMAN LANJUT



Disusun oleh:
Gilbert Finsen Zakaria Lian (2409106110)
Kelas (C'2)

PROGRAM STUDI INFORMATIKA
UNIVERSITAS MULAWARMAN
SAMARINDA
2025

1. Flowchart



2. Analisis Program

(Tuliskan secara ringkas tujuan dan fungsi/manfaat utama dari program yang dibuat.)

Tujuan Program

Program ini bertujuan untuk mengelola data aset dengan fitur multiuser menggunakan konsep array of struct dan nested struct. Setiap user dapat memiliki daftar aset pribadi yang dapat dikelola secara mandiri.

Fungsi dan Manfaat Utama

1. **Registrasi dan Login:** Memungkinkan pengguna untuk membuat akun baru dan masuk dengan validasi hingga 3 kali percobaan.
2. **Manajemen Aset:** Pengguna dapat menambah, melihat, mengubah, dan menghapus data aset yang mereka miliki.
3. **Penggunaan Struct:** Mengorganisir data aset dan pengguna secara efisien menggunakan nested struct.
4. **Data Terpisah per User:** Setiap user memiliki data aset yang tersimpan secara terpisah, menjaga keamanan dan privasi data.
5. **Antarmuka Sederhana:** Menyediakan menu navigasi yang mudah dipahami untuk mengelola aset.

3. Source Code

3.1. Header File dan Pengaturan Dasar

Source Code:

```
#include <iostream>
#include <string>
#include <iomanip>
using namespace std;
```

3.2. Struct

```
struct Aset {
    string nama;
    int jumlah;
    double harga;
};

struct User {
    string nama;
    string nim;
    Aset daftarAset[100];
    int jumlahAsetTersimpan = 0;
};
```

3.3 Variabel Global

```
User users[100];
int totalUsers = 0;
```

3.4 Fungsi Registrasi

```
int registrasi() {
    if (totalUsers < 100) {
        cout << "Nama: ";
        getline(cin >> ws, users[totalUsers].nama);
        cout << "NIM: ";
        getline(cin, users[totalUsers].nim);
        cout << "Registrasi berhasil. Silakan login." << endl;
        totalUsers++;
        return 1;
    } else {
        cout << "User telah penuh." << endl;
        return 0;
    }
}
```

3.5 Fungsi Login

```
int login() {
    string nama, nim;
    int percobaan = 0;

    while (percobaan < 3) {
        cout << "Nama: ";
        getline(cin >> ws, nama);
        cout << "NIM: ";
        getline(cin, nim);

        for (int i = 0; i < totalUsers; i++) {
            if (users[i].nama == nama && users[i].nim == nim) {
                cout << "Login berhasil!" << endl;
                return i;
            }
        }
        percobaan++;
        cout << "Login gagal. Sisa percobaan: " << 3 - percobaan <<
endl;
    }
    cout << "Akses ditolak." << endl;
    return -1;
}
```

3.6 Fungsi Menu Aset

```
int menuAset(User &user)
```

3.7 Tambah Aset

```
case 1: {
    if (user.jumlahAsetTersimpan < 100) {
        cout << "Nama Aset: ";
        getline(cin, user.daftarAset[user.jumlahAsetTersimpan].nama);
        cout << "Jumlah: ";
        cin >> user.daftarAset[user.jumlahAsetTersimpan].jumlah;
        cout << "Harga: ";
        cin >> user.daftarAset[user.jumlahAsetTersimpan].harga;
        user.jumlahAsetTersimpan++;
        cout << "Aset berhasil ditambahkan." << endl;
    } else {
        cout << "Daftar aset penuh." << endl;
    }
    break;
}
```

3.8 Lihat Daftar Aset

```
case 2: {
    if (user.jumlahAsetTersimpan > 0) {
        cout << "\nDaftar Aset:\n";
        cout << setw(5) << "No" << setw(20) << "Nama Aset" << setw(10)
        << "Jumlah" << setw(15) << "Harga" << endl;
        cout << "-----"
        << endl;
        for (int i = 0; i < user.jumlahAsetTersimpan; i++) {
            cout << setw(5) << i + 1 << setw(20) <<
            user.daftarAset[i].nama << setw(10) << user.daftarAset[i].jumlah <<
            setw(15) << user.daftarAset[i].harga << endl;
        }
    } else {
        cout << "Daftar aset kosong." << endl;
    }
    break;
}
```

3.9 Ubah Aset

```
case 3: {
    int nomor;
    cout << "Masukkan nomor aset yang ingin diubah: ";
    cin >> nomor;
    if (nomor > 0 && nomor <= user.jumlahAsetTersimpan) {
        nomor--;
        cout << "Nama Aset Baru: ";
        getline(cin >> ws, user.daftarAset[nomor].nama);
        cout << "Jumlah Baru: ";
        cin >> user.daftarAset[nomor].jumlah;
        cout << "Harga Baru: ";
        cin >> user.daftarAset[nomor].harga;
        cout << "Aset berhasil diubah." << endl;
    } else {
        cout << "Nomor aset tidak valid." << endl;
    }
    break;
}
```


3.10 Hapus Aset

```
case 4: {
    int nomor;
    cout << "Masukkan nomor aset yang ingin dihapus: ";
    cin >> nomor;
    if (nomor > 0 && nomor <= user.jumlahAsetTersimpan) {
        nomor--;
        for (int i = nomor; i < user.jumlahAsetTersimpan - 1; i++) {
            user.daftarAset[i] = user.daftarAset[i + 1];
        }
        user.jumlahAsetTersimpan--;
        cout << "Aset berhasil dihapus." << endl;
    } else {
        cout << "Nomor aset tidak valid." << endl;
    }
    break;
}
```

3.11 Fungsi Main

```
int main() {
    int pilihan;
    do {
        cout << "\n1. Registrasi\n2. Login\n3. Keluar\nPilihan: ";
        cin >> pilihan;
        cin.ignore();

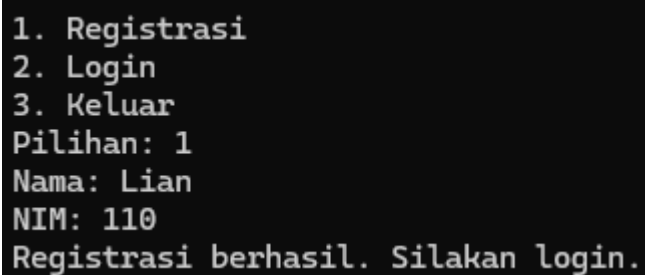
        if (pilihan == 1) registrasi();
        else if (pilihan == 2) {
            int userIndex = login();
            if (userIndex != -1) menuAset(users[userIndex]);
        }
    } while (pilihan != 3);

    cout << "Program selesai." << endl;
    return 0;
}
```

4. Uji Coba dan Hasil Output

(Sertakan tangkapan layar atau hasil output dari program setelah dijalankan.)

4.1 Halaman Untuk Registrasi

A screenshot of a terminal window with a black background and white text. It displays a menu with three options: 1. Registrasi, 2. Login, and 3. Keluar. Below the menu, it shows the user's selection: Pilihan: 1. Then it prompts for a name: Nama: Lian, and a NIM: 110. The final output is: Registrasi berhasil. Silakan login.

```
1. Registrasi
2. Login
3. Keluar
Pilihan: 1
Nama: Lian
NIM: 110
Registrasi berhasil. Silakan login.
```

Gambar 4.1

4.2 Halaman Login

```
1. Registrasi
2. Login
3. Keluar
Pilihan: 2
Nama: Lian
NIM: 110
Login berhasil!
```

Gambar 4.2

4.3 Pilihan Menu

```
Menu:
1. Tambah Aset
2. Lihat Daftar Aset
3. Ubah Aset
4. Hapus Aset
5. Logout
Pilihan Anda: |
```

Gambar 4.3

4.4 Tambah Aset

```
Menu:
1. Tambah Aset
2. Lihat Daftar Aset
3. Ubah Aset
4. Hapus Aset
5. Logout
Pilihan Anda: 1
Nama Aset: meja
Jumlah: 1
Harga: 100
Aset berhasil ditambahkan.
```

```
Menu:
1. Tambah Aset
2. Lihat Daftar Aset
3. Ubah Aset
4. Hapus Aset
5. Logout
Pilihan Anda: 1
Nama Aset: kursi
Jumlah: 4
Harga: 350
Aset berhasil ditambahkan.
```

```
Menu:
1. Tambah Aset
2. Lihat Daftar Aset
3. Ubah Aset
4. Hapus Aset
5. Logout
Pilihan Anda: 1
Nama Aset: TV
Jumlah: 1
Harga: 500
Aset berhasil ditambahkan.
```

Gambar 4.4

4.5 Lihat Aset

Menu:

1. Tambah Aset
2. Lihat Daftar Aset
3. Ubah Aset
4. Hapus Aset
5. Logout

Pilihan Anda: 2

Daftar Aset:

No	Nama Aset	Jumlah	Harga
1	meja	1	100
2	kursi	4	350
3	TV	1	500

Gambar 4.5

4.6 Ubah Aset

```
Menu:
1. Tambah Aset
2. Lihat Daftar Aset
3. Ubah Aset
4. Hapus Aset
5. Logout
Pilihan Anda: 3
Masukkan nomor aset yang ingin diubah: 1
Nama Aset Baru: mobil
Jumlah Baru: 1
Harga Baru: 9000
Aset berhasil diubah.
```

```
Menu:
1. Tambah Aset
2. Lihat Daftar Aset
3. Ubah Aset
4. Hapus Aset
5. Logout
Pilihan Anda: 2
```

Daftar Aset:

No	Nama Aset	Jumlah	Harga
1	mobil	1	9000
2	kursi	4	350
3	TV	1	500

Gambar 4.6

4.7 Hapus Aset

```
Menu:
1. Tambah Aset
2. Lihat Daftar Aset
3. Ubah Aset
4. Hapus Aset
5. Logout
Pilihan Anda: 4
Masukkan nomor aset yang ingin dihapus: 3
Aset berhasil dihapus.
```

```
Menu:
1. Tambah Aset
2. Lihat Daftar Aset
3. Ubah Aset
4. Hapus Aset
5. Logout
Pilihan Anda: 2
```

Daftar Aset:

No	Nama Aset	Jumlah	Harga
1	mobil	1	9000
2	kursi	4	350

Gambar 4.7

4.8 Logout dan keluar program

```
Menu:
1. Tambah Aset
2. Lihat Daftar Aset
3. Ubah Aset
4. Hapus Aset
5. Logout
Pilihan Anda: 5
Logout berhasil.
```

```
1. Registrasi
2. Login
3. Keluar
Pilihan: 3
Program selesai.
```

Gambar 4.8

4.9 Akses Di Tolak Saat Gagal Login

```
1. Registrasi
2. Login
3. Keluar
Pilihan: 2
Nama: li
NIM: 123
Login gagal. Sisa percobaan: 2
Nama: li
NIM: 123
Login gagal. Sisa percobaan: 1
Nama: li
NIM: 123
Login gagal. Sisa percobaan: 0
Akses ditolak.
```

Gambar 4.9

Gambar 4.1 Contoh

Gambar <nomor bab>.[index] <judul-gambar>

5. Langkah-Langkah Git

Git add

```
ASUS@LAPTOP-Lian MINGW64 /d/Praktikum-apl/Post-test-3 (main)
$ git add .
```

Git commit -m

```
ASUS@LAPTOP-Lian MINGW64 /d/Praktikum-apl/Post-test-3 (main)
$ git commit -m "post-test-3"
[main cbf16c4] post-test-3
4 files changed, 166 insertions(+)
create mode 100644 Post-test-3/2409106110-Gilbert Finsen Zakaria Lian-PT-3.cpp
create mode 100644 Post-test-3/2409106110-Gilbert Finsen Zakaria Lian-PT-3.exe
create mode 100644 Post-test-3/Dokumen tanpa judul.docx
create mode 100644 Post-test-3/~$kumen tanpa judul.docx
```


Git push

```
ASUS@LAPTOP-Lian MINGW64 /d/Praktikum-apl/Post-test-3 (main)
$ git push
Enumerating objects: 8, done.
Counting objects: 100% (8/8), done.
Delta compression using up to 2 threads
Compressing objects: 100% (7/7), done.
Writing objects: 100% (7/7), 606.48 KiB | 4.46 MiB/s, done.
Total 7 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/g1brtlia/Praktikum-apl.git
   f24ee7f..cbf16c4  main -> main
```