

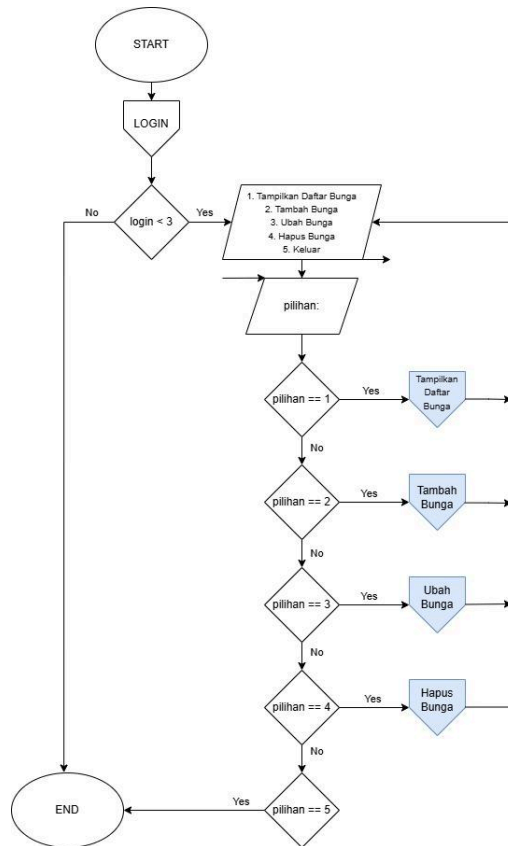
LAPORAN PRAKTIKUM
POSTTEST (2)
ALGORITMA PEMROGRAMAN LANJUT



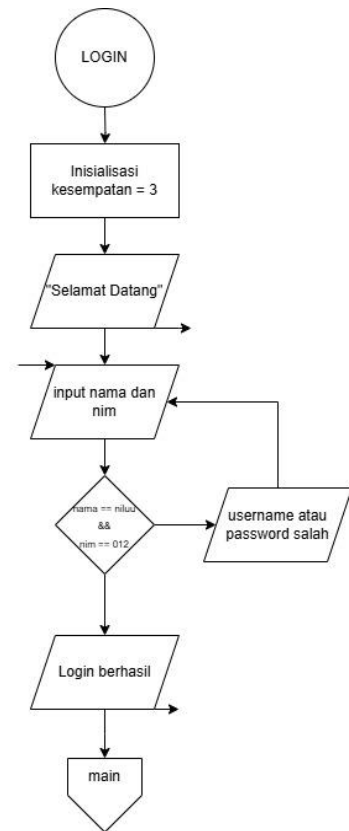
Disusun oleh:
Niluh Fincy Gloria Nathasia M (2409106012)
Kelas (A1'24)

PROGRAM STUDI INFORMATIKA
UNIVERSITAS MULAWARMAN
SAMARINDA
2025

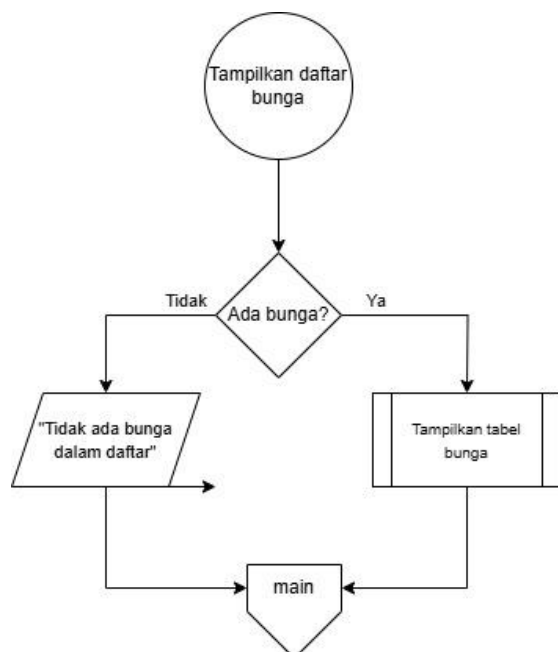
1. Flowchart



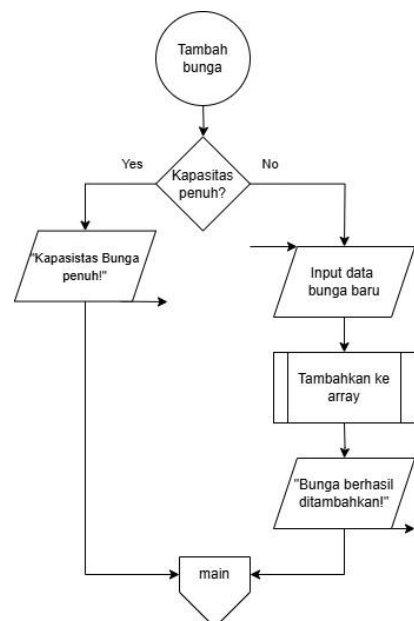
Gambar 1.1 Main Flowchart



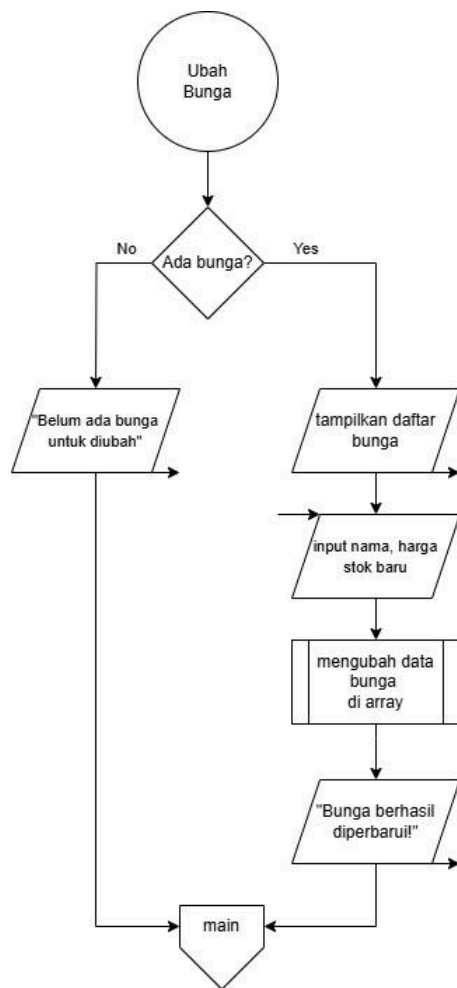
Gambar 1.2 Fungsi Login



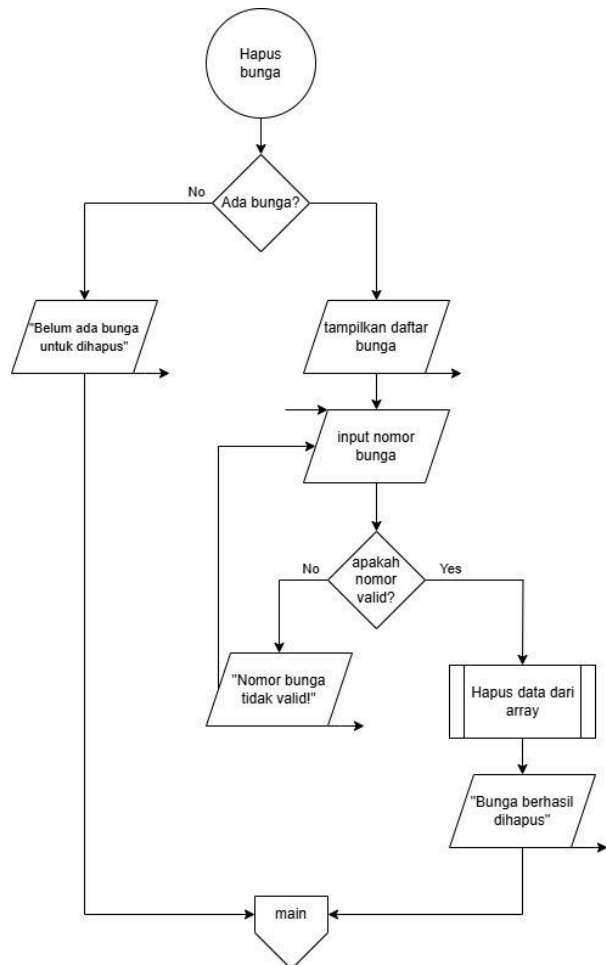
Gambar 1.3 Fungsi Read



Gambar 1.4 Fungsi Create



Gambar 1.5 Fungsi Update



Gambar 1.6 Fungsi Delete

2. Analisis Program

2.1 Deskripsi Singkat Program

Program Fresh Flower bertujuan untuk mengelola data penjualan bunga dengan sistem berbasis C++. Program ini memungkinkan pengguna untuk menambah, melihat, mengubah, dan menghapus data bunga secara interaktif. Fungsi Utama:

1. Login System > Memastikan hanya pengguna yang berwenang dapat mengakses sistem.
2. Tampilkan Daftar Bunga > Menampilkan data bunga dalam format tabel rapi.
3. Tambah Bunga > Menambahkan bunga baru beserta harga dan stoknya.
4. Ubah Data Bunga > Mengedit informasi bunga yang sudah ada.
5. Hapus Bunga > Menghapus data bunga dari daftar.
6. Menu Berulang > Program berjalan terus hingga pengguna memilih keluar.

Dengan program ini, pengguna dapat mengelola stok bunga dengan mudah, memastikan informasi bunga selalu diperbarui, dan mencegah akses tidak sah melalui fitur login.

2.2 Penjelasan Alur & Algoritma

A. Penjelasan Alur

A. Login

- Program meminta pengguna memasukkan Nama dan NIM.
- Jika cocok dengan akun yang telah ditentukan ("niluu", "012"), pengguna bisa masuk ke menu utama.
- Jika salah, pengguna diberi 3 kali kesempatan untuk mencoba lagi.
- Jika gagal 3 kali, program langsung berhenti.

B. Menu Utama

Setelah login berhasil, program menampilkan daftar pilihan fitur:

1. Menampilkan daftar bunga
 - Jika daftar bunga tidak kosong, program mencetak data bunga dalam bentuk tabel.
2. Menambah bunga baru
 - Program meminta pengguna memasukkan Nama Bunga, Harga, dan Stok.
 - Data bunga baru ditambahkan ke array jika kapasitas belum penuh.
3. Mengubah data bunga

- Program menampilkan daftar bunga dan meminta pengguna memilih bunga yang ingin diubah.
 - Setelah memilih, pengguna bisa memasukkan Nama, Harga, dan Stok yang baru.
 - Data bunga diperbarui sesuai dengan input pengguna.
4. Menghapus bunga
- Program menampilkan daftar bunga dan meminta pengguna memilih bunga yang ingin dihapus.
 - Jika nomor valid, data bunga dihapus dengan cara menggeser data bunga setelahnya ke atas.
5. Keluar
- Program menampilkan pesan “Terima kasih telah menggunakan sistem!” lalu berhenti.

C. Perulangan Menu

- Setelah pengguna memilih opsi dari menu (1-4), program akan kembali ke menu utama.
- Jika pengguna memilih 5 (Keluar), program akan menampilkan pesan perpisahan dan berhenti.

B. Algoritma

1. Mulai
2. Login
 - a. Minta pengguna memasukkan Nama dan NIM
 - b. Jika cocok dengan data yang telah ditentukan ("niluu", "012"), lanjut ke menu utama
 - c. Jika salah, kurangi jumlah kesempatan login
 - d. Jika gagal dalam 3 kali percobaan, program berhenti
3. Tampilkan Menu Utama
 - a. 1 - Tampilkan Daftar Bunga
 - Jika ada bunga dalam daftar, tampilkan dalam bentuk tabel
 - Jika tidak ada bunga, tampilkan pesan “Tidak ada bunga dalam daftar”
 - b. 2 - Tambah Bunga
 - Periksa apakah masih ada kapasitas untuk menambahkan bunga
 - Minta pengguna memasukkan Nama Bunga, Harga, dan Stok
 - Simpan data dalam array multidimensi bunga
 - Tambah jumlah bunga (jumlahBunga++)
 - c. 3 - Ubah Bunga
 - Tampilkan daftar bunga
 - Minta pengguna memilih nomor bunga yang ingin diubah
 - Jika nomor valid, minta input Nama Baru, Harga Baru, dan Stok Baru
 - Perbarui data bunga sesuai input pengguna
 - d. 4 - Hapus Bunga
 - Tampilkan daftar bunga
 - Minta pengguna memilih nomor bunga yang ingin dihapus

- Jika nomor valid, geser data bunga setelahnya ke atas untuk menutupi posisi yang dihapus

- Kurangi jumlah bunga (jumlahBunga--)

e. 5 - Keluar

- Tampilkan pesan perpisahan dan keluar dari program

4. Ulangi menu utama sampai pengguna memilih keluar (opsi 5)

5. Selesai

3. Source Code

A. Login

Fitur ini digunakan untuk masuk ke dalam program dengan menginput Nama dan NIM dengan batas percobaan 3 kali.

Source Code:

```
bool login() {
    string nama, nim;
    int kesempatan = 3;

    cout << "=====\n";
    cout << "    Selamat Datang di Fresh Flower  \n";
    cout << "=====\n";

    while (kesempatan > 0) {
        cout << "Masukkan Nama: ";
        getline(cin,nama);
        cout << "Masukkan NIM: ";
        cin >> nim;

        if (nama == "niluu" && nim == "012") {
            return true;
        } else {
            kesempatan--;
            cout << "Login gagal! Kesempatan tersisa: " << kesempatan <<
endl;
        }
    }
    return false;
}
```

B. Fitur Tambah Data (Create)

Fitur ini digunakan untuk menambahkan data baru ke dalam array. Data yang ditambah adalah nama bunga, harga, dan stok bunga.

```
int tambahBunga() {
    if (jumlahBunga < MAX_BUNGA) {
        cin.ignore();
        cout << "Masukkan Nama Bunga: ";
        getline(cin, bunga[jumlahBunga][0]);
    }
```

```

        cout << "Masukkan Harga Bunga/tangkai: ";
        cin >> bunga[jumlahBunga][1];

        cout << "Masukkan Stok Bunga: ";
        cin >> bunga[jumlahBunga][2];

        jumlahBunga++;
        cout << "Bunga berhasil ditambahkan!\n";
        return jumlahBunga;
    } else {
        cout << "Kapasitas bunga penuh!\n";
        return -1;
    }
}

```

C. Fitur Menampilkan Data (Read)

Fitur ini digunakan untuk menampilkan data bunga, nomor, nama, harga, dan stok bunga.

```

int tampilkanBunga() {
    if (jumlahBunga == 0) {
        cout << "Tidak ada bunga dalam daftar.\n";
        return 0;
    }

    cout << "=====\n";
    cout << "| No | Nama Bunga      | Harga | Stok          |\n";
    cout << "=====\n";

    for (int i = 0; i < jumlahBunga; i++) {
        cout << "| " << setw(2) << i + 1
            << " | " << left << setw(16) << bunga[i][0]
            << " | " << right << setw(8) << bunga[i][1]
            << " | " << setw(6) << bunga[i][2] << " |\n";
    }

    cout << "=====\n";
    return jumlahBunga;
}

```

D. Fitur Mengubah Data (Update)

Fitur ini digunakan untuk mengubah data bunga. Mulai dari nama, harga, dan stok bunga.


```

int ubahBunga() {
    if (jumlahBunga == 0) {
        cout << "Belum ada bunga untuk diubah.\n";
        return 0;
    }

    tampilkanBunga();
    int index;
    cout << "Masukkan nomor bunga yang ingin diubah: ";
    cin >> index;

    if (index > 0 && index <= jumlahBunga) {
        index--;
        cin.ignore();
        cout << "Masukkan Nama Baru: ";
        getline(cin, bunga[index][0]);

        cout << "Masukkan Harga Baru: ";
        cin >> bunga[index][1];

        cout << "Masukkan Stok Baru: ";
        cin >> bunga[index][2];

        cout << "Bunga berhasil diperbarui!\n";
        return 1;
    } else {
        cout << "Nomor bunga tidak valid!\n";
        return -1;
    }
}

```

E. Fitur Hapus Data (Delete)

Fitur ini digunakan untuk menghapus data dengan menginput nomor bunga yang akan dihapus datanya.

```

int hapusBunga() {
    if (jumlahBunga == 0) {
        cout << "Belum ada bunga untuk dihapus.\n";
        return 0;
    }

    tampilkanBunga();
    int index;
    cout << "Masukkan nomor bunga yang ingin dihapus: ";

```

```

    cin >> index;

    if (index > 0 && index <= jumlahBunga) {
        index--;
        for (int i = index; i < jumlahBunga - 1; i++) {
            bunga[i][0] = bunga[i + 1][0];
            bunga[i][1] = bunga[i + 1][1];
            bunga[i][2] = bunga[i + 1][2];
        }
        jumlahBunga--;
        cout << "Bunga berhasil dihapus!\n";
        return 1;
    } else {
        cout << "Nomor bunga tidak valid!\n";
        return -1;
    }
}

```

F. Fungsi Menampilkan Menu (main)

Fungsi ini digunakan untuk menampilkan menu yang akan dipilih oleh user, menampilkan, menambah, mengubah, dan menghapus data. Fungsi ini juga yang akan menjalankan fitur-fitur di atas.

```

int main() {
    if (!login()) {
        cout << "Login gagal 3 kali. Program berhenti.\n";
        return 0;
    }

    int pilihan;
    do {
        cout << "\n==== Menu Fresh Flower =====\n";
        cout << "1. Tampilkan Daftar Bunga\n";
        cout << "2. Tambah Bunga\n";
        cout << "3. Ubah Bunga\n";
        cout << "4. Hapus Bunga\n";
        cout << "5. Keluar\n";
        cout << "Pilih menu: ";
        cin >> pilihan;

        switch (pilihan) {
            case 1:
                tampilkanBunga();
                break;

```

```
        case 2:
            tambahBunga();
            break;
        case 3:
            ubahBunga();
            break;
        case 4:
            hapusBunga();
            break;
        case 5:
            cout << "Terima kasih telah menggunakan sistem!\n";
            break;
        default:
            cout << "Pilihan tidak valid. Silakan coba lagi.\n";
    }
} while (pilihan != 5);

return 0;
}
```

4. Uji Coba dan Hasil Output

4.1 Uji Coba

1. Skenario 1: Login Gagal dan Program Berhenti

Alur:

1. Program menampilkan halaman login.
2. Pengguna memasukkan Nama dan NIM yang salah.
3. Program menampilkan pesan error dan mengurangi kesempatan login.
4. Pengguna mencoba lagi sebanyak 3 kali tetapi tetap gagal.
5. Program menampilkan pesan “Login gagal 3 kali. Program berhenti.” dan keluar.

2. Skenario 2: Menampilkan Daftar Bunga dan Menambah Bunga Baru

Alur :

1. Pengguna berhasil login.
2. Program menampilkan menu utama.
3. Pengguna memilih “1. Tampilkan Daftar Bunga”, dan program menampilkan daftar bunga yang tersedia.
4. Pengguna memilih “2. Tambah Bunga” dan memasukkan Nama, Harga, dan Stok bunga baru.
5. Program menyimpan bunga dan menampilkan pesan “Bunga berhasil ditambahkan!”
6. Pengguna memilih “1. Tampilkan Daftar Bunga” lagi untuk memastikan bunga sudah ditambahkan.

4.2 Hasil Output

```
=====
    Selamat Datang di Fresh Flower
=====
Masukkan Nama: niluu
Masukkan NIM: 012
```

Gambar 4.1.2 Login

```
===== Menu Fresh Flower =====
1. Tampilkan Daftar Bunga
2. Tambah Bunga
3. Ubah Bunga
4. Hapus Bunga
5. Keluar
Pilih menu: 1
=====
| No | Nama Bunga      | Harga  | Stok  |
=====
| 1  | Mawar           | 15000  | 20   |
| 2  | Tulip           | 20000  | 15   |
| 3  | Melati          | 10000  | 30   |
| 4  | Lily            | 25000  | 10   |
=====
```

Gambar 4.2.2 Tampilkan Daftar Menu

```
===== Menu Fresh Flower =====
1. Tampilkan Daftar Bunga
2. Tambah Bunga
3. Ubah Bunga
4. Hapus Bunga
5. Keluar
Pilih menu: 2
Masukkan Nama Bunga: Sunflower
Masukkan Harga Bunga/tangkai: 15000
Masukkan Stok Bunga: 30
Bunga berhasil ditambahkan!
```

Gambar 4.2.3 Tambah Bunga

```

===== Menu Fresh Flower =====
1. Tampilkan Daftar Bunga
2. Tambah Bunga
3. Ubah Bunga
4. Hapus Bunga
5. Keluar
Pilih menu: 3
=====
| No | Nama Bunga      | Harga  | Stok  |
=====
| 1  | Mawar           | 15000  | 20    |
| 2  | Tulip           | 20000  | 15    |
| 3  | Melati          | 10000  | 30    |
| 4  | Lily            | 25000  | 10    |
| 5  | Sunflower       | 15000  | 30    |
=====
Masukkan nomor bunga yang ingin diubah: 3
Masukkan Nama Baru: Baby's Breath
Masukkan Harga Baru: 15000
Masukkan Stok Baru: 30
Bunga berhasil diperbarui!

===== Menu Fresh Flower =====
1. Tampilkan Daftar Bunga
2. Tambah Bunga
3. Ubah Bunga
4. Hapus Bunga
5. Keluar
Pilih menu: 1
=====
| No | Nama Bunga      | Harga  | Stok  |
=====
| 1  | Mawar           | 15000  | 20    |
| 2  | Tulip           | 20000  | 15    |
| 3  | Baby's Breath   | 15000  | 30    |
| 4  | Lily            | 25000  | 10    |
| 5  | Sunflower       | 15000  | 30    |
=====

```

Gambar 4.2.4 Ubah Bunga

```

===== Menu Fresh Flower =====
1. Tampilkan Daftar Bunga
2. Tambah Bunga
3. Ubah Bunga
4. Hapus Bunga
5. Keluar
Pilih menu: 4
=====
| No | Nama Bunga          | Harga  | Stok  |
=====
| 1  | Mawar               | 15000  | 20    |
| 2  | Tulip               | 20000  | 15    |
| 3  | Baby's Breath      | 15000  | 30    |
| 4  | Lily                | 25000  | 10    |
| 5  | Sunflower           | 15000  | 30    |
=====
Masukkan nomor bunga yang ingin dihapus: 1
Bunga berhasil dihapus!

```

```

===== Menu Fresh Flower =====
1. Tampilkan Daftar Bunga
2. Tambah Bunga
3. Ubah Bunga
4. Hapus Bunga
5. Keluar
Pilih menu: 1
=====
| No | Nama Bunga          | Harga  | Stok  |
=====
| 1  | Tulip               | 20000  | 15    |
| 2  | Baby's Breath      | 15000  | 30    |
| 3  | Lily                | 25000  | 10    |
| 4  | Sunflower           | 15000  | 30    |
=====

```

Gambar 4.2.5 Hapus Bunga

```

===== Menu Fresh Flower =====
1. Tampilkan Daftar Bunga
2. Tambah Bunga
3. Ubah Bunga
4. Hapus Bunga
5. Keluar
Pilih menu: 5
Terima kasih telah menggunakan sistem!

```

Gambar 4.2.6 Keluar

5. GitHub

A. Git Init

Menginisialisasi Folder di Git

```
PS D:\praktikum-apl\post-test\post-test-2> git init
```

B. Git Add

Menambahkan Folder baru

```
PS D:\praktikum-apl\post-test\post-test-2> git init
Initialized empty Git repository in D:/praktikum-apl/post-test/post-test-2/.git/
PS D:\praktikum-apl\post-test\post-test-2> git add .
```

C. Git Commit

Mengupdate Folder yang telah ditambahkan di Git

```
PS D:\praktikum-apl\post-test\post-test-2> git init
Initialized empty Git repository in D:/praktikum-apl/post-test/post-test-2/.git/
PS D:\praktikum-apl\post-test\post-test-2> git add .
PS D:\praktikum-apl\post-test\post-test-2> git commit -m "Finish Posttest 2"
[master (root-commit) 1941e5b] Finish Posttest 2
4 files changed, 367 insertions(+)
create mode 100644 .vscode/settings.json
create mode 100644 2409106012_NiluhFincyGloriaNathasiaM-PT2.cpp
create mode 100644 2409106012_NiluhFincyGloriaNathasiaM-PT2.exe
create mode 100644 v.xpp
PS D:\praktikum-apl\post-test\post-test-2> █
```

D. Git Push

Memperbarui Update-an Folder di git

```
PS D:\praktikum-apl\post-test\post-test-2> git push -u origin main
```