

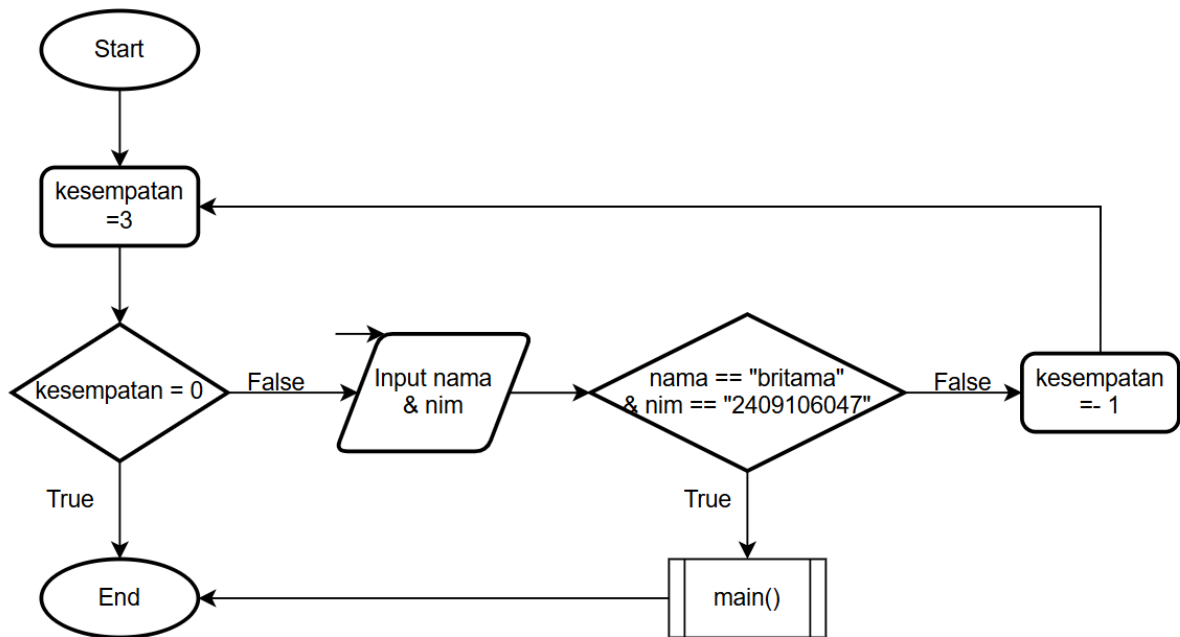
**LAPORAN PRAKTIKUM**  
**POSTTEST 3**  
**ALGORITMA PEMROGRAMAN LANJUT**



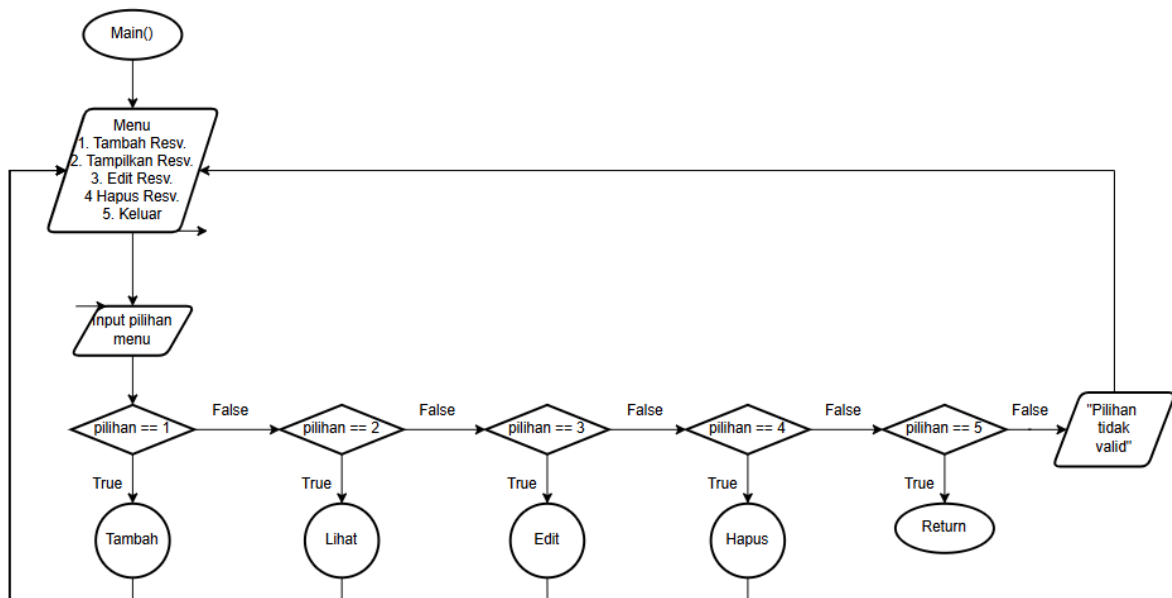
**Disusun oleh:**  
**Muhammad Britama Putra Jaya (2409106047)**  
**Kelas (B1 '24)**

**PROGRAM STUDI INFORMATIKA**  
**UNIVERSITAS MULAWARMAN**  
**SAMARINDA**  
**2025**

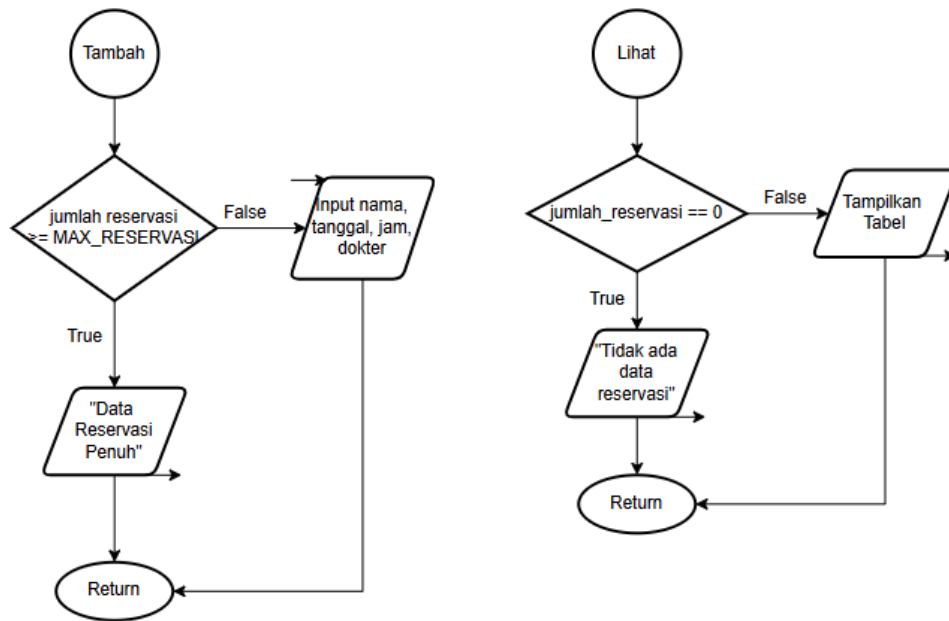
## 1. Flowchart



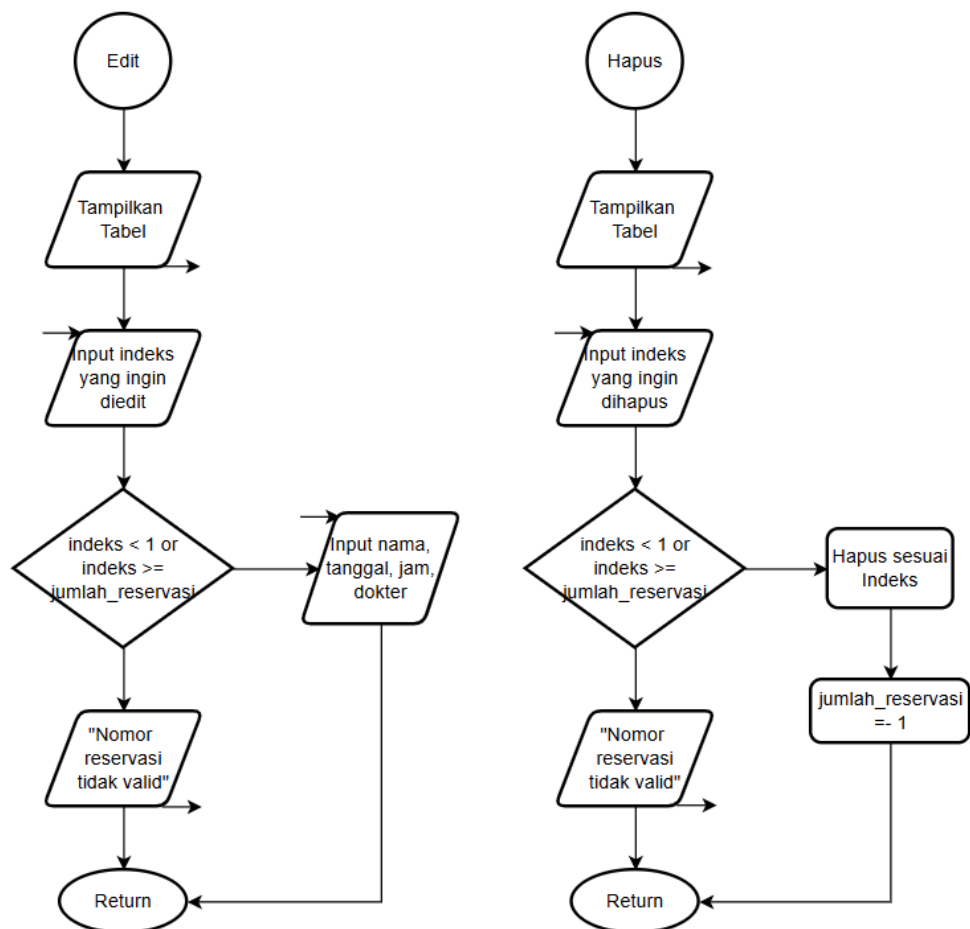
Gambar 1.1 Flowchart Login



Gambar 1.2 Flowchart Menu



Gambar 1.3 Flowchart Tambah dan Lihat Reservasi



Gambar 1.4 Flowchart Edit dan Hapus Reservasi

## 2. Analisis Program

### 2.1 Deskripsi Singkat Program

Pada posttest 3 kali ini, masih kurang lebih sama dengan posttest sebelumnya hanya berbeda di posttest kali ini dimana diminta untuk melakukan CRUD pada array of struct. Disini juga saya menggunakan nested struct. Menyesuaikan dengan judul sistem “Manajemen Reservasi Klinik Gigi”. Untuk fitur kurang lebih sama saja, hanya berbeda sedikit di source code karena menggunakan nested struct.

## 3. Source Code

### A. Struct untuk Data Reservasi (Nested Struct)

Fitur ini digunakan untuk menyimpan data. Dimana “Pasien” menyimpan nama pasien; “Jadwal” menyimpan tanggal dan jam reservasi; “Dokter” menyimpan nama dokter. Untuk Nested Struct nya disini adalah reservasi, dimana menggunakan struct pasien, jadwal, dan dokter.

#### Source Code:

```
struct Pasien {  
    string nama;  
};  
  
struct Jadwal {  
    string tanggal;  
    string jam;  
};  
  
struct Dokter {  
    string nama;  
};  
  
struct Reservasi {  
    Pasien pasien;  
    Jadwal jadwal;  
    Dokter dokter;  
};
```

## B. Tambah Reservasi

Fitur ini digunakan untuk menambah reservasi klinik gigi, dimana program akan mengecek apakah reservasi sudah penuh, meminta user untuk menginput nama pasien, tanggal, jam, dan dokter, serta menambah “jumlah\_reservasi” setelah data berhasil disimpan.

### Source Code:

```
if (pilihan == 1) {
    if (jumlah_reservasi >= MAX_RESERVASI) {
        cout << "Data reservasi penuh!" << endl;
    } else {
        cout << "\nMasukkan Nama Pasien : ";
        getline(cin, reservasi[jumlah_reservasi].pasien.nama);
        cout << "Masukkan Tanggal (DD-MM-YYYY) : ";
        getline(cin, reservasi[jumlah_reservasi].jadwal.tanggal);
        cout << "Masukkan Jam (HH:MM) : ";
        getline(cin, reservasi[jumlah_reservasi].jadwal.jam);
        cout << "Masukkan Nama Dokter : ";
        string nama_dokter;
        getline(cin, nama_dokter);
        reservasi[jumlah_reservasi].dokter.nama = "Drg. " + nama_dokter;
        jumlah_reservasi++;
        cout << "Reservasi berhasil ditambahkan!" << endl;
    }
}
```

### C. Menampilkan Reservasi

Fitur ini digunakan untuk menampilkan data reservasi secara rapi ke dalam bentuk tabel menggunakan `setw()`. Sebelum itu program akan mengecek apakah ada reservasi yang tersedia.

#### Source Code:

```
else if (pilihan == 2 || pilihan == 3 || pilihan == 4) {
    if (jumlah_reservasi == 0) {
        cout << "Tidak ada data reservasi!" << endl;
    } else {
        cout <<
        "\n===== " <<
        endl;
        cout << "| No | Nama Pasien | Tanggal | Jam |" << endl;
        cout <<
        "===== " << endl;
        for (int i = 0; i < jumlah_reservasi; i++) {
            cout << "| " << setw(3) << i+1 << " | "
            << setw(20) << left << reservasi[i].pasien.nama << " | "
            << setw(10) << reservasi[i].jadwal.tanggal << " | "
            << setw(5) << reservasi[i].jadwal.jam << " | "
            << setw(15) << left << reservasi[i].dokter.nama << " | "
            << endl;
        }
        cout <<
        "===== " << endl;
    }
}
```

## D. Mengedit Reservasi

Fitur ini digunakan mengedit reservasi yang ada sesuai dengan nomor reservasi yang ingin diedit.

### Source Code:

```
if (pilihan == 3) {
    int indeks;
    cout << "Masukkan nomor reservasi yang ingin diedit : ";
    cin >> indeks;
    cin.ignore();
    if (indeks < 1 || indeks > jumlah_reservasi) {
        cout << "Nomor reservasi tidak valid!" << endl;
    } else {
        cout << "\nMasukkan Nama Pasien : ";
        getline(cin, reservasi[indeks-1].pasien.nama);
        cout << "Masukkan Tanggal (DD-MM-YYYY) : ";
        getline(cin, reservasi[indeks-1].jadwal.tanggal);
        cout << "Masukkan Jam (HH:MM) : ";
        getline(cin, reservasi[indeks-1].jadwal.jam);
        cout << "Masukkan Nama Dokter : ";
        string nama_dokter;
        getline(cin, nama_dokter);
        reservasi[indeks-1].dokter.nama = "Drg. " + nama_dokter;
        cout << "Reservasi berhasil diperbarui!" << endl;
    }
}
```

## E. Menghapus Reservasi

Fitur ini digunakan untuk menghapus reservasi yang ada sesuai dengan nomor reservasi yang ingin dihapus.

### Source Code:

```
if (pilihan == 4) {
    int indeks;
    cout << "Masukkan nomor reservasi yang ingin dihapus : ";
    cin >> indeks;
    cin.ignore();
    if (indeks < 1 || indeks > jumlah_reservasi) {
        cout << "Nomor reservasi tidak valid!" << endl;
    } else {
        for (int i = indeks - 1; i < jumlah_reservasi - 1; i++) {
            reservasi[i] = reservasi[i + 1];
        }
        jumlah_reservasi--;
        cout << "Reservasi berhasil dihapus!" << endl;
    }
}
```



## 4. Uji Coba dan Hasil Output

### 4.1 Hasil Output

```
=====
                LOGIN SISTEM KLINIK
=====
Masukkan Nama : britama
Masukkan NIM : 2409106047

=====
            MENU MANAJEMEN RESERVASI KLINIK
=====
1. Tambah Reservasi
2. Tampilkan Reservasi
3. Edit Reservasi
4. Hapus Reservasi
5. Keluar
=====
Pilih menu : |
```

Gambar 4.1 Login Berhasil dan Menu Utama

```
=====
            MENU MANAJEMEN RESERVASI KLINIK
=====
1. Tambah Reservasi
2. Tampilkan Reservasi
3. Edit Reservasi
4. Hapus Reservasi
5. Keluar
=====
Pilih menu : 1

Masukkan Nama Pasien : Jarwo
Masukkan Tanggal (DD-MM-YYYY) : 14-09-2025
Masukkan Jam (HH:MM) : 14:00
Masukkan Nama Dokter : Harun
Reservasi berhasil ditambahkan!
```

Gambar 4.2 Tambah Reservasi

```
=====
MENU MANAJEMEN RESERVASI KLINIK
=====
1. Tambah Reservasi
2. Tampilkan Reservasi
3. Edit Reservasi
4. Hapus Reservasi
5. Keluar
=====
Pilih menu : 2

=====
| No | Nama Pasien | Tanggal | Jam | Dokter |
=====
| 1 | Jarwo | 14-09-2025 | 14:00 | Drg. Harun |
=====
```

Gambar 4.3 Tampilkan Reservasi

```

=====
MENU MANAJEMEN RESERVASI KLINIK
=====
1. Tambah Reservasi
2. Tampilkan Reservasi
3. Edit Reservasi
4. Hapus Reservasi
5. Keluar
=====
Pilih menu : 3

=====
| No | Nama Pasien | Tanggal | Jam | Dokter |
=====
| 1 | Jarwo | 14-09-2025 | 14:00 | Drg. Harun |
=====

Masukkan nomor reservasi yang ingin diedit : 1

Masukkan Nama Pasien : Udin
Masukkan Tanggal (DD-MM-YYYY) : 14-09-2025
Masukkan Jam (HH:MM) : 14:30
Masukkan Nama Dokter : Tarno
Reservasi berhasil diperbarui!

=====
MENU MANAJEMEN RESERVASI KLINIK
=====
1. Tambah Reservasi
2. Tampilkan Reservasi
3. Edit Reservasi
4. Hapus Reservasi
5. Keluar
=====
Pilih menu : 2

=====
| No | Nama Pasien | Tanggal | Jam | Dokter |
=====
| 1 | Udin | 14-09-2025 | 14:30 | Drg. Tarno |
=====

```

Gambar 4.4 Mengedit Reservasi

```
=====
MENU MANAJEMEN RESERVASI KLINIK
=====
1. Tambah Reservasi
2. Tampilkan Reservasi
3. Edit Reservasi
4. Hapus Reservasi
5. Keluar
=====
Pilih menu : 4

=====
| No | Nama Pasien | Tanggal | Jam | Dokter |
=====
| 1 | Udin | 14-09-2025 | 14:30 | Drg. Tarno |
=====
Masukkan nomor reservasi yang ingin dihapus : 1
Reservasi berhasil dihapus!

=====
MENU MANAJEMEN RESERVASI KLINIK
=====
1. Tambah Reservasi
2. Tampilkan Reservasi
3. Edit Reservasi
4. Hapus Reservasi
5. Keluar
=====
Pilih menu : 2
Tidak ada data reservasi!
```

Gambar 4.5 Menghapus Reservasi

```
=====
LOGIN SISTEM KLINIK
=====
Masukkan Nama : abc
Masukkan NIM : abc
Login gagal! Sisa kesempatan : 2

=====
LOGIN SISTEM KLINIK
=====
Masukkan Nama : abc
Masukkan NIM : abc
Login gagal! Sisa kesempatan : 1

=====
LOGIN SISTEM KLINIK
=====
Masukkan Nama : abc
Masukkan NIM : abc
Login gagal! Sisa kesempatan : 0
Anda diblokir! Silahkan hubungi admin.
```

Gambar 4.6 Gagal Login

## 5. Git

```
PS C:\Users\my asus\OneDrive\Desktop\praktikum-apl> git add .
warning: in the working copy of 'post-test/post-test-1/.vscode/tasks.json', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'post-test/post-test-1/2409106047_Muhammad Britama Putra Jaya_PT_1.drawio', LF will be replaced by CRLF the next time Git touches it
```

Gambar 5.1 Git Add : Menambah file ke git commit. Dengan menambahkan “.” secara otomatis semua perubahan file yang ada di direktori “praktikum-apl”.

```
PS C:\Users\my asus\OneDrive\Desktop\praktikum-apl> git commit -m "Finish Code Post Test 1"
[main (root-commit) 7310fea] Finish Code Post Test 1
5 files changed, 367 insertions(+)
create mode 100644 pertemuan-1/main.cpp
create mode 100644 post-test/post-test-1/.vscode/tasks.json
create mode 100644 post-test/post-test-1/2409106047_Muhammad Britama Putra Jaya_PT_1.cpp
create mode 100644 post-test/post-test-1/2409106047_Muhammad Britama Putra Jaya_PT_1.drawio
create mode 100644 post-test/post-test-1/2409106047_Muhammad Britama Putra Jaya_PT_1.exe
```

Gambar 5.2 Git Commit : digunakan untuk menyimpan perubahan dalam repository Git secara permanen.

```
PS C:\Users\my asus\OneDrive\Desktop\praktikum-apl> git push -u origin main
info: please complete authentication in your browser...
Enumerating objects: 11, done.
Counting objects: 100% (11/11), done.
Delta compression using up to 4 threads
Compressing objects: 100% (8/8), done.
Writing objects: 100% (11/11), 679.42 KiB | 3.47 MiB/s, done.
Total 11 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/Britamaxx/praktikum-apl.git
 * [new branch]      main -> main
branch 'main' set up to track 'origin/main'.
PS C:\Users\my asus\OneDrive\Desktop\praktikum-apl> █
```

Gambar 5.3 Git Push : digunakan untuk mengirim perubahan (commit) dari repository lokal ke repository jarak jauh, sehingga bisa diakses oleh orang lain atau disinkronkan dengan repository utama.