

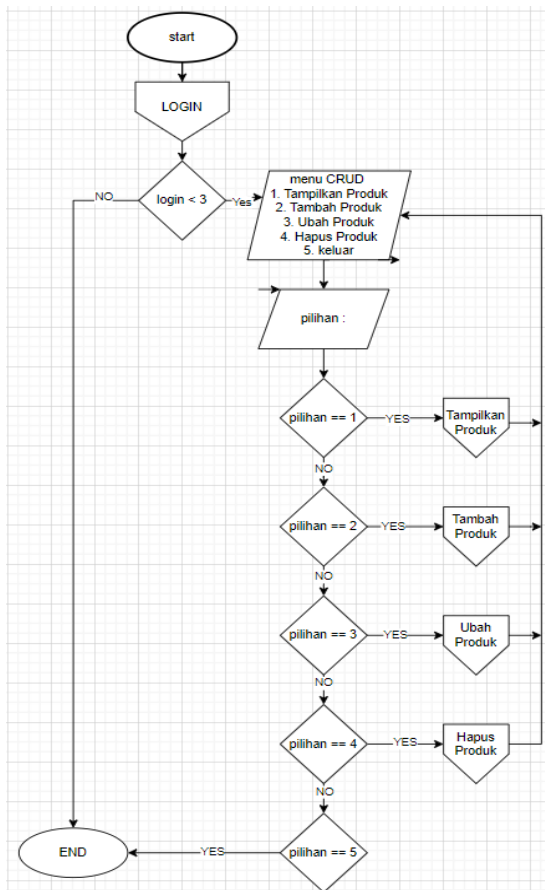
**LAPORAN PRAKTIKUM**  
**POSTTEST 2**  
**ALGORITMA PEMROGRAMAN LANJUT**



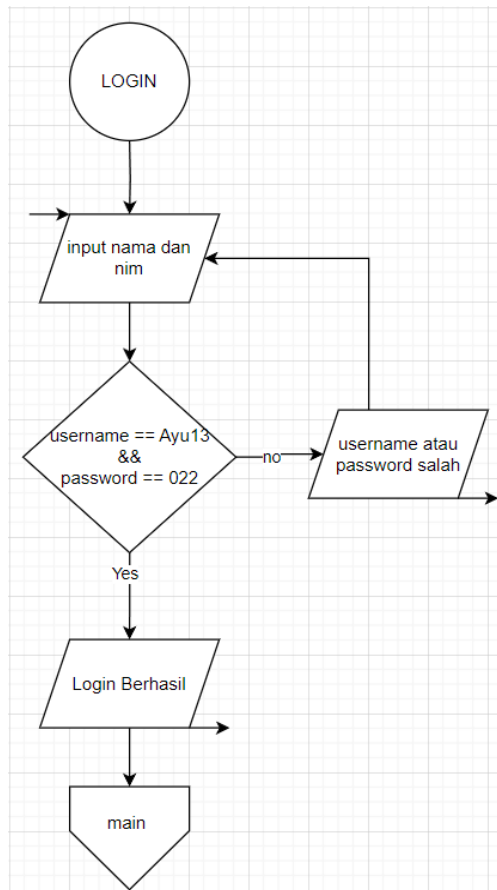
**Disusun oleh:**  
**Ayu Azzahrah Alwi (2409106022)**  
**Kelas A1'24**

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

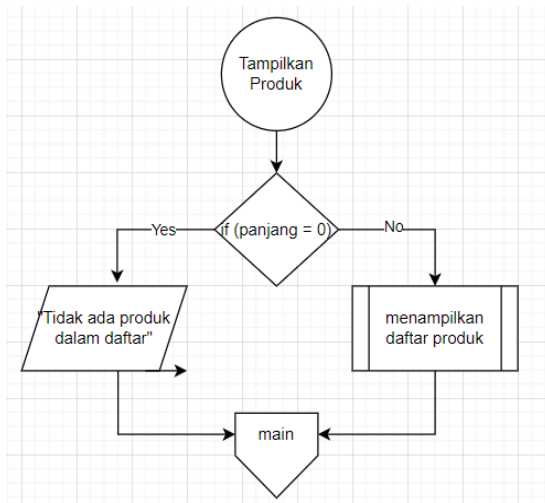
# 1. Flowchart



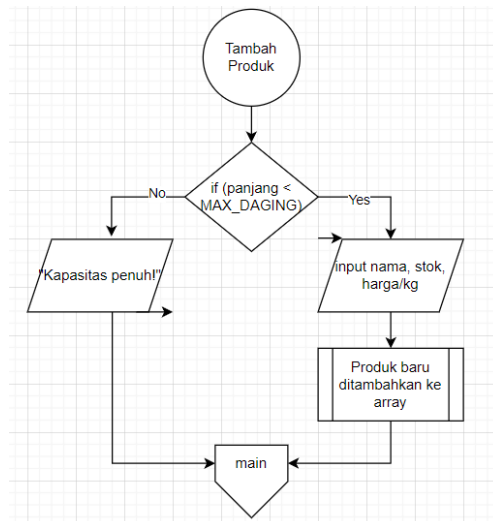
Gambar 1. 2 Main Flow



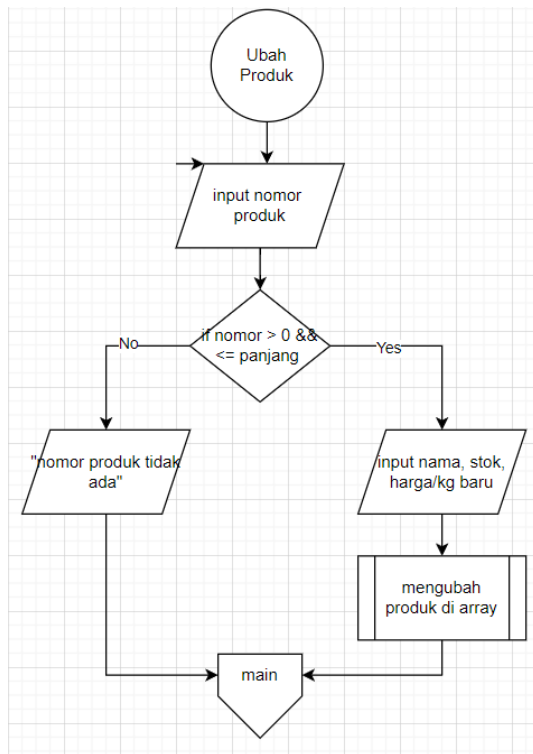
Gambar 1. 1 Login



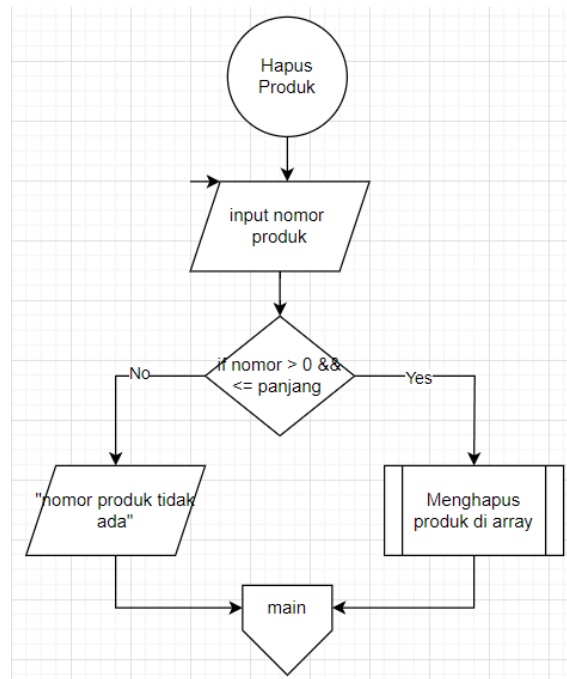
Gambar 1. 4 Tampilkan Produk



Gambar 1. 3 Tambah Produk



Gambar 1. 5 Ubah Produk



Gambar 1. 6 Hapus Produk

## 2. Analisis Program

### 2.1 Deskripsi Singkat Program

Program "Manajemen Toko Daging" dirancang untuk mengelola data produk daging dalam sebuah toko. Program ini memungkinkan pengguna untuk menambahkan, menampilkan, mengubah, dan menghapus data produk dengan sistem login yang memastikan keamanan akses. Manfaat utama dari program ini adalah membantu toko dalam mencatat stok dan harga produk secara sistematis serta memudahkan pengelolaan data produk menggunakan array dua dimensi.

### 2.2 Penjelasan Alur & Algoritma

Program ini terdiri dari beberapa tahap utama, yang mencakup proses login, navigasi menu, serta operasi CRUD (Create, Read, Update, Delete) pada data produk. Berikut adalah alur kerja program :

#### 1. Login Pengguna

- Pengguna harus memasukkan username dan 3 digit terakhir NIM untuk masuk ke sistem.
- Jika pengguna salah memasukkan data login sebanyak tiga kali, program akan berhenti.
- Jika login berhasil, pengguna diarahkan ke menu utama.

#### 2. Menampilkan Menu Utama

- Program menampilkan opsi yang bisa dipilih oleh pengguna:
  1. Tambah Produk
  2. Tampilkan Produk
  3. Ubah Produk
  4. Hapus Produk
  5. Keluar
- Pengguna memasukkan nomor pilihan menu, yang akan menentukan langkah selanjutnya.

#### 3. CRUD Data Produk

- **Tambah Produk:** Pengguna menginputkan nama produk, stok, dan harga, yang kemudian disimpan dalam array dua dimensi.
- **Tampilkan Produk:** Program mencetak daftar produk dalam format tabel agar lebih rapi dan mudah dibaca.

- **Ubah Produk:** Pengguna memilih produk berdasarkan nomor urutnya, lalu memasukkan data baru untuk mengganti informasi yang ada.
- **Hapus Produk:** Pengguna memilih produk yang ingin dihapus, lalu program menggeser data dalam array agar tetap terstruktur.

#### 4. Keluar dari Program

- Jika pengguna memilih opsi Keluar, program akan menampilkan pesan perpisahan dan berhenti.

#### Penjelasan Fungsi dalam Program:

- **tampilkanMenu()** Menampilkan daftar pilihan menu utama.
- **tambahProduk()** Menambahkan produk baru ke dalam array jika kapasitas belum penuh.
- **tampilkanProduk()** Menampilkan semua produk yang telah disimpan dalam format tabel.
- **ubahProduk()** Mengubah informasi produk tertentu berdasarkan input pengguna.
- **hapusProduk()** Menghapus produk yang dipilih dan menyesuaikan susunan array.
- **main()** Mengontrol seluruh proses eksekusi program dari login hingga interaksi dengan menu.

### 3. Source Code

#### A. Login

Fitur ini digunakan untuk memvalidasi user yang ingin menggunakan aplikasi

Source Code :

```
int main() {
    string User, Pass;
    int kesempatan = 0;
    bool login = false;

    while (kesempatan < 3 && !login) {
        cout << "Masukkan Username: ";
        cin >> User;
        cout << "Masukkan Password: ";
        cin >> Pass;

        if (User == "Ayu13" && Pass == "022") {
            login = true;
            cout << "==== LOGIN BERHASIL! =====\n";
        } else {
            cout << "==== Login gagal! Coba lagi. =====\n";
            kesempatan++;
        }
    }
    if (!login) {
        cout << "==== Anda mencoba 3 kali. Program berhenti. =====";
        return 0;
    }
}
```

#### B. Menu

Terdapat beberapa menu untuk memilih fitur pada aplikasi

Source Code :

```
#define MAX_DAGING 100
int panjang = 0;
string produk[MAX_DAGING][3];

int tampilkanMenu() {
    int pilihan ;
    cout << "Menu Program" << endl;
    cout << "1. Tampilkan Produk Daging" << endl;
    cout << "2. Tambah Produk Daging" << endl;
    cout << "3. Ubah Produk Daging" << endl;
}
```

```
cout << "4. Hapus Produk Daging" << endl;  
cout << "5. Keluar" << endl;  
cout << "Pilihan: ";  
cin >> pilihan;  
return pilihan;  
}
```

### C. CRUD Manajemen Toko Daging

Fitur **CRUD** dalam pemrograman ini mengacu pada empat operasi utama yaitu Create (membuat/menambahkan data baru), Read (Mengambil dan menampilkan data yang tersimpan), Update (Mengubah atau memperbarui data yang sudah ada.), Delet (Menghapus data dari Array.) yang digunakan untuk mengelola data produk dalam Array

#### Source Code:

```

int tambahProduk() {
    if (panjang < MAX_DAGING) {
        cout << "Masukkan Nama Produk Daging: ";
        cin.ignore();
        getline(cin, produk[panjang][0]);
        cout << "Masukkan Stok Produk Daging: ";
        cin >> produk[panjang][1];
        cout << "Masukkan Harga 1 kg: ";
        cin >> produk[panjang][2];
        panjang++;
        return 1;
    } else {
        return 0;
    }
}

int tampilkanProduk() {
    if (panjang == 0) {
        cout << "Tidak ada produk dalam daftar.\n";
        return 0;
    }
    cout << "\nDaftar Produk:\n";
    cout << "+-----+-----+-----+" << endl;
    cout << "| No | Nama Produk | Stok | Harga/Kg |" << endl;
    cout << "+-----+-----+-----+" << endl;
    for (int i = 0; i < panjang; i++) {
        cout << "| " << setw(2) << i + 1 << " | " << setw(13) << produk[i][0]
            << " | " << setw(4) << produk[i][1]
            << " | " << setw(5) << produk[i][2] << " |" << endl;
    }
    cout << "+-----+-----+-----+" << endl;
    return 1;
}

int ubahProduk() {
    int indeks;
    cout << "Masukkan nomor produk yang ingin diubah: ";
    cin >> indeks;
    if (indeks > 0 && indeks <= panjang) {
        cout << "Masukkan Nama Produk Baru: ";
        cin.ignore();
        getline(cin, produk[indeks - 1][0]);
        cout << "Masukkan Stok Baru: ";
        cin >> produk[indeks - 1][1];
        cout << "Masukkan Harga Baru: ";
        cin >> produk[indeks - 1][2];
        return 1;
    }
    else {
        return 0;
    }
}

```



```

}
int hapusProduk() {
    int indeks;
    cout << "Masukkan nomor produk yang ingin dihapus: ";
    cin >> indeks;
    if (indeks > 0 && indeks <= panjang) {
        for (int i = indeks - 1; i < panjang - 1; i++) {
            produk[i][0] = produk[i + 1][0];
            produk[i][1] = produk[i + 1][1];
            produk[i][2] = produk[i + 1][2];
        }
        panjang--;
        return 1;
    } else {
        return 0;
    }
}
}

```

## 4. Uji Coba dan Hasil Output

### 4.1 Uji Coba

Untuk memastikan bahwa program berjalan dengan baik dilakukan beberapa skenario uji coba dengan berbagai jenis input. Berikut adalah dua skenario utama yang diuji:

#### 1. Skenario 1: Login dengan Input yang Salah

- **Langkah-langkah:**

1. Menjalankan program.
2. Memasukkan username atau NIM yang salah sebanyak tiga kali.
3. Program menampilkan pesan error dan berhenti setelah tiga kali percobaan gagal.

- **Hasil yang diharapkan:**

- Jika pengguna salah memasukkan login tiga kali, program akan berhenti secara otomatis.
- Jika pengguna memasukkan data yang benar sebelum tiga kali percobaan, mereka dapat mengakses menu utama.

#### 2. Skenario 2: Menambahkan dan Menampilkan Produk

- **Langkah-langkah:**

1. Login dengan username dan NIM yang benar.
2. Memilih menu "Tambah Produk" dan memasukkan data produk (nama, stok, harga).
3. Memilih menu "Tampilkan Produk" untuk memastikan data yang telah dimasukkan muncul dengan format tabel yang rapi.

- **Hasil yang diharapkan:**

- Produk yang baru ditambahkan muncul dalam daftar dengan format tabel.
- Data produk tersimpan dengan benar dalam array.

- Jika tidak ada produk yang tersimpan, program menampilkan pesan "Tidak ada produk dalam daftar."

## 4.2 Hasil Output

```
PS C:\Users\ASUS GK\OneDrive\Dokumen\praktikum-apl\p
Masukkan Username: Ayu13
Masukkan Password: 123
===== Login gagal! Coba lagi. =====
Masukkan Username: Ayu
Masukkan Password: 2232
===== Login gagal! Coba lagi. =====
Masukkan Username: yuyu
Masukkan Password: 432
===== Login gagal! Coba lagi. =====
===== Anda mencoba 3 kali. Program berhenti. =====
PS C:\Users\ASUS GK\OneDrive\Dokumen\praktikum-apl\p
```

Gambar 4. 2 Gagal Login

```
xe
Masukkan Username: Ayu13
Masukkan Password: 022
===== LOGIN BERHASIL! =====
Menu Program
1. Tampilkan Produk Daging
2. Tambah Produk Daging
3. Ubah Produk Daging
4. Hapus Produk Daging
5. Keluar
Pilihan: █
```

Gambar 4. 1 Berhasil Login

```
Menu Program
1. Tampilkan Produk Daging
2. Tambah Produk Daging
3. Ubah Produk Daging
4. Hapus Produk Daging
5. Keluar
Pilihan: 2
Masukkan Nama Produk Daging: rusa
Masukkan Stok Produk Daging: 20
Masukkan Harga 1 kg: 50000
Produk Daging berhasil ditambahkan!
```

Gambar 4. 4 Tambah Produk

```
Menu Program
1. Tampilkan Produk Daging
2. Tambah Produk Daging
3. Ubah Produk Daging
4. Hapus Produk Daging
5. Keluar
Pilihan: 1

Daftar Produk:
+---+-----+-----+-----+
| No | Nama Produk | Stok | Harga/Kg |
+---+-----+-----+-----+
| 1 | sapi | 50 | 64000 |
| 2 | kambing | 44 | 36000 |
| 3 | rusa | 20 | 50000 |
+---+-----+-----+-----+
```

Gambar 4. 3 Tampilkan Produk

```

Menu Program
1. Tampilkan Produk Daging
2. Tambah Produk Daging
3. Ubah Produk Daging
4. Hapus Produk Daging
5. Keluar
Pilihan: 3
Masukkan nomor produk yang ingin diubah: 3
Masukkan Nama Produk Baru: unta
Masukkan Stok Baru: 32
Masukkan Harga Baru: 76000
Produk Daging berhasil diubah!

```

Gambar 4. 6 ubah produk

```

Menu Program
1. Tampilkan Produk Daging
2. Tambah Produk Daging
3. Ubah Produk Daging
4. Hapus Produk Daging
5. Keluar
Pilihan: 1

Daftar Produk:
+----+-----+-----+-----+
| No | Nama Produk | Stok | Harga/Kg |
+----+-----+-----+-----+
| 1 | sapi | 50 | 64000 |
| 2 | kambing | 44 | 36000 |
| 3 | unta | 32 | 76000 |
+----+-----+-----+-----+

```

Gambar 4. 5 daftar setelah diubah

```

Menu Program
1. Tampilkan Produk Daging
2. Tambah Produk Daging
3. Ubah Produk Daging
4. Hapus Produk Daging
5. Keluar
Pilihan: 4
Masukkan nomor produk yang ingin dihapus: 3
Produk Daging berhasil dihapus!

```

Gambar 4. 8 Hapus Produk

```

Menu Program
1. Tampilkan Produk Daging
2. Tambah Produk Daging
3. Ubah Produk Daging
4. Hapus Produk Daging
5. Keluar
Pilihan: 1

Daftar Produk:
+----+-----+-----+-----+
| No | Nama Produk | Stok | Harga/Kg |
+----+-----+-----+-----+
| 1 | sapi | 50 | 64000 |
| 2 | kambing | 44 | 36000 |
+----+-----+-----+-----+

```

Gambar 4. 7 Daftar Setelah Dihapus

```

Menu Program
1. Tampilkan Produk Daging
2. Tambah Produk Daging
3. Ubah Produk Daging
4. Hapus Produk Daging
5. Keluar
Pilihan: 5
Terima kasih telah menggunakan program ini!

```

Gambar 4. 9 Keluar dari perogram

## 5. Langkah-Langkah GIT

### 1. Git Init

Git init adalah perintah Git yang digunakan untuk menginisialisasi repository Git baru dalam sebuah folder. Cukup ketik “git init” pada terminal vscode

```
PS C:\Users\ASUS GK\OneDrive\Dokumen\praktikum-apl> git init
Reinitialized existing Git repository in C:/Users/ASUS GK/OneDrive/Dokumen/praktikum-apl/.git/
PS C:\Users\ASUS GK\OneDrive\Dokumen\praktikum-apl> █
```

Gambar 5. 1 git\_init

### 2. Git add

Git add adalah perintah Git yang digunakan untuk menambahkan perubahan pada file ke staging area sebelum dilakukan commit. Perintah ini memungkinkan Git mengetahui file mana saja yang akan dimasukkan dalam commit berikutnya

```
PS C:\Users\ASUS GK\OneDrive\Dokumen\praktikum-apl> git add .
PS C:\Users\ASUS GK\OneDrive\Dokumen\praktikum-apl> █
```

Gambar 5. 2 Git\_add

### 2. Git commit

Git commit adalah perintah Git yang digunakan untuk menyimpan perubahan yang telah ditambahkan ke staging area ke dalam repository. Setiap commit akan memiliki hash unik, pesan commit, dan menyimpan snapshot dari perubahan yang dilakukan.

```
PS C:\Users\ASUS GK\OneDrive\Dokumen\praktikum-apl> git commit -m "finish Post Test 2"
[main 80fb9df] finish Post Test 2
6 files changed, 269 insertions(+)
create mode 100644 kelas/pertemuan-2/pertemuan2.cpp
rename post-test/{post-test1-1 => post-test-1}/2409106022-AyuAzzahrahAlwi-PT-1.cpp (100%)
rename post-test/{post-test1-1 => post-test-1}/2409106022-AyuAzzahrahAlwi-PT-1.exe (100%)
rename post-test/{post-test1-1 => post-test-1}/2409106022-AyuAzzahrahAlwi-PT-1.pdf (100%)
create mode 100644 post-test/post-test-2/2409106022-AyuAzzahrahAlwi-PT-2.cpp
create mode 100644 post-test/post-test-2/2409106022-AyuAzzahrahAlwi-PT-2.exe
PS C:\Users\ASUS GK\OneDrive\Dokumen\praktikum-apl> █
```

Gambar 5. 3 Git\_Commit

### 3. Git remote

Git remote adalah perintah Git yang digunakan untuk menghubungkan repository lokal dengan repository remote (misalnya di GitHub, GitLab, atau Bitbucket). Dengan perintah ini, kita bisa mengelola koneksi ke repository jarak jauh, memungkinkan push, pull, dan fetch dari repository tersebut.

```
PS C:\Users\ASUS GK\OneDrive\Dokumen\praktikum-apl> git remote set-url https://github.com/ayuaa13/praktikum-apl.git
```

Gambar 5. 4 Git\_remot

#### 4. Git Push

Git push adalah perintah Git yang digunakan untuk mengirim (mengunggah) perubahan dari repository lokal ke repository remote (seperti GitHub, GitLab, atau Bitbucket). Perintah ini memastikan bahwa perubahan yang sudah dikomit di lokal tersedia di repository jarak jauh sehingga bisa diakses oleh orang lain atau untuk cadangan.

```
PS C:\Users\ASUS GK\OneDrive\Dokumen\praktikum-apl> git push -u origin main
```

Gambar 5. 5 Git\_Push