

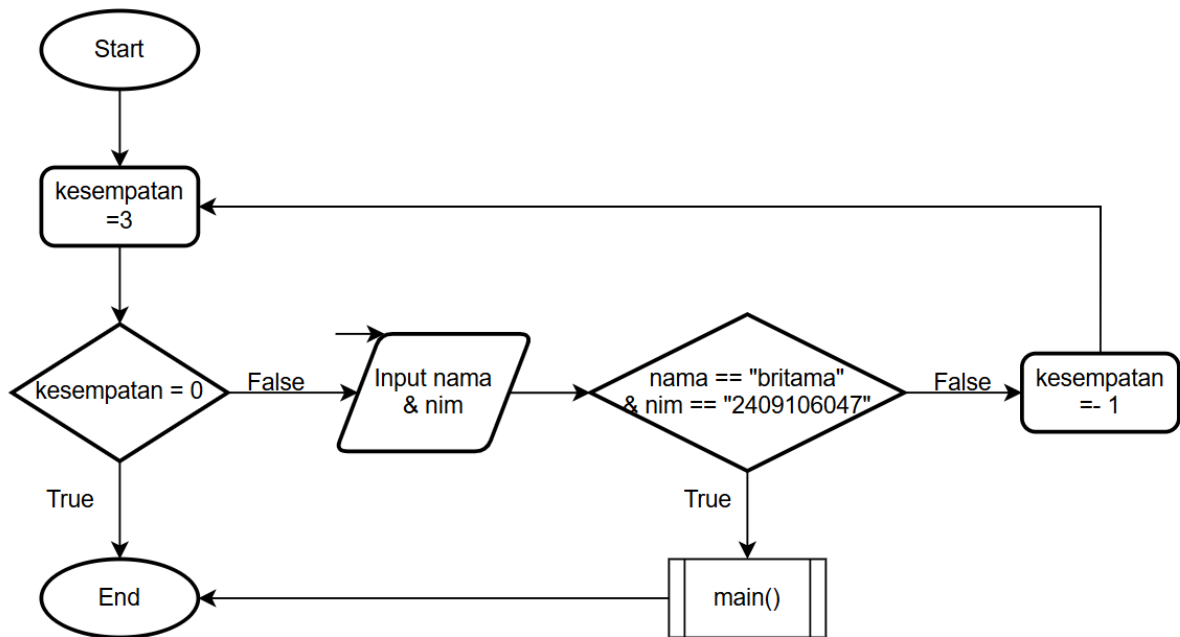
**LAPORAN PRAKTIKUM**  
**POSTTEST 4**  
**ALGORITMA PEMROGRAMAN LANJUT**



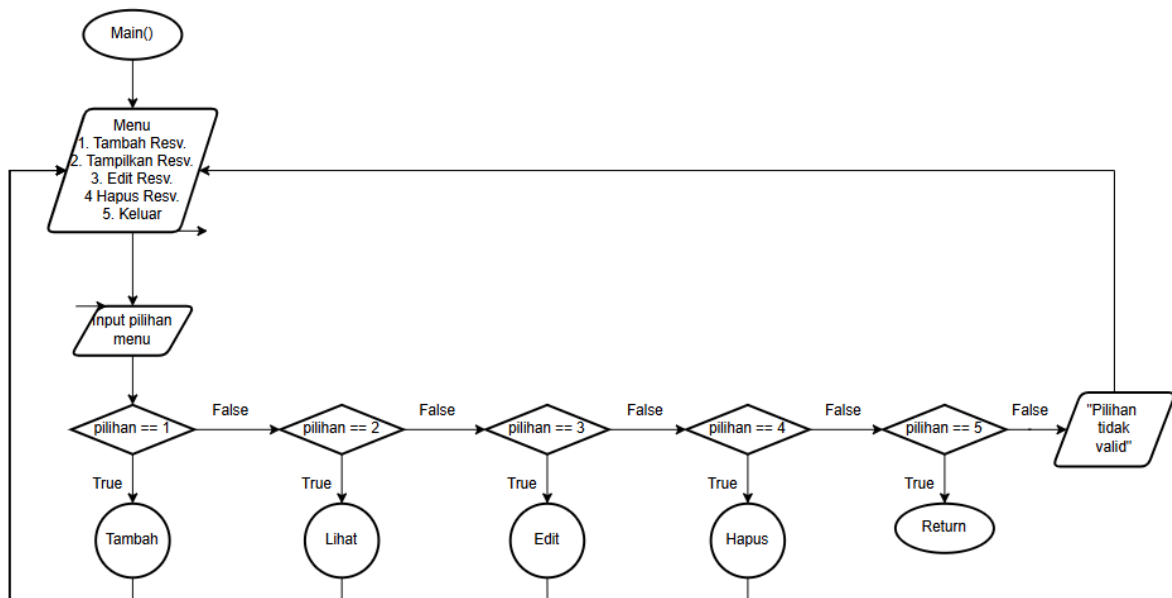
**Disusun oleh:**  
**Muhammad Britama Putra Jaya (2409106047)**  
**Kelas (B1 '24)**

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

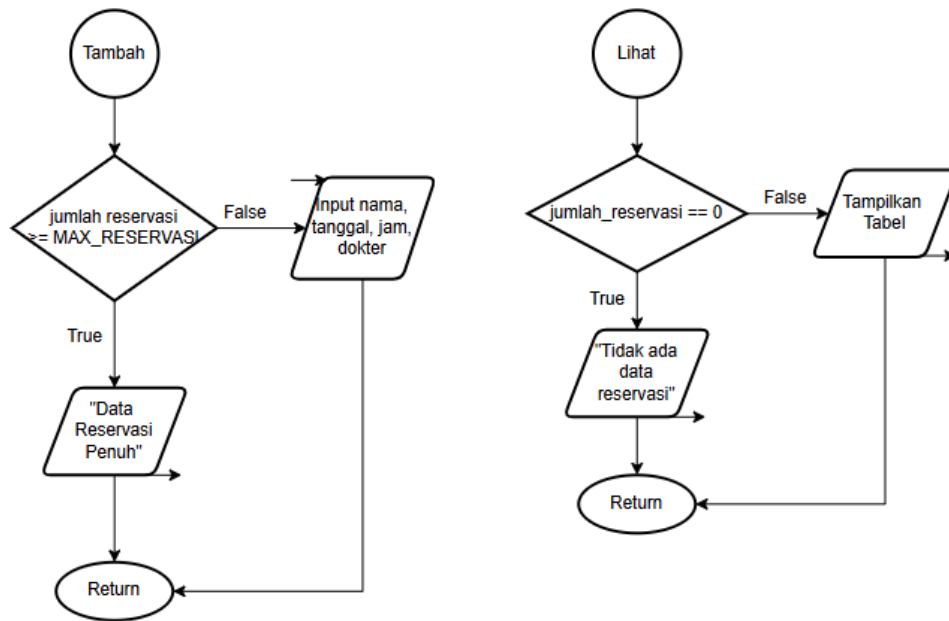
## 1. Flowchart



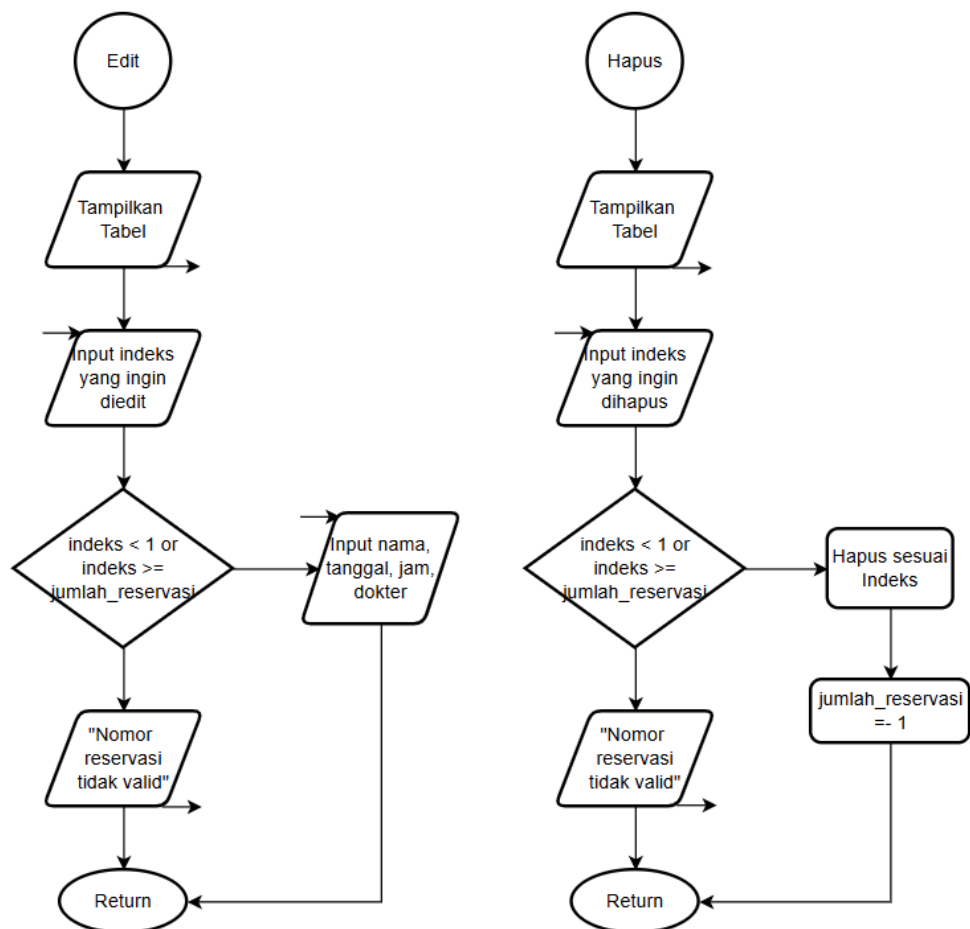
Gambar 1.1 Flowchart Login



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 4 kali ini, masih kurang lebih sama dengan posttest sebelumnya. Hanya saja pada posttest kali ini saya menambahkan dan mengedit beberapa hal, seperti mengubah pilihan menu menjadi sub menu/fungsi dengan parameternya, prosedur, fungsi rekursif dan juga fungsi overloading.

## 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:

```
void tambah_reservasi(Reservasi data[], int &jumlah) {
    if (jumlah >= MAX_RESERVASI) {
        cout << "Data reservasi penuh!" << endl;
        return;
    }

    cout << "\nMasukkan Nama Pasien : ";
    getline(cin, data[jumlah].pasien.nama);
    cout << "Masukkan Tanggal (DD-MM-YYYY) : ";
    getline(cin, data[jumlah].jadwal.tanggal);
    cout << "Masukkan Jam (HH:MM) : ";
    getline(cin, data[jumlah].jadwal.jam);
    cout << "Masukkan Nama Dokter : ";
    string nama_dokter;
    getline(cin, nama_dokter);
    data[jumlah].dokter.nama = "Drg. " + nama_dokter;
    jumlah++;
    cout << "Reservasi berhasil ditambahkan!" << endl;
}
```

### C. Menampilkan Reservasi

Fitur ini digunakan untuk menampilkan data reservasi.

**Source Code:**

```
void lihat_reservasi(Reservasi data[], int jumlah) {
    tampilkan_tabel();
    for (int i = 0; i < jumlah; i++) {
        cout << "| " << setw(3) << i + 1 << " | "
            << setw(20) << left << data[i].pasien.nama << " | "
            << setw(10) << data[i].jadwal.tanggal << " | "
            << setw(5) << data[i].jadwal.jam << " | "
            << setw(15) << left << data[i].dokter.nama << " |" << endl;
    }
    cout <<
    "===== " << endl;
    cout << "Total reservasi saat ini: " << hitung_reservasi(jumlah) << endl;
}
```

## D. Mengedit Reservasi

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

### Source Code:

```
void edit_reservasi(Reservasi data[], int jumlah) {
    lihat_reservasi(data, jumlah);

    int indeks;
    cout << "Masukkan nomor reservasi yang ingin diedit : ";
    cin >> indeks;
    cin.ignore();
    if (indeks < 1 || indeks > jumlah) {
        cout << "Nomor reservasi tidak valid!" << endl;
        return;
    }

    cout << "\n--- Data Lama ---" << endl;
    cout << "Nama Pasien : " << data[indeks - 1].pasien.nama << endl;
    cout << "Tanggal      : " << data[indeks - 1].jadwal.tanggal << endl;
    cout << "Jam          : " << data[indeks - 1].jadwal.jam << endl;
    cout << "Dokter       : " << data[indeks - 1].dokter.nama << endl;

    cout << "\n--- Masukkan Data Baru ---" << endl;
    cout << "Nama Pasien : ";
    getline(cin, data[indeks - 1].pasien.nama);
    cout << "Tanggal (DD-MM-YYYY) : ";
    getline(cin, data[indeks - 1].jadwal.tanggal);
    cout << "Jam (HH:MM) : ";
    getline(cin, data[indeks - 1].jadwal.jam);
    cout << "Nama Dokter : ";
    string nama_dokter;
    getline(cin, nama_dokter);
    data[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:

```
void hapus_reservasi(Reservasi data[], int &jumlah) {
    lihat_reservasi(data, jumlah);

    int indeks;
    cout << "Masukkan nomor reservasi yang ingin dihapus : ";
    cin >> indeks;
    cin.ignore();
    if (indeks < 1 || indeks > jumlah) {
        cout << "Nomor reservasi tidak valid!" << endl;
        return;
    }

    cout << "\nAnda akan menghapus reservasi berikut:" << endl;
    cout << "Nama Pasien : " << data[indeks - 1].pasien.nama << endl;
    cout << "Tanggal      : " << data[indeks - 1].jadwal.tanggal << endl;
    cout << "Jam          : " << data[indeks - 1].jadwal.jam << endl;
    cout << "Dokter       : " << data[indeks - 1].dokter.nama << endl;

    char konfirmasi;
    cout << "Yakin ingin menghapus? (y/n): ";
    cin >> konfirmasi;
    cin.ignore();

    if (konfirmasi == 'y' || konfirmasi == 'Y') {
        for (int i = indeks - 1; i < jumlah - 1; i++) {
            data[i] = data[i + 1];
        }
        jumlah--;
        cout << "Reservasi berhasil dihapus!" << endl;
    } else {
        cout << "Penghapusan dibatalkan." << endl;
    }
}
```



## F. Fungsi Rekursif

Fungsi ini digunakan untuk menghitung total jumlah reservasi secara rekursif.

**Source Code:**

```
int hitung_reservasi(int index) {  
    if (index == 0)  
        return 0;  
    return 1 + hitung_reservasi(index - 1);  
}
```

## G. Fungsi Overloading

Fungsi ini digunakan untuk menampilkan tabel dan header.

**Source Code:**

```
void tampilkan_tabel() {  
    cout <<  
    "\n===== " <<  
    endl;  
    cout << "| No | Nama Pasien | Tanggal | Jam | Dokter  
|" << endl;  
    cout <<  
    "===== " << endl;  
}  
  
void tampilkan_tabel(string pesan) {  
    cout << "\n== " << pesan << " ==" << endl;  
    tampilkan_tabel();  
}
```

## 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 | Harun | 14-09-2025 | 14:00 | Drg. Jarwo |
=====
Total reservasi saat ini: 1
```

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 | Harun | 14-09-2025 | 14:00 | Drg. Jarwo |
=====
Total reservasi saat ini: 1
Masukkan nomor reservasi yang ingin diedit : 1

--- Data Lama ---
Nama Pasien : Harun
Tanggal : 14-09-2025
Jam : 14:00
Dokter : Drg. Jarwo

--- Masukkan Data Baru ---
Nama Pasien : Jawir
Tanggal (DD-MM-YYYY) : 14-09-2025
Jam (HH:MM) : 14:30
Nama Dokter : Harun
Reservasi berhasil diperbarui!

```

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-4/.vscode/tasks.json', 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 4"  
[main dec9241] Finish Code Post Test 4  
3 files changed, 252 insertions(+)  
create mode 100644 post-test/post-test-4/.vscode/tasks.json  
create mode 100644 post-test/post-test-4/2409106047_Muhammad Britama Putra Jaya_PT_4.cpp  
create mode 100644 post-test/post-test-4/2409106047_Muhammad Britama Putra Jaya_PT_4.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  
Enumerating objects: 10, done.  
Counting objects: 100% (10/10), done.  
Delta compression using up to 4 threads  
Compressing objects: 100% (7/7), done.  
Writing objects: 100% (8/8), 700.27 KiB | 5.88 MiB/s, done.  
Total 8 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)  
remote: Resolving deltas: 100% (1/1), completed with 1 local object.  
To https://github.com/Britamaxx/praktikum-apl.git  
c12e169..dec9241 main -> main  
branch 'main' set up to track 'origin/main'.
```

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.