

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

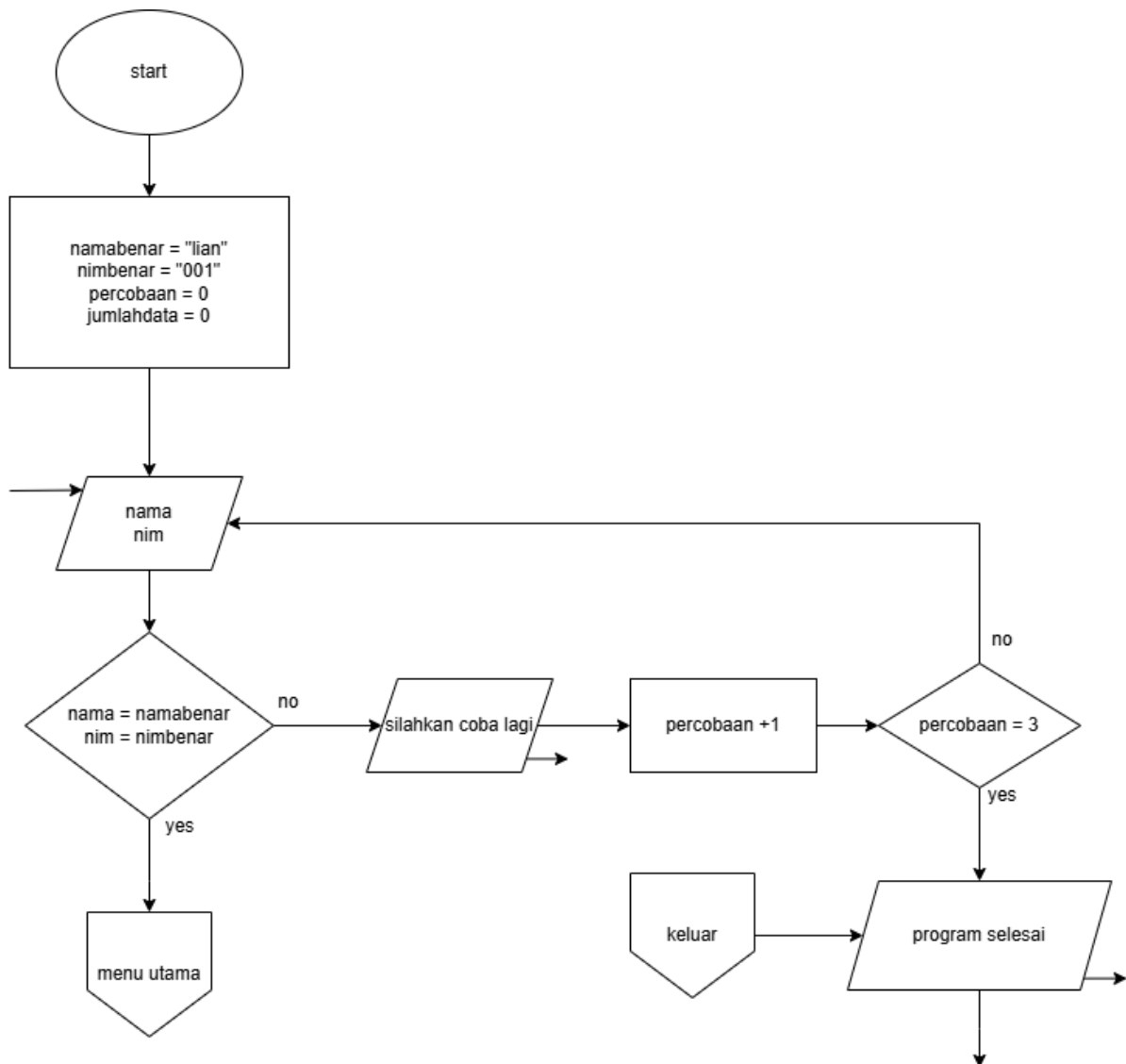


**Disusun oleh:**  
**Dwiki Aprilian Aryanda (2409106001)**  
**Kelas (A1 '24)**

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

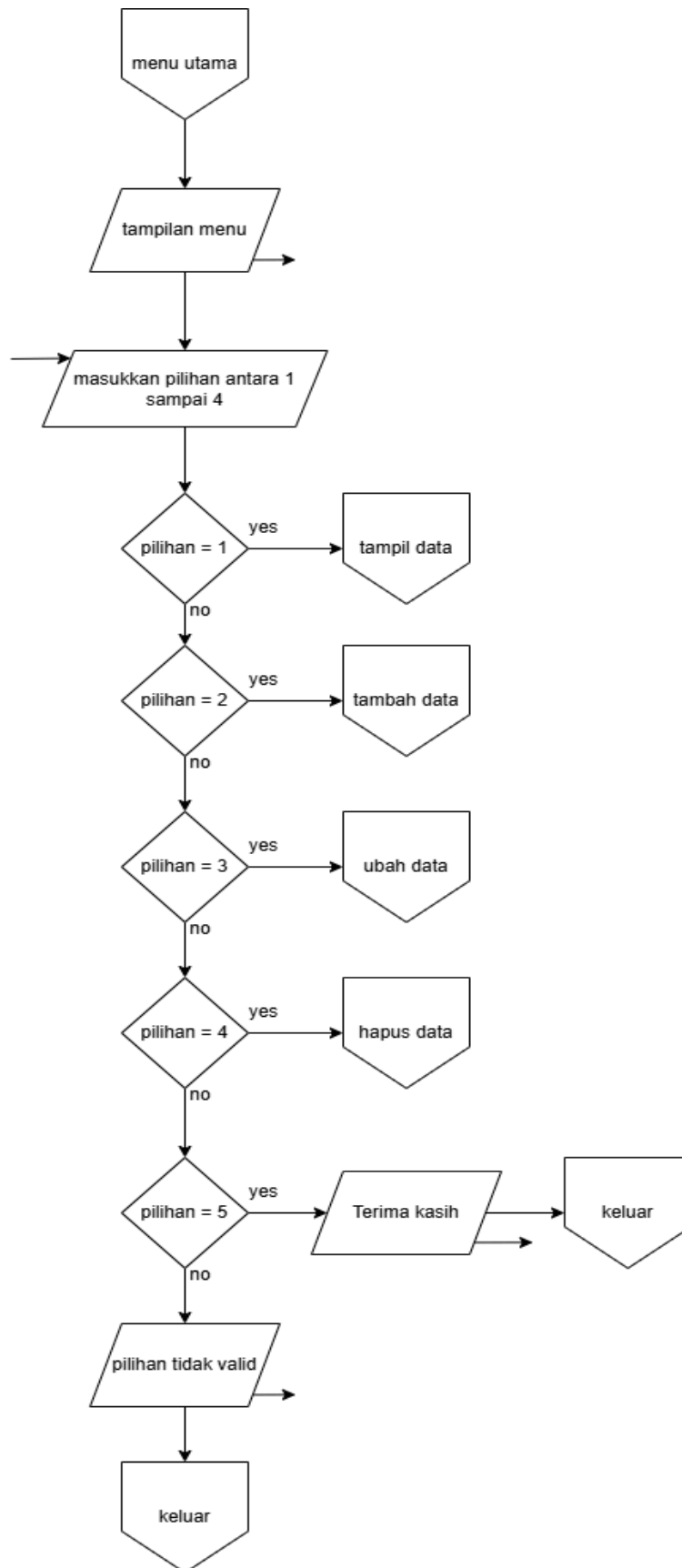
## 1. Flowchart

### 1.1 Menu Login



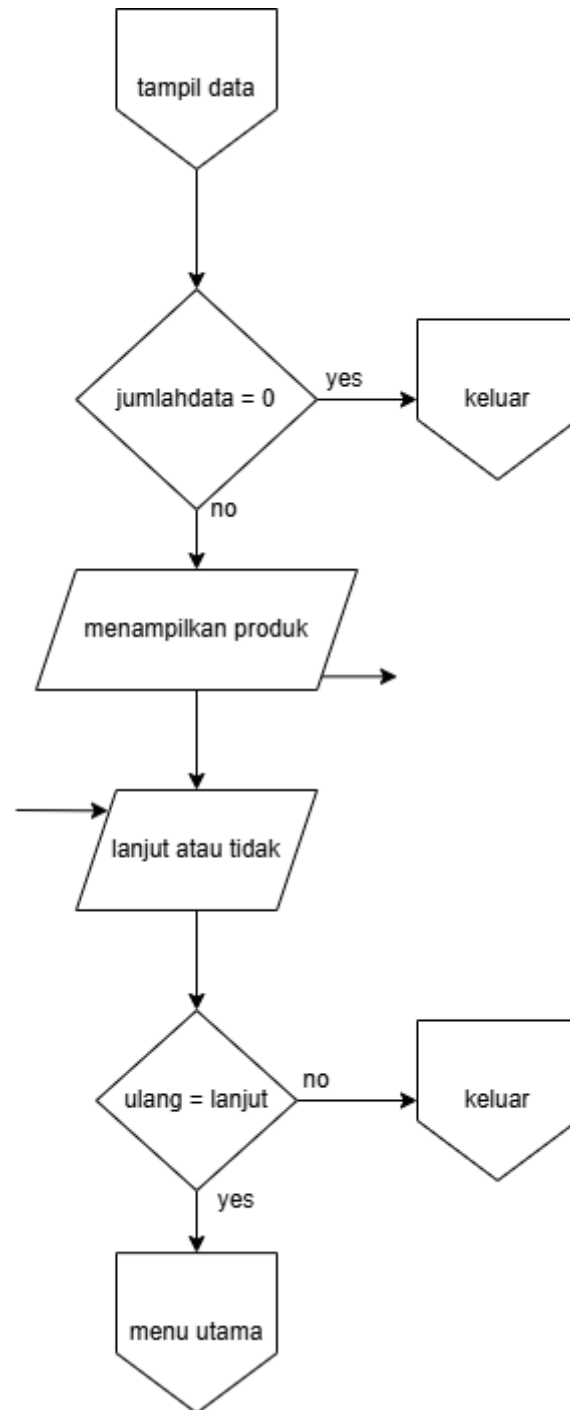
Gambar 1.1 Menu Login

## 1.2 Menu Utama



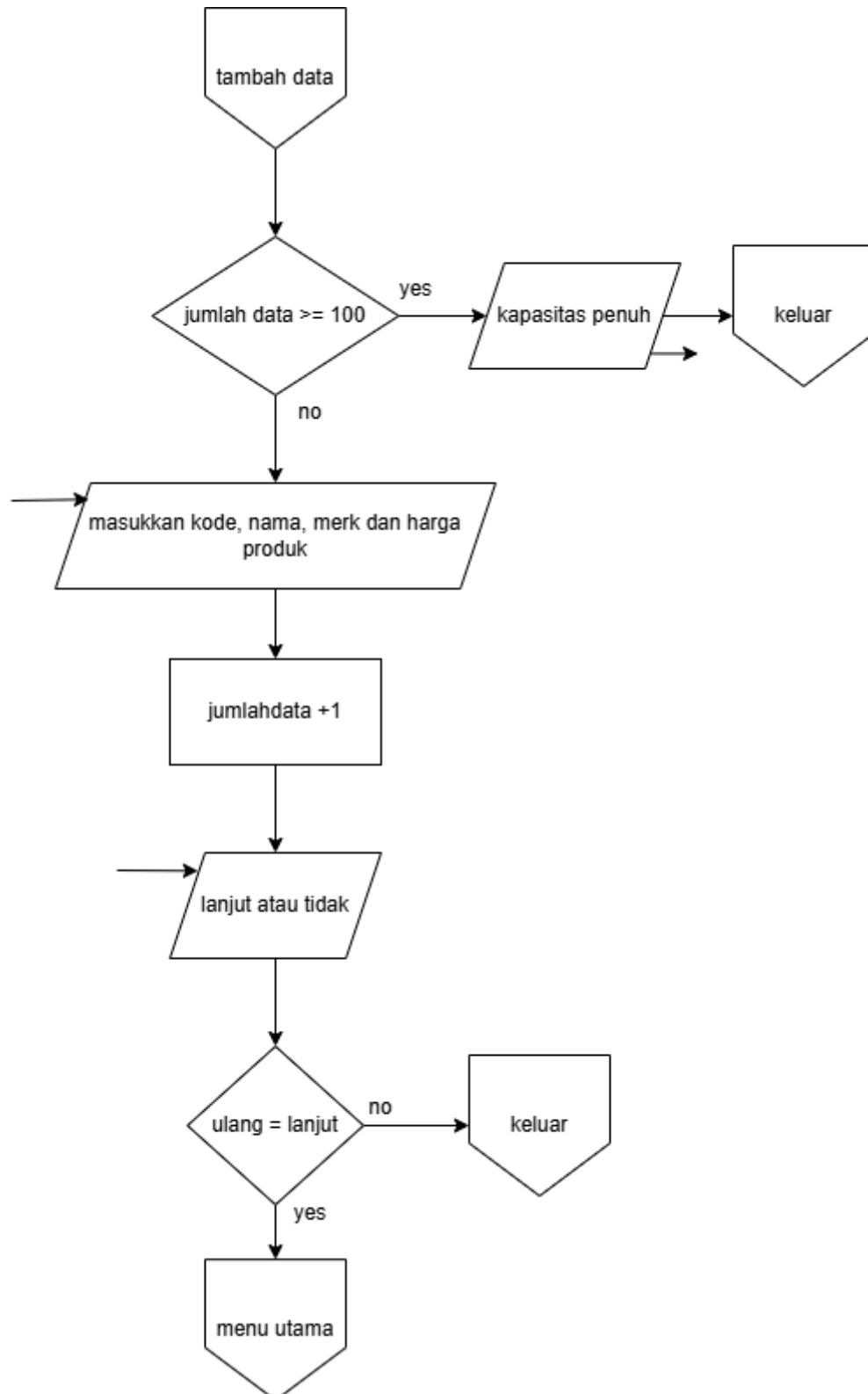
Gambar 1.2 Menu Konversi

### 1.3 Tampil Data



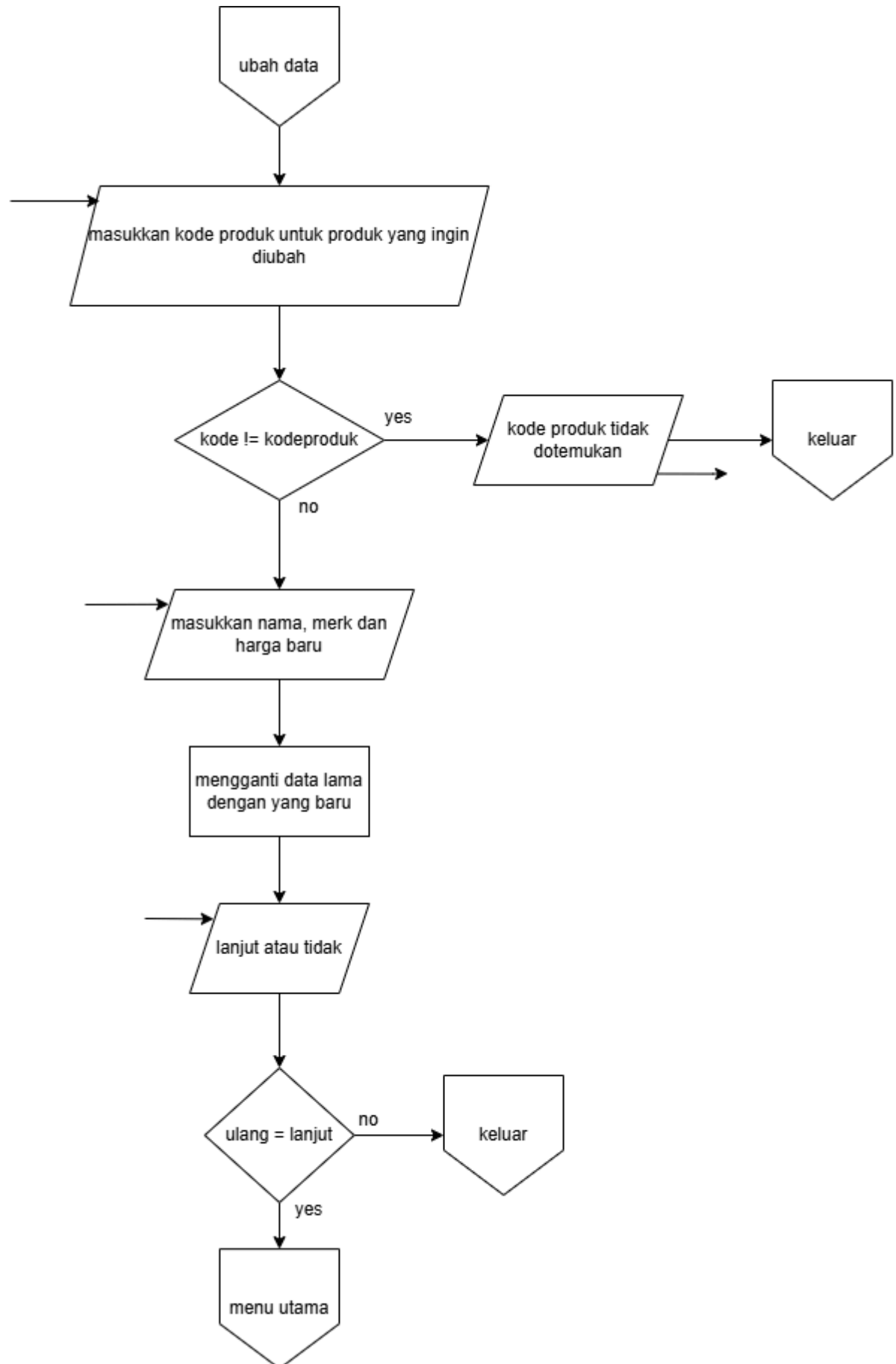
Gambar 1.3 Tampil Data

## 1.4 Tambah Data



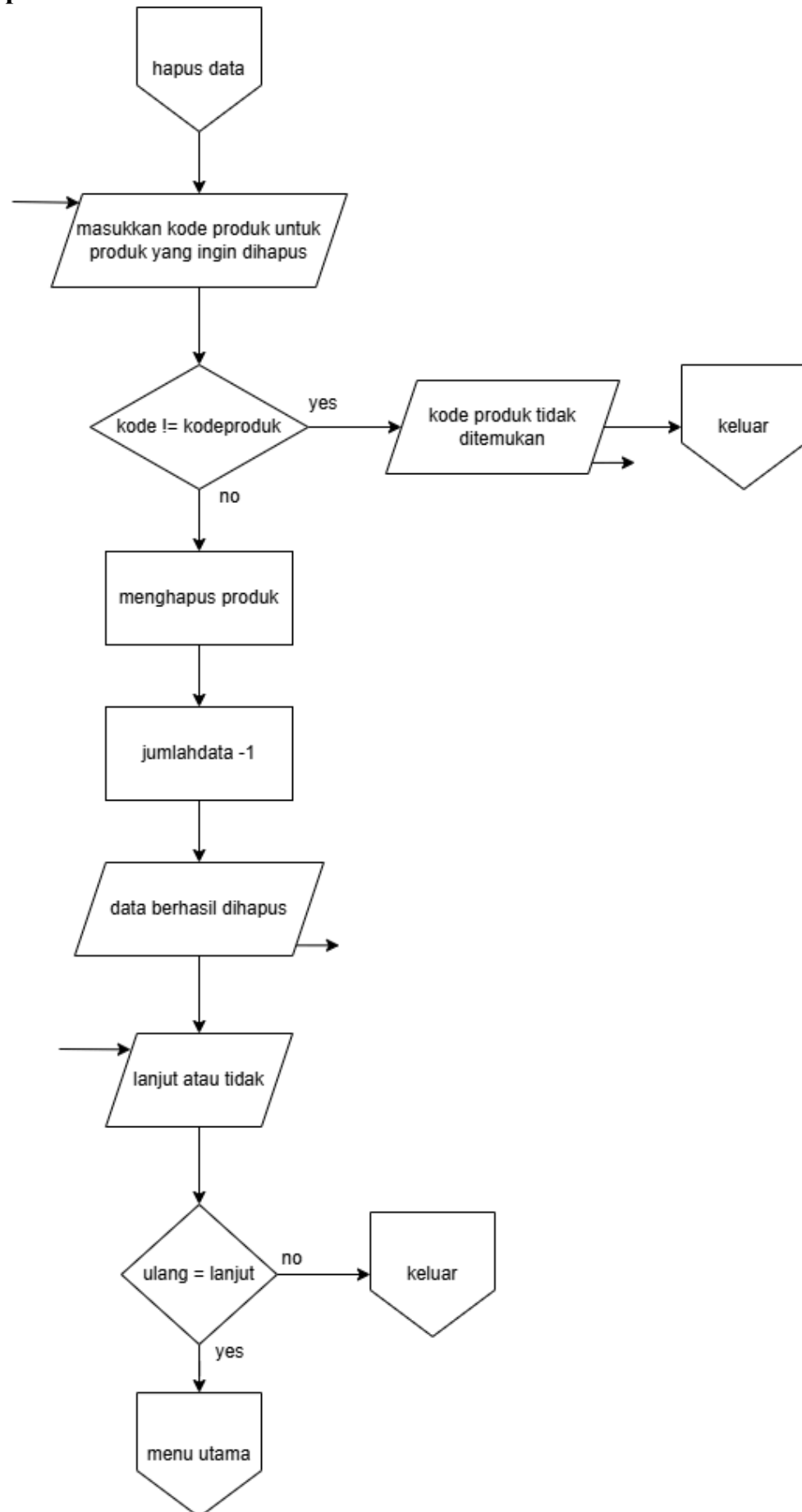
Gambar 1.4 Tambah Data

## 1.5 Ubah Data



Gambar 1.5 Ubah Data

## 1.6 Hapus Data



Gambar 1.6 Hapus Data

## **2. Analisis Program**

### **2.1 Deskripsi Singkat Program**

Program ini bertujuan untuk mengelola data produk handphone, di mana pengguna harus login dengan nama "lian" dan NIM "001" dalam maksimal 3 percobaan untuk mengakses fitur manajemen; setelah login berhasil, pengguna dapat menambah, menampilkan dalam tabel rapi menggunakan `setw`, mengubah, atau menghapus data handphone (kode, nama, merk, harga) yang disimpan dalam empat array satu dimensi berkapasitas 100 entri, dengan fungsi berbasis `int` dan `bool` untuk melaporkan status operasi, sehingga memungkinkan pengelolaan data secara efisien hingga pengguna memilih keluar dari menu berulang.

### **2.2 Penjelasan Alur & Algoritma**

#### **2.2.1 Penjelasan Alur**

##### **Inisialisasi:**

- Program dimulai dengan mendeklarasikan empat array satu dimensi (kodeProduk, namaProduk, merkProduk, hargaProduk) untuk menyimpan data handphone, masing-masing berkapasitas 100 entri, serta variabel jumlahData yang diinisialisasi dengan 0 untuk melacak jumlah entri

##### **Proses Login:**

- Fungsi login() dipanggil pertama kali di main(). Pengguna diminta memasukkan nama dan NIM.
- Input dibandingkan dengan nilai tetap: nama "lian" dan NIM "001".
- Jika cocok, fungsi mengembalikan true, dan program lanjut ke menu utama. Jika salah, percobaan bertambah (maksimal 3 kali); setelah 3 kali gagal, fungsi mengembalikan false, dan



program berhenti.

#### **Menu Utama:**

- Jika login berhasil, program menampilkan menu dengan 5 opsi (4 menu + keluar).
- Pengguna memilih opsi dan program akan berjalan sesuai yang pengguna pilih

#### **Pilihan 1: Tampil Data:**

- Jika jumlahData adalah 0, menampilkan "Tidak ada data handphone!" dan mengembalikan 0.
- Jika ada data, menampilkan tabel dengan kolom "Kode", "Nama Produk", "Merk", dan "Harga" menggunakan setw untuk perataan, lalu mengembalikan jumlahData.

#### **Pilihan 2: Tambah Data:**

- Memeriksa apakah jumlahData < 100. Jika penuh, mengembalikan false.
- Meminta input kode, nama, merk, dan harga; getline() digunakan untuk nama dan merk agar bisa menerima spasi, lalu menambah jumlahData dan mengembalikan true

#### **Pilihan 3: Ubah Data:**

- Meminta kode produk, mencari di array kodeProduk. Jika ditemukan, meminta input baru untuk nama, merk, dan harga, lalu mengembalikan true. Jika tidak ditemukan, mengembalikan false.

#### **Pilihan 4: Hapus Data:**

- Meminta kode produk, mencari di array, dan jika ditemukan,

menggeser elemen berikutnya untuk mengisi posisi yang dihapus, mengurangi jumlahData, lalu mengembalikan true. Jika tidak ditemukan, mengembalikan false.

**Pengulangan atau Berhenti:**

- Pengguna diminta memilih apakah ingin melanjutkan (y/n).
- Jika 'y' atau 'Y', menu ditampilkan lagi; jika tidak, program berhenti.

**Output Akhir:**

- Pesan penutup ditampilkan saat pengguna keluar atau gagal login 3 kali

### 2.2.2 Algoritma

1. Mulai

2. Deklarasi Variabel :

- string : kodeProduk[100], namaProduk[100], merkproduk[100]
- int : hargaProduk[100], jumlahData = 0
- double : nilai

3. Proses Login

Dekslarasi Variabel:

- string : nama, nim, namabenar = "lian", nimbenar = "001"
- int : percobaan = 0

Input nama and nim

- a. Jika input sama dengan namabenar dan nimbenar maka login berhasil dan lanjut ke menu utama
- b. Jika input salah maka percobaan +1 dan kembali ke menu login
- c. Jika percobaan = 3 maka program dihentikan

4. Tampilkan Menu Utama

Dekslarasi Variabel:

- char : ulang

Input pilihan

- a. Jika pilihan adalah 1, 2, 3, atau 4 maka pengguna akan menuju program sesuai yang pengguna pilih
- b. Jika pilih 5 atau angka selain itu maka program akan dihentikan

5. Pilihan 1: Tampil Data

- a. Jika jumlahData = 0, maka program akan berhenti
- b. Jika jumlahData >0, maka data akan ditampilkan

6. Pilihan 2: Tambah Data

- a. Jika jumlahData >= 100 atau jumlahData = 0, maka program akan berhenti

- b. Jika selain itu, maka pengguna dapat menambahkan kode, nama, merk, dan harga produk

7. Pilihan 3: Ubah Data

Pengguna memasukkan kode produk dari produk yang ingin diubah

- a. Jika kode  $\neq$  kodeProduk, program akan berhenti
- b. Jika kode = kodeProduk, pengguna dapat mengubah nama, merk dan harga produk

8. Pilihan 4: Hapus Data

Pengguna memasukkan kode produk dari produk yang ingin diubah

- a. Jika kode  $\neq$  kodeProduk, program akan berhenti
- b. Jika kode = kodeProduk, produk akan dihapus

9. Selesai

### 3. Source Code

#### A. Login

Fitur ini digunakan sebagai syarat untuk menjalankan program konversi suhu. Data yang diminta berupa username dan password.

Source Code:

```
int main() {
    string nama, nim;
    string namabenar = "lian";
    string nimbenar = "001";
    int percobaan = 0;

    while (percobaan < 3) {
        cout << "\n=== Login Sistem ===" << endl;
        cout << "Masukkan Nama: ";
        cin >> nama;
        cin.ignore();
        cout << "Masukkan NIM: ";
        cin >> nim;
        cin.ignore();

        if (nama == namabenar && nim == nimbenar) {
            cout << "Login berhasil!" << endl;
            break;
        } else {
            percobaan++;
            cout << "Login gagal! Sisa percobaan: " << (3 - percobaan) << endl;
        }
    }
    if (percobaan >= 3) {
        cout << "Cukurukuk gak bisa nyoba login lagi" << endl;
        return 0;
    }
}
```

## B. Menu Utama

Menu Utama berisi beberapa pilihan fitur. Pilihan 1 sampai 4 merupakan fitur utama dan pilihan 5 merupakan fitur keluar dari program.

### Source Code:

```
char ulang;
do {
    cout << "=====\n";
    cout << "=== Manajemen Produk Handphone ===\n";
    cout << "=====\n";
    cout << "|1. Tampilkan Data          |\n";
    cout << "|2. Tambah Data             |\n";
    cout << "|3. Ubah Data               |\n";
    cout << "|4. Hapus Data              |\n";
    cout << "|5. Keluar                  |\n";
    cout << "=====\n";
    cout << "Pilih menu (1-5): ";

    int pilihan;
    cin >> pilihan;
```

## C. Menu Tampil Data

Fitur untuk menampilkan data produk

### Source Code:

```
case 1: // Tampilkan Data
    if (jumlahData == 0) {
        cout << "Tidak ada data handphone!" << endl;
    } else {
        cout << "\n=== Data Handphone ===" << endl;
        cout << setw(10) << "Kode" << setw(20) << "Nama Produk" <<
        setw(15) << "Merk" << setw(15) << "Harga" << endl;
        cout << string(60, '-') << endl;
        for (int i = 0; i < jumlahData; i++) {
            cout << setw(10) << kodeProduk[i]
                << setw(20) << namaProduk[i]
                << setw(15) << merkProduk[i]
                << setw(15) << hargaProduk[i] << endl;
        }
    }
    break;
```

## D. Menu Tambah Data

Fitur untuk menambahkan data.

Source Code:

```
case 2: // Tambah Data
    if (jumlahData >= 100) {
        cout << "Kapasitas penuh!" << endl;
    } else {
        cout << "\n=== Tambah Data Handphone ===" << endl;
        cout << "Masukkan Kode Produk: ";
        cin >> kodeProduk[jumlahData];
        cout << "Masukkan Nama Produk: ";
        cin.ignore();
        getline(cin, namaProduk[jumlahData]);
        cout << "Masukkan Merk Produk: ";
        getline(cin, merkProduk[jumlahData]);
        cout << "Masukkan Harga: ";
        cin >> hargaProduk[jumlahData];
        jumlahData++;
        cout << "Data berhasil ditambahkan!" << endl;
    }
    break;
```

## E. Menu Ubah Data

Fitur untuk mengubah data.

Source Code:

```
case 3: // Ubah Data
{
    string kode;
    cout << "\nMasukkan Kode Produk yang akan diubah: ";
    cin >> kode;
    bool ditemukan = false;
    for (int i = 0; i < jumlahData; i++) {
        if (kodeProduk[i] == kode) {
            cout << "Masukkan Nama Produk baru: ";
            cin.ignore();
            getline(cin, namaProduk[i]);
            cout << "Masukkan Merk Produk: ";
            getline(cin, merkProduk[i]);
            cout << "Masukkan Harga baru: ";
            cin >> hargaProduk[i];
            cout << "Data berhasil diubah!" << endl;
        }
    }
}
```

```
                ditemukan = true;
                break;
            }
        }
        if (!ditemukan) {
            cout << "Kode produk tidak ditemukan!" << endl;
        }
    }
    break;
```



## F. Menu Hapus Data

Fitur untuk menghapus data.

Source Code:

```
case 4: // Hapus Data
{
    string kode;
    cout << "\nMasukkan Kode Produk yang akan dihapus: ";
    cin >> kode;
    bool ditemukan = false;
    for (int i = 0; i < jumlahData; i++) {
        if (kodeProduk[i] == kode) {
            for (int j = i; j < jumlahData - 1; j++) {
                kodeProduk[j] = kodeProduk[j + 1];
                namaProduk[j] = namaProduk[j + 1];
                merkProduk[j] = merkProduk[j + 1];
                hargaProduk[j] = hargaProduk[j + 1];
            }
            jumlahData--;
            cout << "Data berhasil dihapus!" << endl;
            ditemukan = true;
            break;
        }
    }
    if (!ditemukan) {
        cout << "Kode produk tidak ditemukan!" << endl;
    }
}
break;
```

## G. Konfirmasi Ulang

Fitur ini meminta pengguna yang telah menjalankan fitur untuk memilih apakah ingin melanjutkan program atau tidak (y/n).

Source Code:

```
    cout << "\nKembali ke menu? (y/n): ";
    cin >> ulang;
} while (ulang == 'y' || ulang == 'Y');

cout << "Program selesai!" << endl;
return 0;
}
```

## **4. Uji Coba dan Hasil Output**

### **4.1 Uji Coba**

#### **1. Skenario 1: Jika Login Berhasil**

Saat user memasukkan nama “lian” dan nim “001”, maka program akan berlanjut dan masuk ke menu berikutnya

#### **2. Skenario 2 : Jika Login Gagal 3 Kali**

Saat user memasukkan nama selain “lian” dan nim selain “001” sampai 3 kali, maka program akan berhenti karena kesempatan percobaan sudah habis

#### **3. Skenario 3 : Jika User Memilih Pilihan 1 Sebelum Menambah Data**

Program akan menampilkan output “Tidak ada data handphone!” karena belum ada data produk yang ditambahkan

#### **4. Skenario 4 : Jika User Memilih Pilihan 1 dan Sudah Menambah Data di Pilihan 2**

Program akan menampilkan data-data handphone yang sudah user tambahkan, setelah itu user akan diberi pilihan untuk melanjutkan program atau berhenti (y/n)

#### **5. Skenario 5 : Jika User Memilih Pilihan 2**

User dapat menambahkan kode, nama, merk dan harga produk yang kemudian akan disimpan oleh program selama user terus melanjutkan program, setelah itu user akan diberi pilihan untuk melanjutkan program atau berhenti (y/n)

#### **6. Skenario 6 : Jika User Memilih Pilihan 3**

User akan diminta untuk memasukkan kode produk untuk produk yang ingin diubah, jika kode produk yang dimasukkan ada maka user dapat mengubah nama, merk, dan harga pada produk itu, setelah itu user akan diberi pilihan untuk melanjutkan program atau berhenti (y/n)

7. Skenario 7 : Jika User Memilih Pilihan 4

User akan diminta untuk memasukkan kode produk untuk produk yang ingin dihapus, jika kode produk yang dimasukkan ada maka produk tersebut akan dihapus dari data produk, setelah itu user akan diberi pilihan untuk melanjutkan program atau berhenti (y/n)

8. Skenario 8 : Jika User Memilih Pilihan 5

Program akan berhenti saat user memilih pilihan 5

## 4.2 Hasil Output

```
=== Login Sistem ===  
Masukkan Nama: lian  
Masukkan NIM: 001  
Login berhasil!  
=====
```

| === Manajemen Produk Handphone === |  |
|------------------------------------|--|
| 1. Tampilkan Data                  |  |
| 2. Tambah Data                     |  |
| 3. Ubah Data                       |  |
| 4. Hapus Data                      |  |
| 5. Keluar                          |  |

```
=====
```

Pilih menu (1-5):

Gambar 4.1 Login Berhasil

```
=== Login Sistem ===  
Masukkan Nama: ad  
Masukkan NIM: 12  
Login gagal! Sisa percobaan: 2  
  
=== Login Sistem ===  
Masukkan Nama: dw  
Masukkan NIM: 33  
Login gagal! Sisa percobaan: 1  
  
=== Login Sistem ===  
Masukkan Nama: gf  
Masukkan NIM: 55  
Login gagal! Sisa percobaan: 0  
Cukurukuk gak bisa nyoba login lagi
```

Gambar 4.2 Login Gagal 3 Kali

```

=====
=== Manajemen Produk Handphone ===
=====
|1. Tampilkan Data      |
|2. Tambah Data         |
|3. Ubah Data           |
|4. Hapus Data          |
|5. Keluar              |
=====
Pilih menu (1-5): 1
Tidak ada data handphone!

Kembali ke menu? (y/n): 

```

Gambar 4.3 User Memilih Pilihan 1 Sebelum Menambah Data

```

=====
=== Manajemen Produk Handphone ===
=====
|1. Tampilkan Data      |
|2. Tambah Data         |
|3. Ubah Data           |
|4. Hapus Data          |
|5. Keluar              |
=====
Pilih menu (1-5): 1

=== Data Handphone ===
      Kode      Nama Produk      Merk      Harga
-----
      00      ipong 12 promex      apel      2000000
      01                      poco      xiami      4000000

Kembali ke menu? (y/n): 

```

Gambar 4.4 User Memilih Pilihan 1 dan Sudah Menambah Data di Pilihan 2

```
=====
=== Manajemen Produk Handphone ===
=====
|1. Tampilkan Data          |
|2. Tambah Data            |
|3. Ubah Data              |
|4. Hapus Data             |
|5. Keluar                 |
=====
Pilih menu (1-5): 2

=== Tambah Data Handphone ===
Masukkan Kode Produk: 02
Masukkan Nama Produk: redmi
Masukkan Merk Produk: xiami
Masukkan Harga: 1200000
Data berhasil ditambahkan!

Kembali ke menu? (y/n):
```

Gambar 4.5 User Memilih Pilihan 2

```
=====
=== Manajemen Produk Handphone ===
=====
|1. Tampilkan Data      |
|2. Tambah Data         |
|3. Ubah Data           |
|4. Hapus Data          |
|5. Keluar              |
=====
Pilih menu (1-5): 3

Masukkan Kode Produk yang akan diubah: 00
Masukkan Nama Produk baru: j2 prime
Masukkan Merk Produk: samsung
Masukkan Harga baru: 10000
Data berhasil diubah!

Kembali ke menu? (y/n): y
=====
=== Manajemen Produk Handphone ===
=====
|1. Tampilkan Data      |
|2. Tambah Data         |
|3. Ubah Data           |
|4. Hapus Data          |
|5. Keluar              |
=====
Pilih menu (1-5): 1

=== Data Handphone ===
      Kode      Nama Produk      Merk      Harga
-----
      00      j2 prime      amsung      10000
      01      poco      xiami      4000000
      02      redmi      xiami      1200000

Kembali ke menu? (y/n):
```

Gambar 4.6 User Memilih Pilihan 3

```
=====
=== Manajemen Produk Handphone ===
=====
|1. Tampilkan Data      |
|2. Tambah Data         |
|3. Ubah Data           |
|4. Hapus Data          |
|5. Keluar              |
=====
Pilih menu (1-5): 4

Masukkan Kode Produk yang akan dihapus: 02
Data berhasil dihapus!

Kembali ke menu? (y/n): y
=====
=== Manajemen Produk Handphone ===
=====
|1. Tampilkan Data      |
|2. Tambah Data         |
|3. Ubah Data           |
|4. Hapus Data          |
|5. Keluar              |
=====
Pilih menu (1-5): 1

=== Data Handphone ===
      Kode      Nama Produk      Merk      Harga
-----
      00      j2 prime      amsung      10000
      01      poco      xiami      4000000

Kembali ke menu? (y/n):
```

Gambar 4.7 User Memilih Pilihan 4

```
=====
=== Manajemen Produk Handphone ===
=====
|1. Tampilkan Data      |
|2. Tambah Data         |
|3. Ubah Data           |
|4. Hapus Data          |
|5. Keluar              |
=====
Pilih menu (1-5): 5
Terima kasih!
```

Gambar 4.8 User Memilih Pilihan 5



## 5. Git

### 5.1 Git Init

Git init merupakan command yang berfungsi untuk menginisiasi repository yang ada pada file lokal dan berlokasi di folder .git

```
ASUS-GK@DESKTOP-M11TQOT MINGW64 ~/documents/github/praktikum-apl (main)
$ git init
Reinitialized existing Git repository in C:/Users/ASUS-GK/Documents/GitHub/praktikum-apl/.git/
```

Gambar 5.1 Git Init

### 5.2 Git Add

Menambahkan file yang ingin di commit.

```
ASUS-GK@DESKTOP-M11TQOT MINGW64 ~/documents/github/praktikum-apl/post-test/post-test-apl-2 (main)
$ git add 2409106001_DwikiAprilianAryanda-PT-2.cpp
```

Gambar 5.2 Git Add

### 5.3 Git Commit

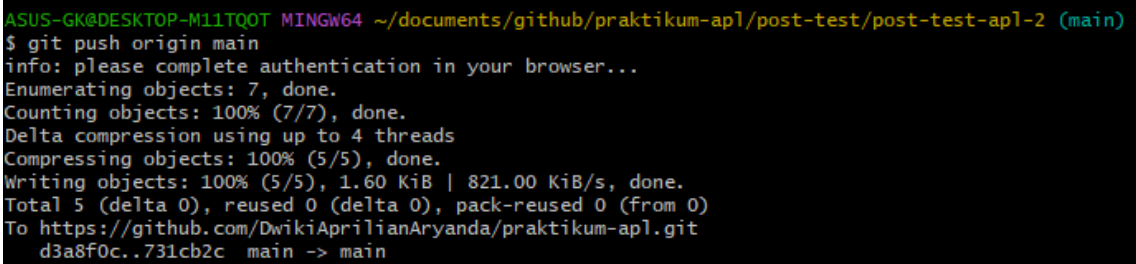
Menyimpan perubahan yang disimpan pada folder lokal dengan sebuah pesan.

```
ASUS-GK@DESKTOP-M11TQOT MINGW64 ~/documents/github/praktikum-apl/post-test/post-test-apl-2 (main)
$ git commit -m "Program done"
[main 731cb2c] Program done
1 file changed, 179 insertions(+)
create mode 100644 post-test/post-test-apl-2/2409106001_DwikiAprilianAryanda-PT-2.cpp
```

Gambar 5.3 Git Commit

## 5.4 Git Push

Git push berfungsi untuk mengupload semua yang ada di lokal ke github.

A terminal window with a black background and green text. The prompt is 'ASUS-GK@DESKTOP-M11TQOT MINGW64 ~/documents/github/praktikum-apl/post-test/post-test-apl-2 (main)'. The command '\$ git push origin main' has been entered. The output shows a series of status messages: 'info: please complete authentication in your browser...', 'Enumerating objects: 7, done.', 'Counting objects: 100% (7/7), done.', 'Delta compression using up to 4 threads', 'Compressing objects: 100% (5/5), done.', 'Writing objects: 100% (5/5), 1.60 KiB | 821.00 KiB/s, done.', 'Total 5 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)', and 'To https://github.com/DwikiAprilianAryanda/praktikum-apl.git'. The final line shows the commit hash 'd3a8f0c..731cb2c' and the branch transition 'main -> main'.

```
ASUS-GK@DESKTOP-M11TQOT MINGW64 ~/documents/github/praktikum-apl/post-test/post-test-apl-2 (main)
$ git push origin main
info: please complete authentication in your browser...
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 4 threads
Compressing objects: 100% (5/5), done.
Writing objects: 100% (5/5), 1.60 KiB | 821.00 KiB/s, done.
Total 5 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/DwikiAprilianAryanda/praktikum-apl.git
d3a8f0c..731cb2c main -> main
```

Gambar 5.4 Git Push

