

**LAPORAN PRAKTIKUM**

**POSTTEST (6)**

**ALGORITMA PEMROGRAMAN LANJUT**



**Disusun oleh:**

**Dewi astuti (2409106007)**

**Kelas (A1'24)**

**PROGRAM STUDI INFORMATIKA**

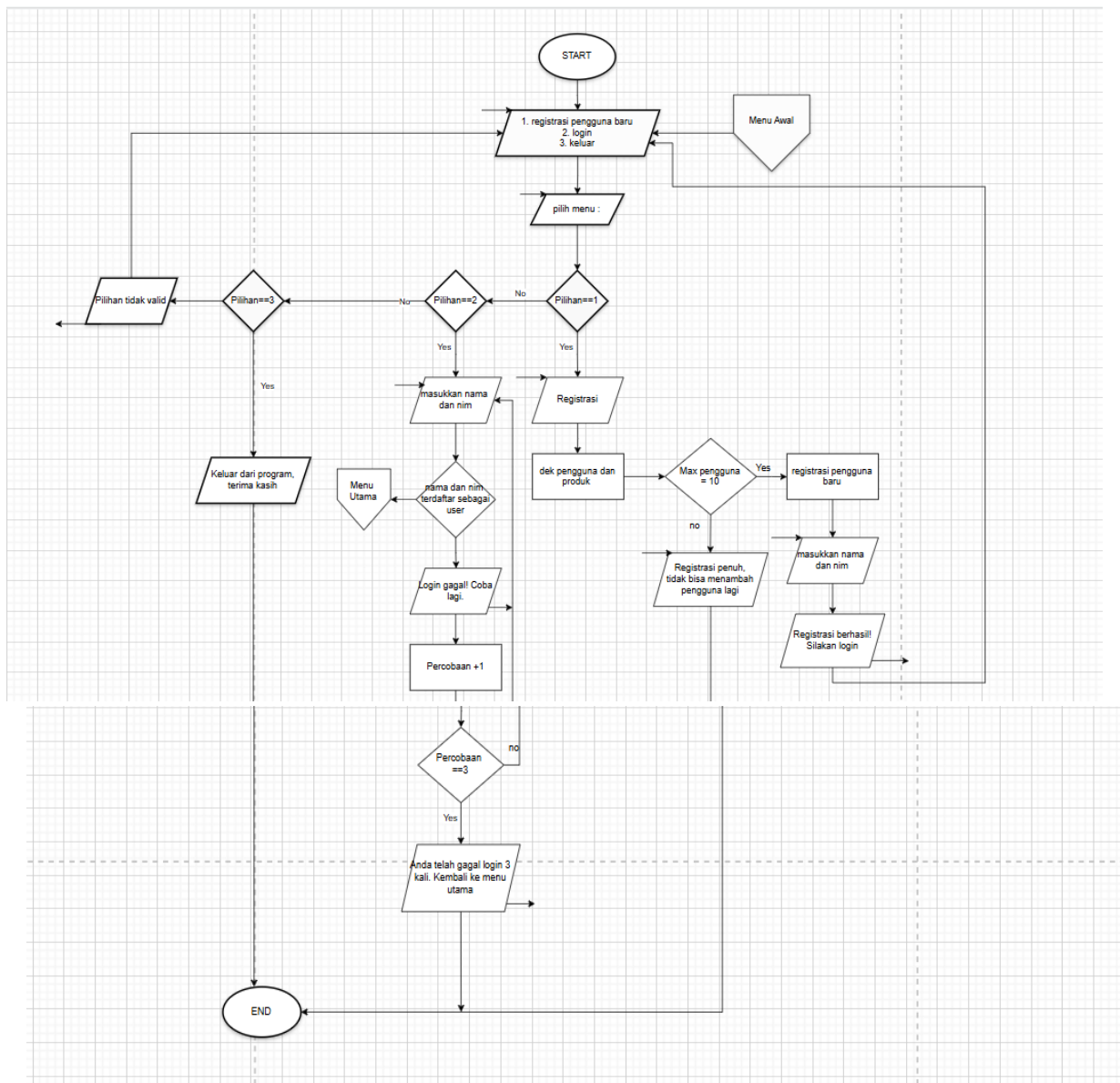
**UNIVERSITAS MULAWARMAN**

**SAMARINDA**

**2025**

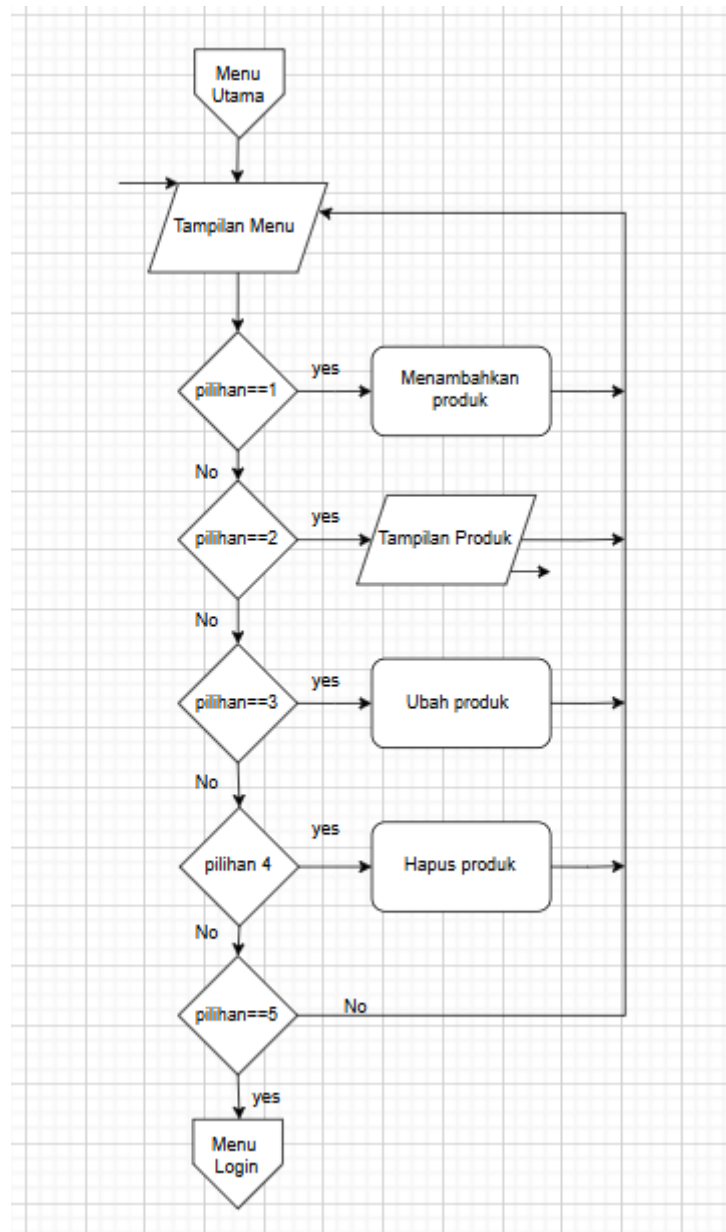
# 1. Flowchart

## 1.1 Menu Login



gambar 1.1 Menu Login

## 1.2 Menu Utama



*gambar 1.2 Menu Utama*

## 2. Analisis Program

Program Manajemen Produk Perangkat Jaringan ini dibuat menggunakan bahasa C++ dengan konsep Array of Struct. Program ini dirancang untuk membantu dalam pengelolaan data perangkat jaringan, seperti router, access point, dan switch. Program ini memiliki dua fitur utama, yaitu manajemen pengguna dan manajemen produk perangkat jaringan. Pengguna harus melakukan registrasi dan login terlebih dahulu sebelum dapat mengakses fitur manajemen

produk. Setelah login, pengguna dapat menambah, melihat, memperbarui, dan menghapus produk dalam sistem. Fungsi & Manfaat Utamanya yaitu ;

1. Manajemen Pengguna (Register & Login)

- Pengguna dapat mendaftarkan akun dengan memasukkan Nama dan NIM
- Sistem memastikan hanya pengguna yang telah terdaftar yang bisa mengakses fitur manajemen produk.
- Batas percobaan login sebanyak 3 kali untuk mencegah akses yang tidak sah.

2. Manajemen Produk Perangkat Jaringan (CRUD - Create, Read, Update, Delete)

- Menambah perangkat jaringan baru ke dalam daftar, seperti router atau switch.
- Menampilkan daftar perangkat dalam tabel yang rapi dengan informasi nama produk, tipe, harga, dan stok.
- Memperbarui informasi produk, termasuk harga dan jumlah stok.
- Menghapus produk dari daftar berdasarkan indeks.

3. Struktur Data yang Digunakan

- struct pengguna → Menyimpan informasi pengguna yang telah mendaftar.
- struct produk → Menyimpan informasi perangkat jaringan, seperti nama, tipe, harga, dan stok.

4. Keunggulan Program

- Mudah digunakan, dengan tampilan menu yang sederhana dan sistem input yang jelas.
- Menggunakan Array of Struct, sehingga memudahkan pengelolaan data pengguna dan produk.
- Memastikan keamanan login dengan sistem validasi yang membatasi percobaan login.

### 3. Source Code

#### 3.1 Struktur Data

- Struct Produk menyimpan semua informasi penting tentang produk perangkat jaringan seperti kode, nama, tipe, harga, dan jumlah stok.

```
struct Profil {  
    string Nama;  
    string Nim;  
};  
  
struct Pengguna {  
    Profil profil;  
};
```

- Struct Produk

```
struct Produk {  
    string kode;  
    string nama;  
    string tipe;  
    int harga;  
    int stok;  
};
```

#### 3.3 Registrasi & Login Pengguna

- Fungsi registrasi

```
void registrasi(Pengguna* daftar, int &jumlah)
```

- Fungsi login

```
int login(Pengguna* daftar, int jumlah)
```

#### 3.4 Manajemen Produk

- Tambah Produk

Menambahkan produk baru ke array daftar Produk.

```
void tambahProduk(Produk* daftar, int &jumlah)
```

- Tampilkan Produk (rekursif)
  - Menampilkan daftar produk satu per satu menggunakan **rekursi**.
  - Menghitung total stok semua produk.

```
void tampilkanProduk(Produk* arr, int index, int n, int &total)
```

Update Produk

Mengubah data produk berdasarkan index.

```
void updateProduk(Produk* daftar, int jumlah)
```

- Hapus Produk

Menghapus produk dan menggeser elemen array ke kiri agar tidak kosong.

```
void hapusProduk(Produk* daftar, int &jumlah)
```

### 3.5 Menu Utama

- Menu Login (di luar login)

User bisa mendaftar dan melakukan login.

```
1. Register  
2. Login  
3. Keluar
```

- **Menu Setelah Login**

Setelah login berhasil, user bisa mengelola produk.

```
1. Tambah Produk  
2. Tampilkan Produk  
3. Update Produk  
4. Hapus Produk  
5. Logout
```

### 3.6 Pengurutan Produk

Pengguna memilih salah satu opsi pengurutan, dan program akan mengurutkan produk menggunakan fungsi `sort()` dan comparator yfungsi ang sesuai. Tiga jenis pengurutan

yang bisa dipilih: berdasarkan nama (A-Z), harga (tertinggi ke terendah), atau stok (terbanyak ke tersedikit).

```
if (pilihanSort == 1) {  
    sort(arr, arr + n, urutkanNama); // Menyortir berdasarkan nama produk  
}
```

### 3.7 Fungsi Sort()

Fungsi Sort() digunakan untuk menyortir array . Fungsi comparator digunakan untuk menentukan urutan berdasarkan nama, harga, atau stok.

```
sort(arr, arr + n, urutkanNama);
```

### 3.8 Fungsi setw() dan cin.ignore()

setw(): Digunakan untuk mengatur lebar kolom saat menampilkan data, sehingga tampilan lebih rapi.

cin.ignore(): Digunakan untuk membersihkan input buffer setelah menggunakan cin, sehingga tidak terjadi masalah ketika menggunakan getline().

```
cout << left << setw(5) << "No" << setw(10) << "Kode" << setw(20) << "Nama  
Produk";  
cin.ignore(); // Membersihkan buffer input setelah menggunakan cin
```

## 4. Uji Coba dan Hasil Output

### 4. 1 Masuk Ke menu Login

```
PS D:\Praktikum-apl\post-test\post-t  
106007_DewiAstuti_PT_3 } ; if ($?) {  
  
Menu Login:  
1. Register  
2. Login  
3. Keluar  
Pilih menu: █
```

### 4.2 Masuk ke menu 1 yaitu register

```
Pilih menu: 1  
Masukkan Nama: dewi astuti  
Masukkan Nim: 2409106007  
Registrasi berhasil! Silakan login.
```

4.3 setelah melakukan registrasi bisa masuk ke menu login

```
2. Login
3. Keluar
Pilih menu: 2
Masukkan Username: dewi astuti
Masukkan Password: 2409106007
Login berhasil! Selamat datang, dewi astuti.
```

4.4 setelah login kita akan masuk ke menu utama lalu pilih menu 2

```
Menu Utama:
1. Tambah Produk
2. Tampilkan Produk
3. Update Produk
4. Hapus Produk
5. Logout
Pilih menu: 2

Daftar Produk:
```

No	Nama Produk	Tipe	Harga (Rp)	Stok
0	TP-LINK 2340	Router	450000	2
1	Cisco 2901	Router	3500000	5
2	MikroTik hAP AC2	Access Point	850000	10

4.5 setelah melihat tampilan produk kita lalu kita pilih no 1 tambah produk

```
Pilih menu: 1
Masukkan Nama Produk: NAS (Network Attached storage)
Masukkan Tipe Produk: cloud storage
Masukkan Harga Produk (Rp): 2000000
Masukkan Stok Produk: 3
Produk berhasil ditambahkan!
```

4. 6 lalu kita akan melihat produk yang sudah ditambahkan, kita akan memilih menu 2



Pilih menu: 2

Tampilkan Produk Berdasarkan:

- | 1 | Nama Produk A-Z
- | 2 | Harga Produk (Tertinggi ke Terendah)
- | 3 | Stok Produk (Terbanyak ke Tersedikit)
- | 4 | Kembali ke Menu Utama

Pilih opsi untuk sortir:

lalu setelah itu kita bisa pilih sorting produk berdasarkan 3 menu diatas

- menu 1

Daftar Produk:						
No	Kode	Nama Produk	Tipe	Harga (Rp)	Stok	Alamat Pointer
0	P002	Cisco 2901	Router	3500000	5	0x7ff7ea35e2e0
1	P003	MikroTik hAP AC2	Access Point	850000	10	0x7ff7ea35e348
2	P001	TP-LINK 2340	Router	450000	2	0x7ff7ea35e3b0

- menu 2

Daftar Produk:						
No	Kode	Nama Produk	Tipe	Harga (Rp)	Stok	Alamat Pointer
0	P002	Cisco 2901	Router	3500000	5	0x7ff7ea35e2e0
1	P003	MikroTik hAP AC2	Access Point	850000	10	0x7ff7ea35e348
2	P001	TP-LINK 2340	Router	450000	2	0x7ff7ea35e3b0

- menu 3

Daftar Produk:						
No	Kode	Nama Produk	Tipe	Harga (Rp)	Stok	Alamat Pointer
0	P003	MikroTik hAP AC2	Access Point	850000	10	0x7ff7ea35e2e0
1	P002	Cisco 2901	Router	3500000	5	0x7ff7ea35e348
2	P001	TP-LINK 2340	Router	450000	2	0x7ff7ea35e3b0
Total seluruh stok produk: 17 unit						

4.7 Setelah itu kita lihat, lalu kita pilih menu ke 3 yaitu update

```
Pilih menu: 3
Masukkan nomor produk yang ingin diupdate: 3
Masukkan Nama Produk baru: NAS
Masukkan Tipe Produk baru: Storage Cloud
Masukkan Harga Produk baru (Rp): 2500000
Masukkan Stok Produk baru: 4
Produk berhasil diupdate!
```

4.8 Setelah itu kita pilih menu 4 yaitu hapus

```
Pilih menu: 4
Masukkan nomor produk yang ingin dihapus: 3
Produk berhasil dihapus!

Menu Utama:
1. Tambah Produk
2. Tampilkan Produk
3. Update Produk
4. Hapus Produk
5. Logout
Pilih menu: 
```

4.9 setelah itu kita lihat tampilan produk apakah produk yang kita hapus sudah terhapus

```
Daftar Produk:
No    Nama Produk      Tipe           Harga (Rp)     Stok
-----
0     TP-LINK 2340     Router         450000         2
1     Cisco 2901       Router         3500000        5
2     MikroTik hAP AC2 Access Point    850000         10
```

4.10 lalu setelah itu kita pilih keluar untuk kembali kemenu login

```

5. Logout
Pilih menu: 5
Logout berhasil. Kembali ke menu utama.

Menu Login:
1. Register
2. Login
3. Keluar
Pilih menu: █

```

4.11 Setelah itu pilih menu ke 3 lalu keluar

```

Menu Login:
1. Register
2. Login
3. Keluar
Pilih menu: 3
Terima kasih! Program selesai.
PS D:\Praktikum-apl\post-test\post-test3> █

```

## 5. GIT

5.1 Git Init (Inisiasi Repository Git), lalu git add . , lalu gunakan perintah git commit -m "Finish Commit Post-test 3."

```

PS D:\Praktikum-apl> git add .
warning: in the working copy of '.vscode/settings.json', LF will be replaced by CRLF the next time Git touches it
PS D:\Praktikum-apl> git commit -m "Finish commit Posttest 4"
[main 2e008c3] Finish commit Posttest 4
4 files changed, 268 insertions(+)
create mode 100644 post-test/post-test6/2409106007_DewiAstuti_PT6.cpp
create mode 100644 post-test/post-test6/2409106007_DewiAstuti_PT6.exe
create mode 100644 post-test/post-test6/2409106007_DewiAstuti_PT6.pdf

```

5.2 Lalu ketik 'git push -u origin main', untuk mengupload semua file tadi ke cloud github.

```

PS D:\Praktikum-apl> git push origin main
Enumerating objects: 18, done.
Counting objects: 100% (18/18), done.
Delta compression using up to 12 threads
Compressing objects: 100% (11/11), done.
Writing objects: 100% (12/12), 850.62 KiB | 4.97 MiB/s, done.
Total 12 (delta 2), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To https://github.com/dewiasttt/praktikum-apl.git
  77a79b5..956cb96  main -> main
PS D:\Praktikum-apl> █

```