

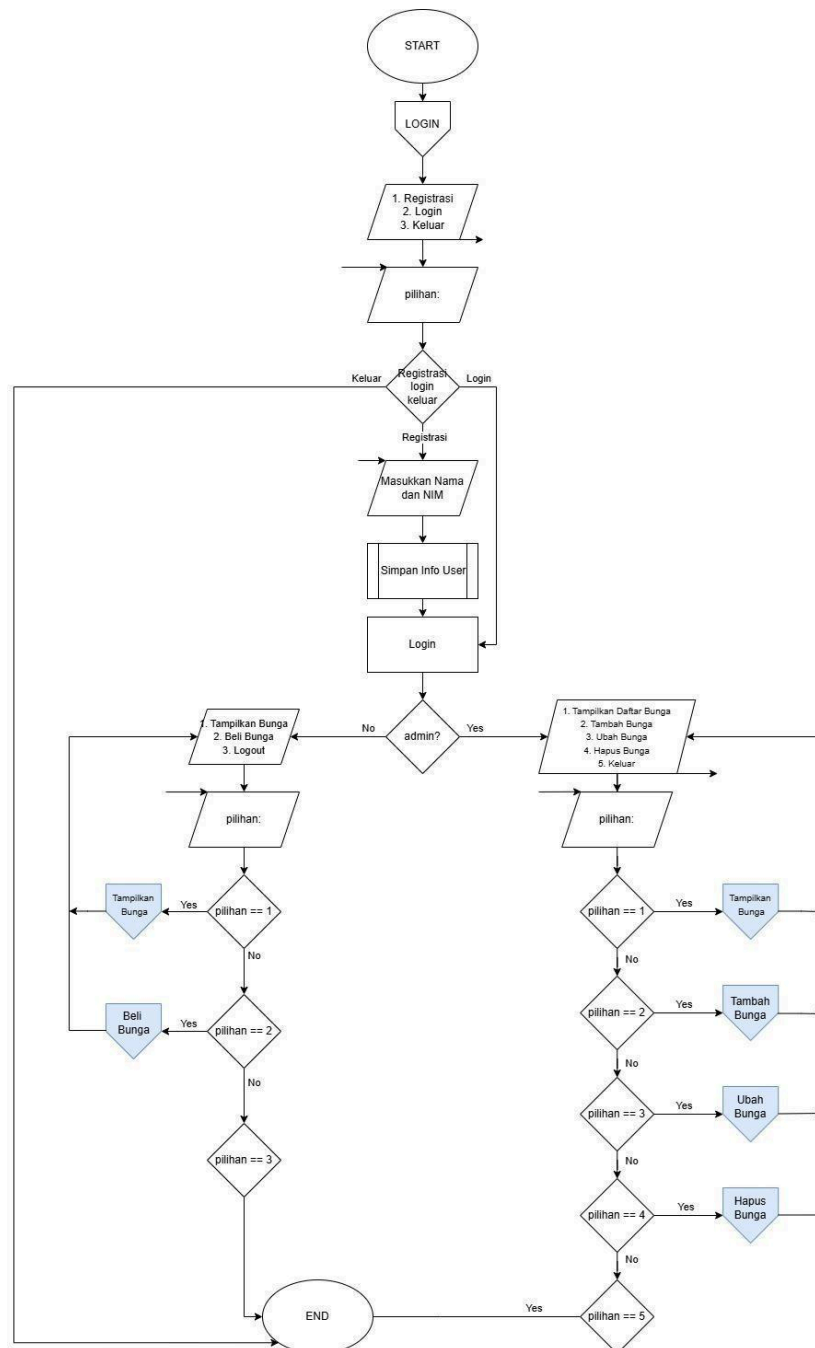
LAPORAN PRAKTIKUM
POSTTEST (5)
ALGORITMA PEMROGRAMAN LANJUT



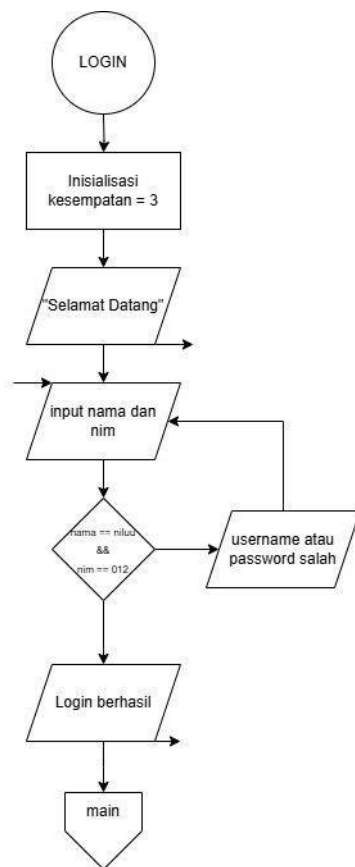
Disusun oleh:
Niluh Fincy Gloria Nathasia M (2409106012)
Kelas (A1 '24)

PROGRAM STUDI INFORMATIKA
UNIVERSITAS MULAWARMAN
SAMARINDA
2025

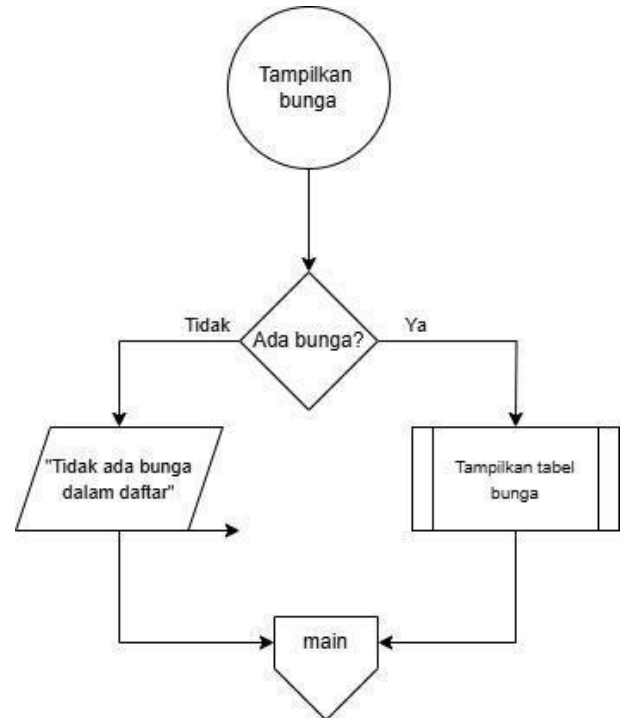
1. Flowchart



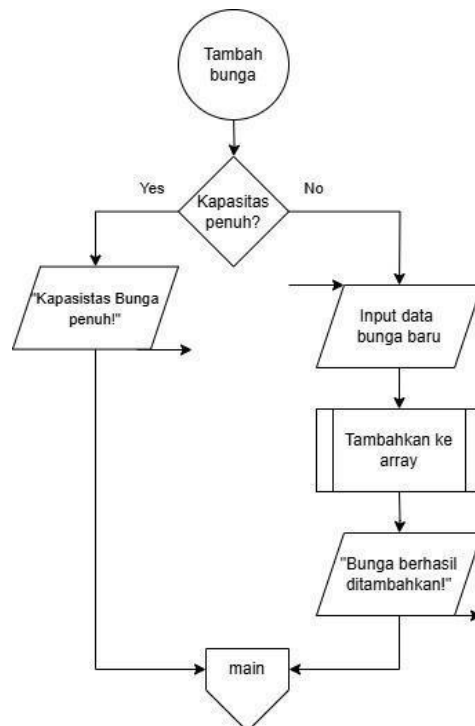
Gambar 1.1 Main Flowchart



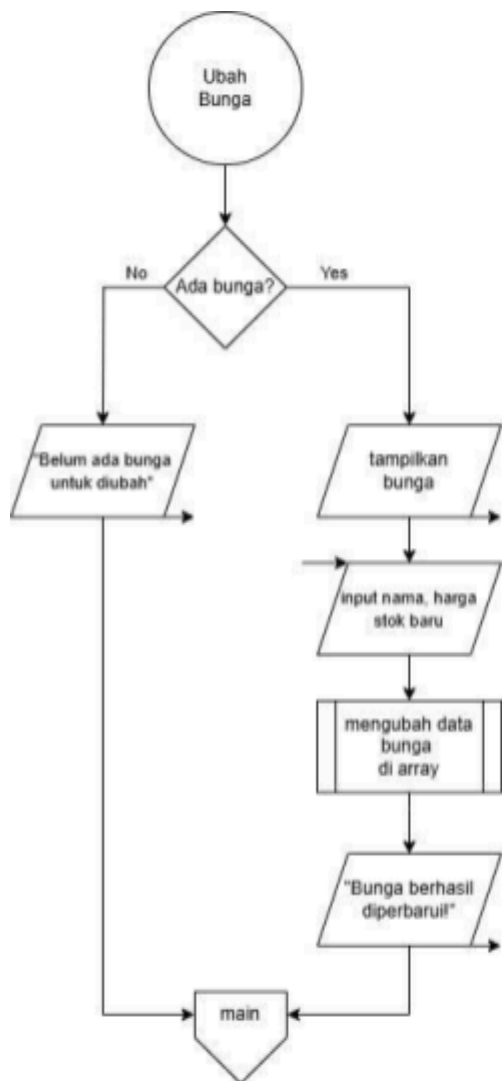
Gambar 1.2 Fungsi Login



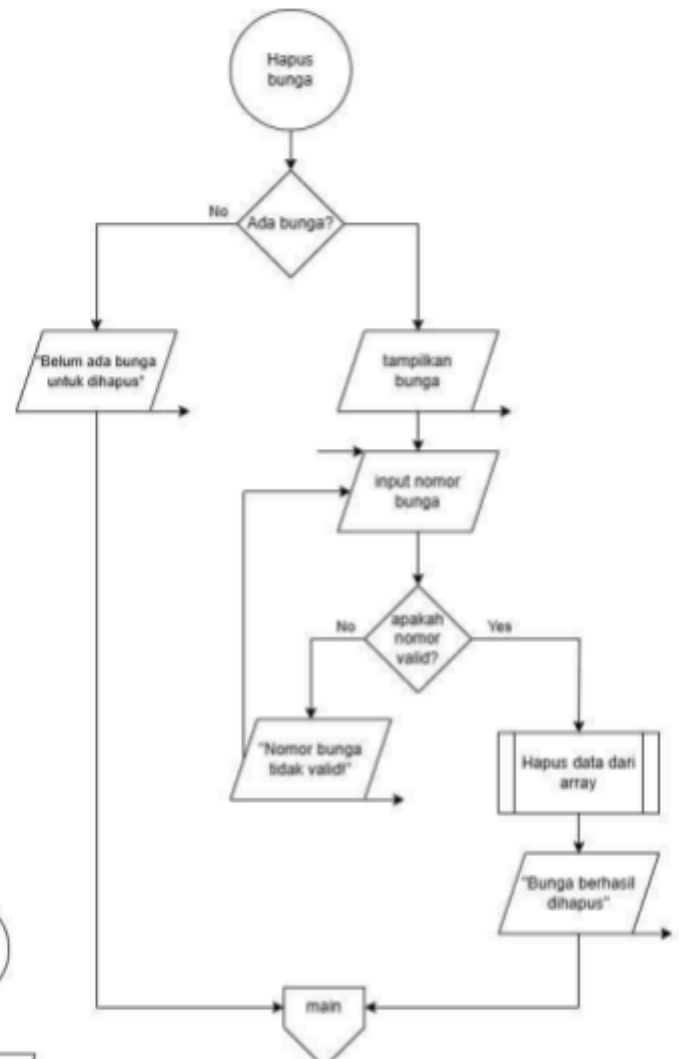
Gambar 1.3 Tampilkan Bunga



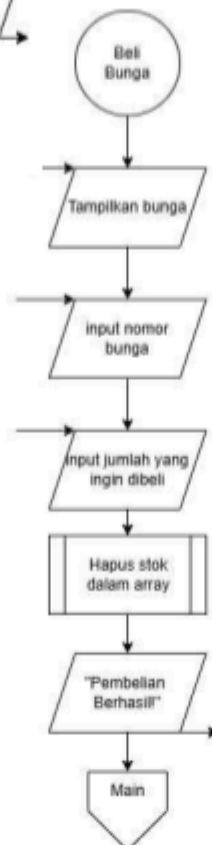
Gambar 1.4 Tambah Bunga



Gambar 1.5 Ubah Bunga



Gambar 1.6 Hapus Bunga



Gambar 1.7 Beli Bunga

2. Analisis Program

Program ini bertujuan untuk mengelola sistem penjualan bunga dengan fitur CRUD (Create, Read, Update, Delete) yang mencakup manajemen pengguna, admin, serta transaksi pembelian bunga.

Fungsi/Manfaat Utama:

1. Registrasi & Login: Pengguna dapat mendaftar dan login dengan autentikasi berbasis nama dan NIM.
2. Manajemen Bunga: Admin dapat menambah, mengedit, dan menghapus data bunga.
3. Transaksi Pembelian: Pengguna dapat melihat daftar bunga, membeli bunga, dan stok bunga diperbarui otomatis.
4. Keamanan & Kontrol: Login terbatas dengan maksimal 3 kali percobaan, serta perbedaan akses antara user dan admin.
5. Tampilan Data Rapi: Menggunakan format tabel dengan `setw()` untuk meningkatkan keterbacaan data.

3. Source Code

A. Fitur Tambah Data

Fitur ini digunakan untuk menambahkan data baru. Data yang ditambahkan berupa nama, harga, dan stok bunga.

Source Code:

```
void tambahBunga(Bunga daftar[], int &jumlah) {
    if (jumlah < 10) {
        cout << "Masukkan Nama Bunga: ";
        getline(cin, daftar[jumlah].nama);
        cout << "Masukkan Harga: ";
        cin >> daftar[jumlah].harga;
        cout << "Masukkan Stok: ";
        cin >> daftar[jumlah].stok;
        cin.ignore();
        jumlah++;
        cout << "Bunga Berhasil Ditambahkan!\n";
    } else {
        cout << "Daftar bunga penuh!\n";
    }
}
```

B. Fitur Lihat Data

Fitur ini digunakan untuk melihat data yang telah ditambahkan.

Source Code:

```
// Fungsi overloading
void tampilkanBunga(Bunga bunga) {
    cout << left << setw(20) << bunga.nama
         << setw(15) << bunga.harga
         << setw(10) << bunga.stok << endl;
}

void tampilkanBunga(Bunga daftar[], int jumlah) {
    cout << left << setw(5) << "No"
         << setw(20) << "Nama Bunga"
         << setw(15) << "Harga"
         << setw(10) << "Stok" << endl;
    cout << string(50, '-') << endl;
    for (int i = 0; i < jumlah; i++) {
        cout << left << setw(5) << i + 1;
        tampilkanBunga(daftar[i]);
    }
}

// Fungsi dengan parameter dereference (int*)
void tampilkanJumlahBunga(int *jumlah) {
```

```
cout << "Jumlah jenis bunga saat ini: " << *jumlah << " jenis.\n";  
}
```

C. Fitur Hapus Data

Fitur ini digunakan untuk menghapus data bunga.

Source Code:

```
void hapusBunga(Bunga daftar[], int &jumlah) {
    int hapus;
    cout << "Masukkan nomor bunga yang akan dihapus: ";
    cin >> hapus;
    if (hapus >= 1 && hapus <= jumlah) {
        for (int i = hapus - 1; i < jumlah - 1; i++) {
            daftar[i] = daftar[i + 1];
        }
        jumlah--;
        cout << "Bunga berhasil dihapus!\n";
    } else {
        cout << "Nomor tidak valid!\n";
    }
}
```

D. Fitur Ubah Data

Fitur ini digunakan untuk mengubah data bunga.

Source Code:

```
void ubahBunga(Bunga daftar[], int jumlah) {
    int ubah;
    cout << "Masukkan nomor bunga yang akan diubah: ";
    cin >> ubah;
    cin.ignore();
    if (ubah >= 1 && ubah <= jumlah) {
        cout << "Masukkan Nama Baru: ";
        getline(cin, daftar[ubah - 1].nama);
        cout << "Masukkan Harga Baru: ";
        cin >> daftar[ubah - 1].harga;
        cout << "Masukkan Stok Baru: ";
        cin >> daftar[ubah - 1].stok;
        cin.ignore();
        cout << "Bunga Berhasil Diubah!\n";
    } else {
        cout << "Nomor tidak valid!\n";
    }
}
```


E. Fitur Login

Fitur ini digunakan untuk login.

Source Code:

```
int main() {
    int pilihan;

    while (true) {
        cout << "\n===== MENU UTAMA =====\n";
        cout << "1. Register\n2. Login\n3. Keluar\nPilihan: ";
        cin >> pilihan;
        cin.ignore();

        if (pilihan == 1) {
            if (jumlahUser < 10) {
                cout << "Masukkan Nama: ";
                getline(cin, users[jumlahUser].nama);
                cout << "Masukkan NIM: ";
                getline(cin, users[jumlahUser].nim);
                jumlahUser++;
                cout << "Registrasi Berhasil!\n";
            } else {
                cout << "User penuh!\n";
            }
        } else if (pilihan == 2) {
            User currentUser;
            if (login(currentUser)) {
                if (currentUser.nama == "nilu") {
                    menuAdmin(daftarBunga, jumlahBunga);
                } else {
                    menuUser(currentUser, daftarBunga, jumlahBunga, transaksi,
jumlahTransaksi);
                }
            } else {
                break;
            }
        } else if (pilihan == 3) {
            cout << "Terima kasih telah menggunakan program ini.\n";
            break;
        } else {
            cout << "Pilihan tidak valid!\n";
        }
    }

    return 0;
}
```

F. Fitur Beli Bunga

Fitur ini digunakan untuk membeli bunga.

Source Code:

```
} else if (pilihan == 2) {
    tampilkanBunga(daftar, jumlah);
    int bungaIndex, jumlahBeli;
    cout << "Masukkan nomor bunga yang ingin dibeli: ";
    cin >> bungaIndex;
    cout << "Masukkan jumlah: ";
    cin >> jumlahBeli;
    cin.ignore();

    if (bungaIndex < 1 || bungaIndex > jumlah || jumlahBeli >
daftar[bungaIndex - 1].stok) {
        cout << "Data tidak valid atau stok tidak mencukupi!\n";
        continue;
    }

    daftar[bungaIndex - 1].stok -= jumlahBeli;
    transaksi[jumlahTransaksi++] = {user, daftar[bungaIndex - 1],
jumlahBeli};
    cout << "Pembelian berhasil!\n";
}
```

4. Uji Coba dan Hasil Output

```
===== MENU UTAMA =====
1. Register
2. Login
3. Keluar
Pilihan: 2
Masukkan Nama: nilu
Masukkan NIM: 012
Login Admin Berhasil!
```

Gambar 4.1 Menu Login Admin

```
===== MENU ADMIN =====
1. Tampilkan Bunga
2. Tambah Bunga
3. Hapus Bunga
4. Ubah Bunga
5. Logout
Pilihan: 1
```

```
===== DAFTAR BUNGA =====
```

No	Nama Bunga	Harga	Stok
1	Mawar	15000	10
2	Melati	12000	8
3	Tulip	25000	5
4	Anggrek	20000	7

Gambar 4.2 Tampilkan Bunga

```
===== MENU ADMIN =====
1. Tampilkan Bunga
2. Tambah Bunga
3. Hapus Bunga
4. Ubah Bunga
5. Logout
Pilihan: 2
Masukkan Nama Bunga: Baby's Breath
Masukkan Harga: 15000
===== MENU ADMIN =====
1. Tampilkan Bunga
2. Tambah Bunga
3. Hapus Bunga
4. Ubah Bunga
5. Logout
Pilihan: 2
Masukkan Nama Bunga: Sunflower
Masukkan Harga: 18000
Masukkan Stok: 10
Bunga Berhasil Ditambahkan!
```

Gambar 4.3 Tambah Bunga

```

===== MENU ADMIN =====
1. Tampilkan Bunga
2. Tambah Bunga
3. Hapus Bunga
4. Ubah Bunga
5. Logout
Pilihan: 1

===== DAFTAR BUNGA =====
No    Nama Bunga          Harga      Stok
-----
1     Mawar                15000      10
2     Melati               12000      8
3     Tulip                25000      5
4     Anggrek              20000      7
5     Baby's Breath        15000      15
6     Sunflower            18000      10

```

Gambar 4.4 Tampilkan Bunga

```

===== MENU ADMIN =====
1. Tampilkan Bunga
2. Tambah Bunga
3. Hapus Bunga
4. Ubah Bunga
5. Logout
Pilihan: 3
Masukkan nomor bunga yang akan dihapus: 4
Bunga berhasil dihapus!

```

Gambar 4.5 Hapus Bunga

```

===== MENU ADMIN =====
1. Tampilkan Bunga
2. Tambah Bunga
3. Hapus Bunga
4. Ubah Bunga
5. Logout
Pilihan: 1

===== DAFTAR BUNGA =====
No    Nama Bunga          Harga      Stok
-----
1     Mawar                15000      10
2     Melati               12000      8
3     Tulip                25000      5
4     Baby's Breath        15000      15
5     Sunflower            18000      10

```

Gambar 4.6 Tampilkan Bunga

```

===== MENU ADMIN =====
1. Tampilkan Bunga
2. Tambah Bunga
3. Hapus Bunga
4. Ubah Bunga
5. Logout
Pilihan: 4
Masukkan nomor bunga yang akan diubah: 2
Masukkan Nama Baru: Peony Bud
Masukkan Harga Baru: 14000
Masukkan Stok Baru: 19
Bunga Berhasil Diubah!

```

Gambar 4.7 Ubah Bunga

===== MENU ADMIN =====

1. Tampilkan Bunga
2. Tambah Bunga
3. Hapus Bunga
4. Ubah Bunga
5. Logout

Pilihan: 1

===== DAFTAR BUNGA =====

No	Nama Bunga	Harga	Stok

1	Mawar	15000	10
2	Peony Bud	14000	19
3	Tulip	25000	5
4	Baby's Breath	15000	15
5	Sunflower	18000	10

Gambar 4.8 Tampilkan Bunga

===== MENU ADMIN =====

1. Tampilkan Bunga
2. Tambah Bunga
3. Hapus Bunga
4. Ubah Bunga
5. Logout

Pilihan: 5

Logout berhasil! Kembali ke menu utama.

Gambar 4.9 Logout

===== MENU UTAMA =====

1. Register
2. Login
3. Keluar

Pilihan: 1

Masukkan Nama: aca

Masukkan NIM: 27

Registrasi Berhasil!

Gambar 4.10 Menu User Register

===== MENU UTAMA =====

1. Register
2. Login
3. Keluar

Pilihan: 2

Masukkan Nama: aca

Masukkan NIM: 27

Login Berhasil!

Gambar 4.11 Menu User Login

===== MENU USER =====

1. Tampilkan Bunga
2. Beli Bunga
3. Logout

Pilihan: 1

===== DAFTAR BUNGA =====

No	Nama Bunga	Harga	Stok
1	Mawar	15000	10
2	Peony Bud	14000	19
3	Tulip	25000	5
4	Baby's Breath	15000	15
5	Sunflower	18000	10

Gambar 4.12 Tampilkan Bunga User

===== MENU USER =====

1. Tampilkan Bunga
2. Beli Bunga
3. Logout

Pilihan: 2

===== DAFTAR BUNGA =====

No	Nama Bunga	Harga	Stok
1	Mawar	15000	10
2	Peony Bud	14000	19
3	Tulip	25000	5
4	Baby's Breath	15000	15
5	Sunflower	18000	10

Masukkan nomor bunga yang ingin dibeli: 4

Masukkan jumlah yang ingin dibeli: 5

Pembelian berhasil!

Gambar 4.13 Beli Bunga

===== MENU USER =====

1. Tampilkan Bunga
2. Beli Bunga
3. Logout

Pilihan: 3

Logout berhasil! Kembali ke menu utama.

Gambar 4.14 Logout User

5. Langkah-Langkah Git pada VSCode

A. Git Init

Menginisialisasi Folder di Git

```
PS C:\Users\ACER\Downloads\Praktikum_apl> git init
```

B. Git Add

Menambahkan Folder baru

```
PS C:\Users\ACER\Downloads\Praktikum_apl> git add .
```

C. Git Commit

Mengupdate Folder yang telah ditambahkan di Git

```
PS C:\Users\ACER\Downloads\Praktikum_apl> git commit -m "Finish Posttest 5"
[main 8b04e87] Finish Posttest 5
6 files changed, 521 insertions(+)
```

D. Git Push

Memperbarui Update-an Folder di git

```
PS C:\Users\ACER\Downloads\Praktikum_apl> git push -u origin main
```