

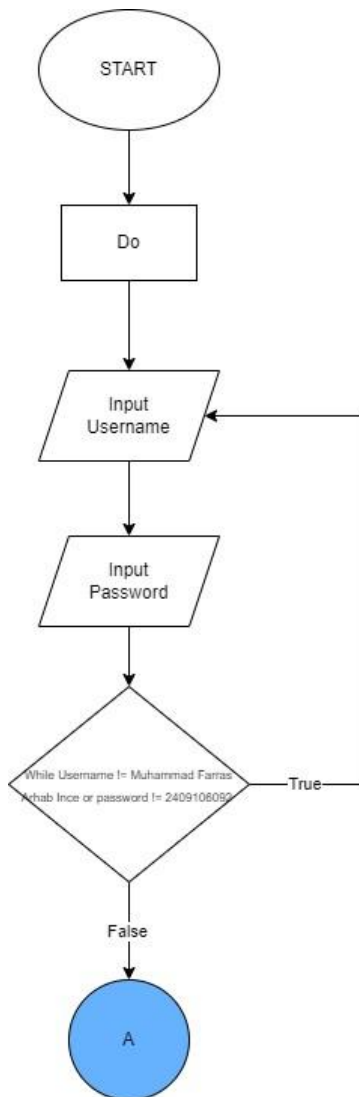
**LAPORAN PRAKTIKUM**  
**POSTTEST I**  
**ALGORITMA PEMROGRAMAN LANJUT**



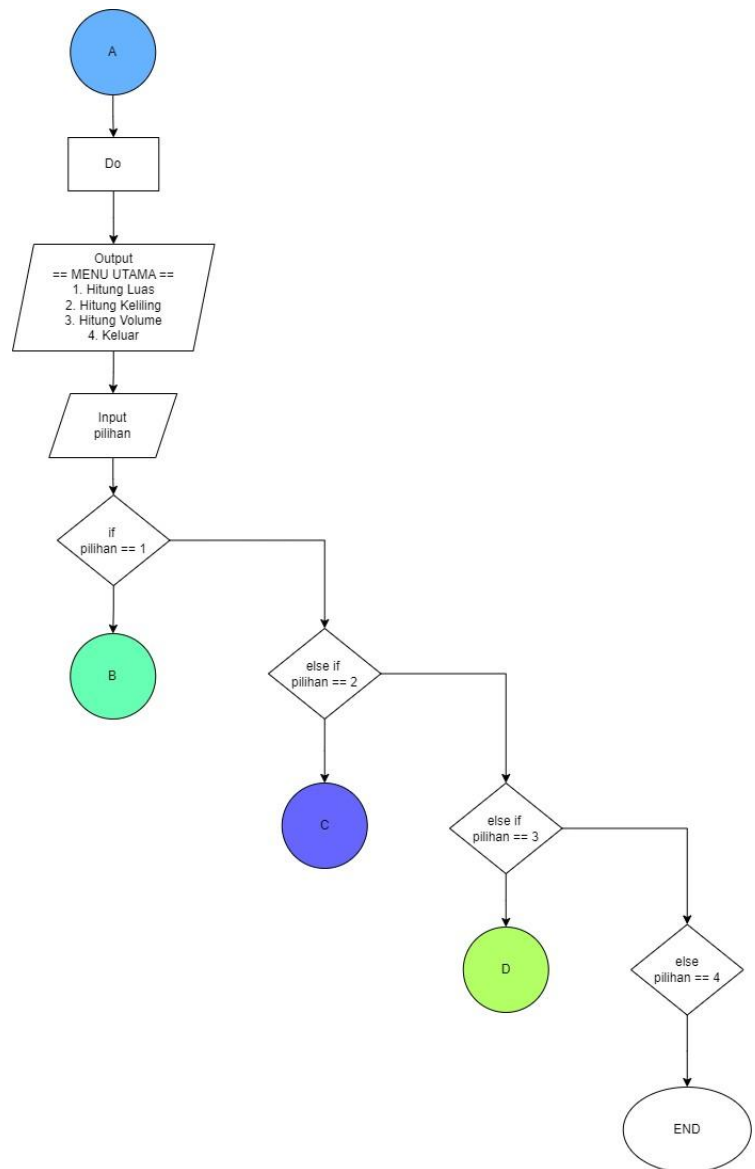
**Disusun oleh:**  
**Muhammad Farras Arhab Ince (2409106092)**  
**Kelas (C1 '24)**

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

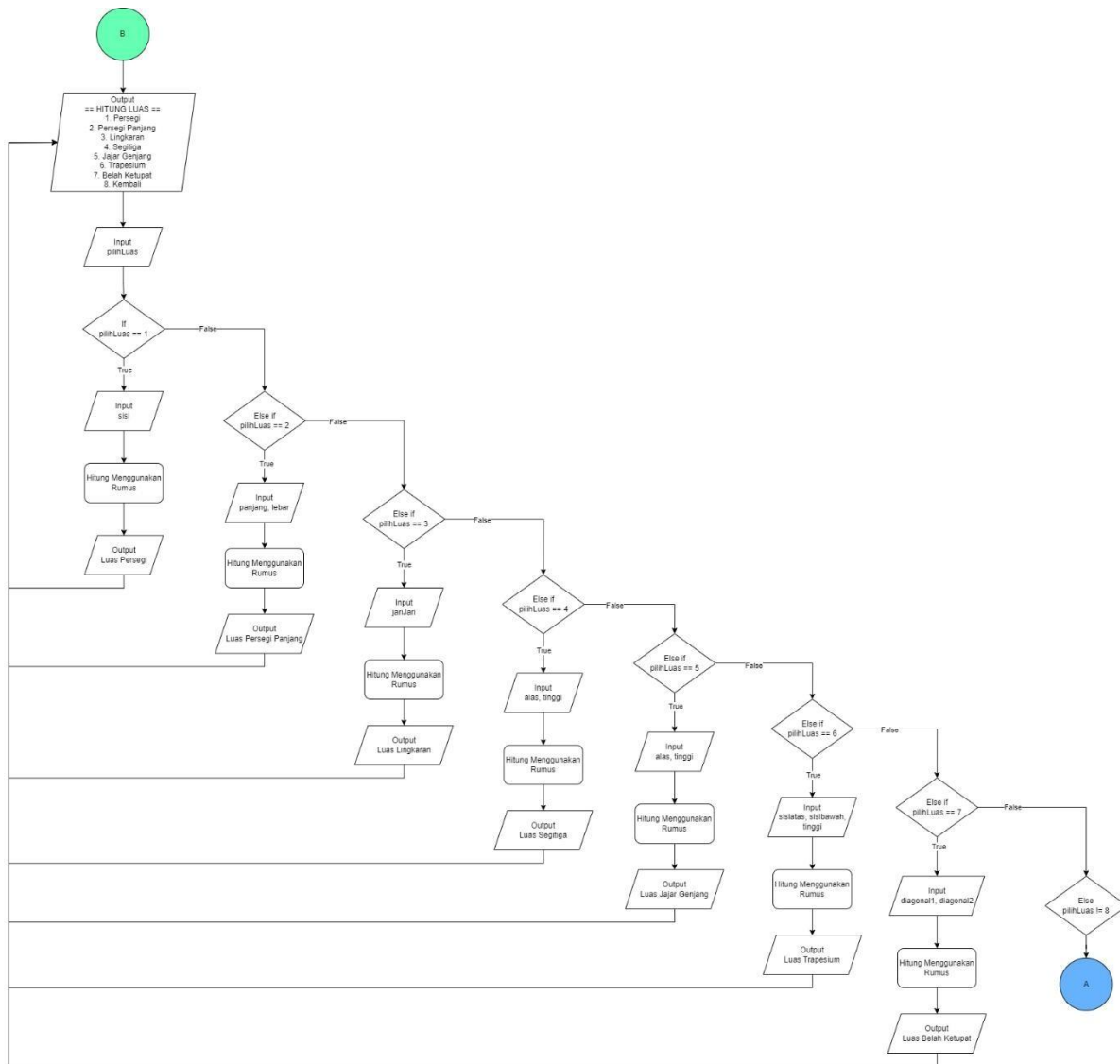
## 1. Flowchart



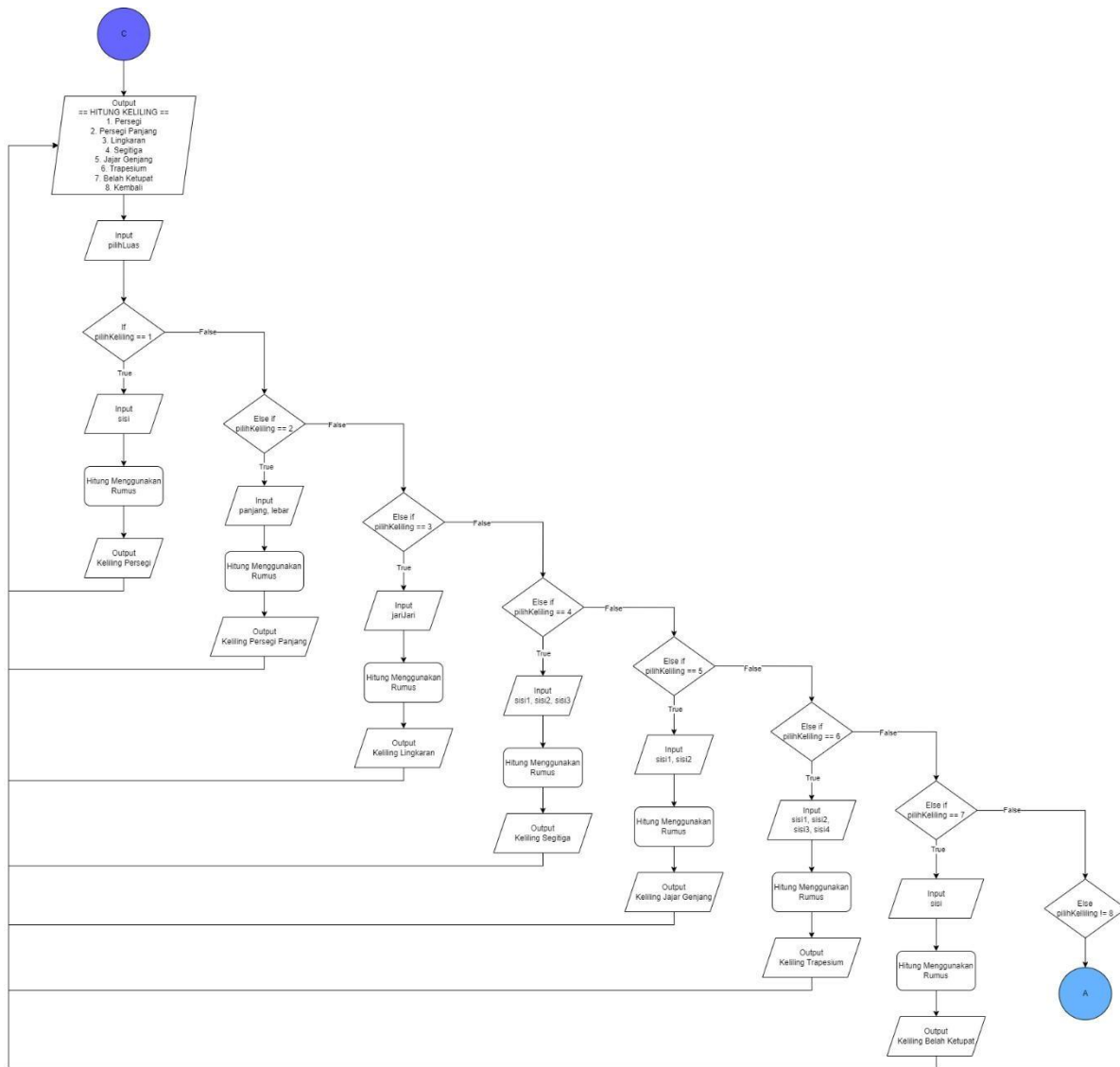
Gambar 1.1 Flowchart Sistem Login



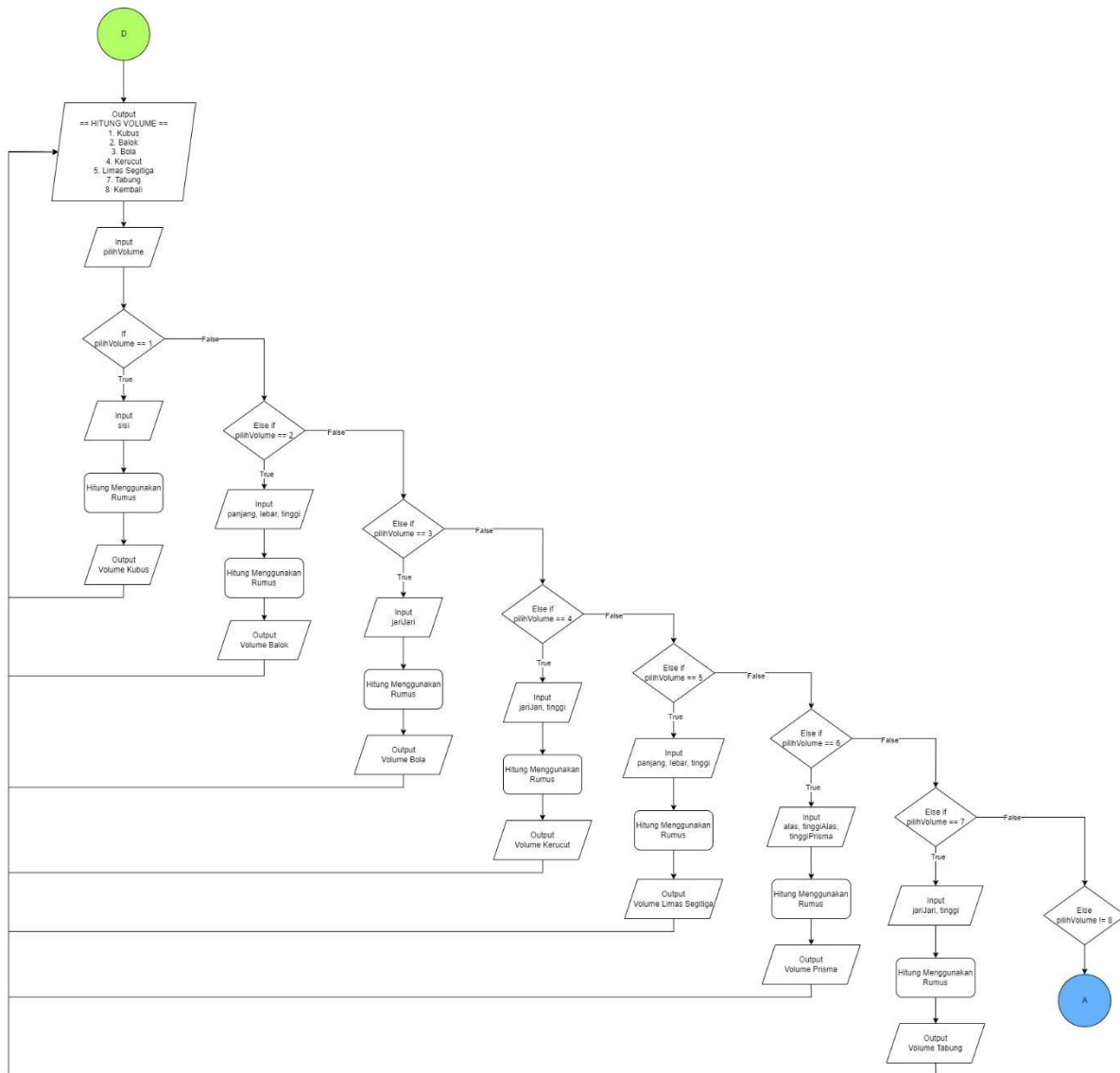
Gambar 1.2 Flowchart Menu Utama



Gambar 1.3 Flowchart Menu Hitung Luas



Gambar 1.4 Flowchart Menu Hitung Keliling



Gambar 1.5 Flowchart Menu Hitung Volume

## 2. Analisis Program

### 2.1 Deskripsi Singkat Program

Program ini merupakan aplikasi berbasis menu yang memungkinkan pengguna untuk menghitung luas, keliling, dan volume berbagai bangun datar serta bangun ruang. Pengguna harus melakukan login terlebih dahulu sebelum mengakses menu utama. Program ini menyediakan berbagai pilihan perhitungan, termasuk luas dan keliling persegi, persegi panjang, lingkaran, segitiga, serta volume kubus, balok, bola, dan lainnya. Program ini dirancang dengan struktur *do-while* agar pengguna dapat terus melakukan perhitungan tanpa harus menjalankan ulang program secara manual.

### 2.2 Penjelasan Alur & Algoritma

Program ini merupakan aplikasi berbasis menu yang memungkinkan pengguna untuk menghitung luas, keliling, dan volume berbagai bangun datar serta bangun ruang. Berikut adalah alur kerja program secara berurutan:

#### 1. Login Pengguna

Program pertama-tama meminta pengguna untuk memasukkan username dan password. Jika username atau password salah, pengguna diminta untuk mengulangnya sampai berhasil login. Jika login berhasil, program menampilkan pesan selamat datang dan melanjutkan ke menu utama.

#### 2. Menu Utama:

Setelah login berhasil, pengguna diberikan beberapa pilihan utama:

- 1. Hitung Luas
- 2. Hitung Keliling
- 3. Hitung Volume
- 4. Keluar dari program

Program akan terus menampilkan menu ini hingga pengguna memilih opsi 4 untuk keluar.

#### 3. Menu Hitung Luas

Jika pengguna memilih opsi 1, program akan menampilkan submenu untuk memilih jenis bangun datar yang ingin dihitung luasnya. Setiap jenis bangun datar memiliki rumus perhitungan tersendiri, misalnya:

- **Persegi:**  $\text{sisi} \times \text{sisi}$
- **Lingkaran:**  $\pi \times r^2$
- **Segitiga:**  $\frac{1}{2} \times \text{alas} \times \text{tinggi}$

Setelah perhitungan selesai, hasil ditampilkan, dan pengguna dikembalikan ke submenu "Hitung Luas" hingga mereka memilih untuk kembali ke menu utama.

#### 4. Menu Hitung Keliling

Jika pengguna memilih opsi 2, program akan menampilkan submenu untuk memilih jenis bangun datar yang ingin dihitung kelilingnya. Setiap jenis bangun datar memiliki rumus keliling tersendiri, seperti:

- **Persegi:**  $4 \times \text{sisi}$
- **Lingkaran:**  $2 \times \pi \times r$
- **Segitiga:** jumlah panjang tiga sisi

Setelah perhitungan selesai, hasil ditampilkan, dan pengguna dikembalikan ke submenu "Hitung Keliling" hingga mereka memilih untuk kembali ke menu utama.

#### 5. Menu Hitung Volume

Jika pengguna memilih opsi 3, program akan menampilkan submenu untuk memilih jenis bangun ruang yang ingin dihitung volumenya. Setiap jenis bangun ruang memiliki rumus volume tersendiri, misalnya:

- **Kubus:**  $\text{sisi}^3$
- **Balok:**  $\text{panjang} \times \text{lebar} \times \text{tinggi}$
- **Bola:**  $(4/3) \times \pi \times r^3$

Setelah perhitungan selesai, hasil ditampilkan, dan pengguna dikembalikan ke submenu "Hitung Volume" hingga mereka memilih untuk kembali ke menu utama.

#### 6. Keluar Dari Program

Jika pengguna memilih opsi 4 pada menu utama, program akan berhenti dan keluar.

### 3. Source Code

#### A. Login

Fitur ini memastikan hanya pengguna yang diizinkan dapat mengakses program dengan memverifikasi username dan password yang dimasukkan. Jika salah, pengguna harus mencoba lagi hingga login berhasil.

#### Source Code:

```
int main() {
    string username, password;
    do {
        cout << "== SILAHKAN MASUK ==\n"
              << "Masukkan username: ";
        getline(cin, username);
        cout << "Masukkan password: ";
        cin >> password;
        cin.ignore();

        if (username != "Muhammad Farras Arhab Ince" || password !=
            "2409106092") {
            cout << "Login gagal! Coba lagi.\n";
        }
    } while (true);
}
```

```

    }
    } while (username != "Muhammad Farras Arhab Ince" || password !=
"2409106092");

    cout << "Login berhasil! Selamat datang, " << username << "!\n";

```

## B. Menu

Bagian ini menyediakan navigasi utama bagi pengguna untuk memilih fitur yang tersedia, yaitu menghitung luas, keliling, atau volume berbagai bangun.

### Source Code:

```

int pilihan;
do {
    cout << "\n== MENU UTAMA ==\n"
        << "1. Hitung Luas\n"
        << "2. Hitung Keliling\n"
        << "3. Hitung Volume\n"
        << "4. Keluar\n"
        << "Pilih menu: ";
    cin >> pilihan;
}

```

## C. Perhitungan Luas

Program menghitung luas beberapa bangun datar, seperti persegi dan lingkaran, berdasarkan input pengguna. Rumus luas diterapkan langsung dalam perhitungan.

### Source Code:

```

double sisi;
cout << "Masukkan panjang sisi: ";
cin >> sisi;
cout << "Luas Persegi: " << sisi * sisi << endl;

double panjang, lebar;
cout << "Masukkan panjang: ";
cin >> panjang;
cout << "Masukkan lebar: ";
cin >> lebar;
cout << "Luas Persegi Panjang: " << panjang * lebar << endl;

```

## D. Perhitungan Keliling

Seperti perhitungan luas, fitur ini menghitung keliling bangun datar seperti persegi dan lingkaran dengan rumus yang sesuai.

### Source Code:

```

double jariJari;
cout << "Masukkan jari-jari: ";
cin >> jariJari;

```



```

cout << "Luas Lingkaran: " << 3.14 * jariJari * jariJari << endl;

double alas, tinggi;
cout << "Masukkan panjang alas: ";
cin >> alas;
cout << "Masukkan tinggi: ";
cin >> tinggi;
cout << "Luas Segitiga: " << 0.5 * alas * tinggi << endl;

```

## E. Perhitungan Volume

Fitur ini menghitung volume bangun ruang, seperti kubus dan bola, menggunakan input dari pengguna dan menerapkan rumus volume yang sesuai.

### Source Code:

```

double sisi1, sisi2;
cout << "Masukkan panjang sisi pertama: ";
cin >> sisi1;
cout << "Masukkan Panjang sisi kedua: ";
cin >> sisi2;
cout << "Keliling Jajar Genjang: " << 2 * (sisi1 + sisi2) << endl;

double sisi1, sisi2, sisi3, sisi4;
cout << "Masukkan panjang sisi pertama: ";
cin >> sisi1;
cout << "Masukkan panjang sisi kedua: ";
cin >> sisi2;
cout << "Masukkan panjang sisi ketiga: ";
cin >> sisi3;
cout << "Masukkan panjang sisi keempat: ";
cin >> sisi4;
cout << "Keliling Trapesium: " << sisi1 + sisi2 + sisi3 + sisi4 << endl;

```

## 4. Uji Coba dan Hasil Output

### 4.1 Uji Coba

#### 1. Login Gagal

Seorang pengguna membuka aplikasi dan memasukkan username serta password, tetapi memasukkan nama yang salah. Sistem segera merespons dengan menampilkan pesan **"Login gagal! Coba lagi."**.

#### 2. Login Berhasil

Pengguna mencoba masuk kembali, tetapi kali ini dengan benar. Setelah menekan tombol enter, sistem memverifikasi data dan menampilkan pesan "Login berhasil! Selamat datang, Muhammad Farras Arhab Ince!". Pengguna kini dapat mengakses fitur yang tersedia di dalam aplikasi.

#### 3. Menghitung Luas Persegi

Setelah berhasil login, pengguna memilih menu Hitung Luas. Dari daftar bangun datar yang tersedia, pengguna memilih Persegi, lalu memasukkan panjang sisi 2. Sistem kemudian menghitung luasnya dan menampilkan hasil "Luas Persegi: 4", yang sesuai dengan rumus luas persegi ( $sisi \times sisi$ ).

#### 4. Menghitung Keliling Persegi

Pengguna ingin menghitung keliling bangun datar, maka ia memilih menu Hitung Keliling. Dari daftar pilihan, ia kembali memilih Persegi dan memasukkan panjang sisi 4. Sistem segera menampilkan hasil "Keliling Persegi: 16", yang sesuai dengan rumus keliling ( $4 \times sisi$ ).

#### 5. Menghitung Volume Balok

Selanjutnya, pengguna ingin menghitung volume bangun ruang. Ia memilih menu Hitung Volume, lalu memilih Balok. Sistem meminta pengguna memasukkan panjang 6, lebar 7, dan tinggi 2. Setelah perhitungan selesai, sistem menampilkan "Volume Balok: 84", yang sesuai dengan rumus volume balok ( $panjang \times lebar \times tinggi$ ).

#### 6. Menangani Input Tidak Valid

Pengguna mencoba memasukkan angka negatif saat memilih bangun ruang untuk dihitung volumenya. Sistem mendeteksi input yang tidak valid dan menampilkan kembali daftar pilihan tanpa memproses perhitungan. Hal ini menunjukkan bahwa sistem telah dilengkapi dengan validasi input yang mencegah kesalahan dalam pengoperasian.

## 4.2 Hasil Output

### 1. Login Gagal

```
== SILAHKAN MASUK ==  
Masukkan username: Muhammad Farras  
Masukkan password: 2409106092  
Login gagal! Coba lagi.
```

Gambar 4.2.1 Pengujian Login Gagal

### 2. Login Berhasil

```
== SILAHKAN MASUK ==  
Masukkan username: Muhammad Farras Arhab Ince  
Masukkan password: 2409106092  
Login berhasil! Selamat datang, Muhammad Farras Arhab Ince!
```

Gambar 4.2.2 Pengujian Login Berhasil

### 3. Menghitung Luas Persegi

```
== MENU UTAMA ==  
1. Hitung Luas  
2. Hitung Keliling  
3. Hitung Volume  
4. Keluar  
Pilih menu: 1  
  
== HITUNG LUAS ==  
1. Persegi  
2. Persegi Panjang  
3. Lingkaran  
4. Segitiga  
5. Jajar Genjang  
6. Trapesium  
7. Belah Ketupat  
8. Kembali  
Pilih bangun datar: 1  
Masukkan panjang sisi: 2  
Luas Persegi: 4
```

Gambar 4.2.3 Pengujian Menghitung Luas Persegi

#### 4. Menghitung Keliling Persegi

```
== MENU UTAMA ==
1. Hitung Luas
2. Hitung Keliling
3. Hitung Volume
4. Keluar
Pilih menu: 2

== HITUNG KELILING ==
1. Persegi
2. Persegi Panjang
3. Lingkaran
4. Segitiga
5. Jajar Genjang
6. Trapesium
7. Belah Ketupat
8. Kembali
Pilih bangun datar: 1
Masukkan panjang sisi: 4
Keliling Persegi: 16
```

Gambar 4.2.4 Pengujian Menghitung Keliling Persegi

#### 5. Menghitung Volume Balok

```
== MENU UTAMA ==
1. Hitung Luas
2. Hitung Keliling
3. Hitung Volume
4. Keluar
Pilih menu: 3

== HITUNG VOLUME ==
1. Kubus
2. Balok
3. Bola
4. Kerucut
5. Limas Segiempat
6. Prisma Segitiga
7. Tabung
8. Kembali
Pilih bangun ruang: 2
Masukkan panjang: 6
Masukkan lebar: 7
Masukkan tinggi: 2
Volume Balok: 84
```

Gambar 4.2.5 Pengujian Menghitung Volume Balok

## 6. Menangani Input Tidak Valid

```
== HITUNG VOLUME ==
1. Kubus
2. Balok
3. Bola
4. Kerucut
5. Limas Segiempat
6. Prisma Segitiga
7. Tabung
8. Kembali
Pilih bangun ruang: -1

== HITUNG VOLUME ==
1. Kubus
2. Balok
3. Bola
4. Kerucut
5. Limas Segiempat
6. Prisma Segitiga
7. Tabung
8. Kembali
Pilih bangun ruang: |
```

Gambar 4.2.6 Pengujian Input Tidak Valid

## 5. Git

```
MINGW64:/d/Kuliah/SEMESTER 2/PRAKTIKUM APL/praktikum-apl
farra@LAPTOP-SQ4287S4 MINGW64 /d/Kuliah/SEMESTER 2/PRAKTIKUM APL/praktikum-apl
$ git config --global user.email "farrasarhab06@gmail.com"

farra@LAPTOP-SQ4287S4 MINGW64 /d/Kuliah/SEMESTER 2/PRAKTIKUM APL/praktikum-apl
$ git init
Initialized empty Git repository in D:/Kuliah/SEMESTER 2/PRAKTIKUM APL/praktikum-apl/.git/

farra@LAPTOP-SQ4287S4 MINGW64 /d/Kuliah/SEMESTER 2/PRAKTIKUM APL/praktikum-apl (master)
$ git add .

farra@LAPTOP-SQ4287S4 MINGW64 /d/Kuliah/SEMESTER 2/PRAKTIKUM APL/praktikum-apl (master)
$ git branch -M main

farra@LAPTOP-SQ4287S4 MINGW64 /d/Kuliah/SEMESTER 2/PRAKTIKUM APL/praktikum-apl (main)
$

git remote add origin https://github.com/parashrb/praktikum-apl.git

farra@LAPTOP-SQ4287S4 MINGW64 /d/Kuliah/SEMESTER 2/PRAKTIKUM APL/praktikum-apl (main)
$ git commit -m "Update Paras Keren"
[main (root-commit) 74e7b02] Update Paras Keren
1 file changed, 0 insertions(+), 0 deletions(-)
create mode 100644 Post-Test/Post-Test-1/2409106092-MuhammadFarrasArhabInce-PT-1.cpp

farra@LAPTOP-SQ4287S4 MINGW64 /d/Kuliah/SEMESTER 2/PRAKTIKUM APL/praktikum-apl (main)
$ git push -u origin main
info: please complete authentication in your browser...
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 12 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (5/5), 347 bytes | 347.00 KiB/s, done.
Total 5 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/parashrb/praktikum-apl.git
 * [new branch] main -> main
branch 'main' set up to track 'origin/main'.

farra@LAPTOP-SQ4287S4 MINGW64 /d/Kuliah/SEMESTER 2/PRAKTIKUM APL/praktikum-apl (main)
$

farra@LAPTOP-SQ4287S4 MINGW64 /d/Kuliah/SEMESTER 2/PRAKTIKUM APL/praktikum-apl (main)
$
```

Gambar 5 Terminal Git Bash

### 1. Mengatur Email Global untuk Git

```
$ git config --global user.email "farrasarhab06@gmail.com"
```

Perintah ini digunakan untuk mengatur email global Git, yang akan digunakan dalam setiap commit yang dibuat oleh pengguna. Email ini dikaitkan dengan identitas pengguna dalam repository Git.

### 2. Menginisialisasi Repository Git

```
$ git init
```

Perintah ini menginisialisasi (membuat) repository Git dalam folder Praktikum-Apl. Jika sudah ada repository Git di folder tersebut, maka Git hanya akan mengingatkan bahwa repository sudah ada (reinitialized).

### 3. Menambahkan File ke Staging Area

```
$ git add .
```

Perintah ini menambahkan semua file yang ada di dalam folder ke staging area.

Staging area adalah tempat sementara sebelum file dikomit ke dalam repository.

4. Menambahkan Remote Repository (Gagal karena Sudah Ada)

```
$ git remote add origin https://github.com/parashrb/parashrb-praktikum-apl.git
```

Perintah ini digunakan untuk menambahkan repository remote dengan nama origin.

Error: "remote origin already exists", ini terjadi karena sebelumnya sudah ada repository remote yang dikaitkan dengan nama origin.

5. Membuat Commit dengan Pesan "update paras keren"

```
$ git commit -m "update paras keren"
```

Perintah ini menyimpan perubahan dalam repository dengan commit dan pesan "update paras keren".

File yang dikomit:

- o Post-test/Post-test-1/2409106092-MuhammadFarrasArhabInce-PT-1.cpp
- o Post-test/Post-test-1/2409106092- MuhammadFarrasArhabInce -PT-1.exe

6. Mendorong (Push) Perubahan ke Repository Remote

```
$ git push -u origin main
```

Perintah ini mengunggah (push) perubahan ke repository remote pada branch main.

Karena ini adalah push pertama, flag -u digunakan untuk mengatur branch lokal main agar terhubung dengan branch main di remote repository.

Proses yang terjadi:

- o Menghitung objek (Enumerating objects: 6).
- o Mengompresi objek sebelum mengunggahnya.
- o Menulis (mengunggah) objek ke GitHub.
- o Menampilkan informasi bahwa branch main sekarang dilacak oleh remote repository origin/main.