

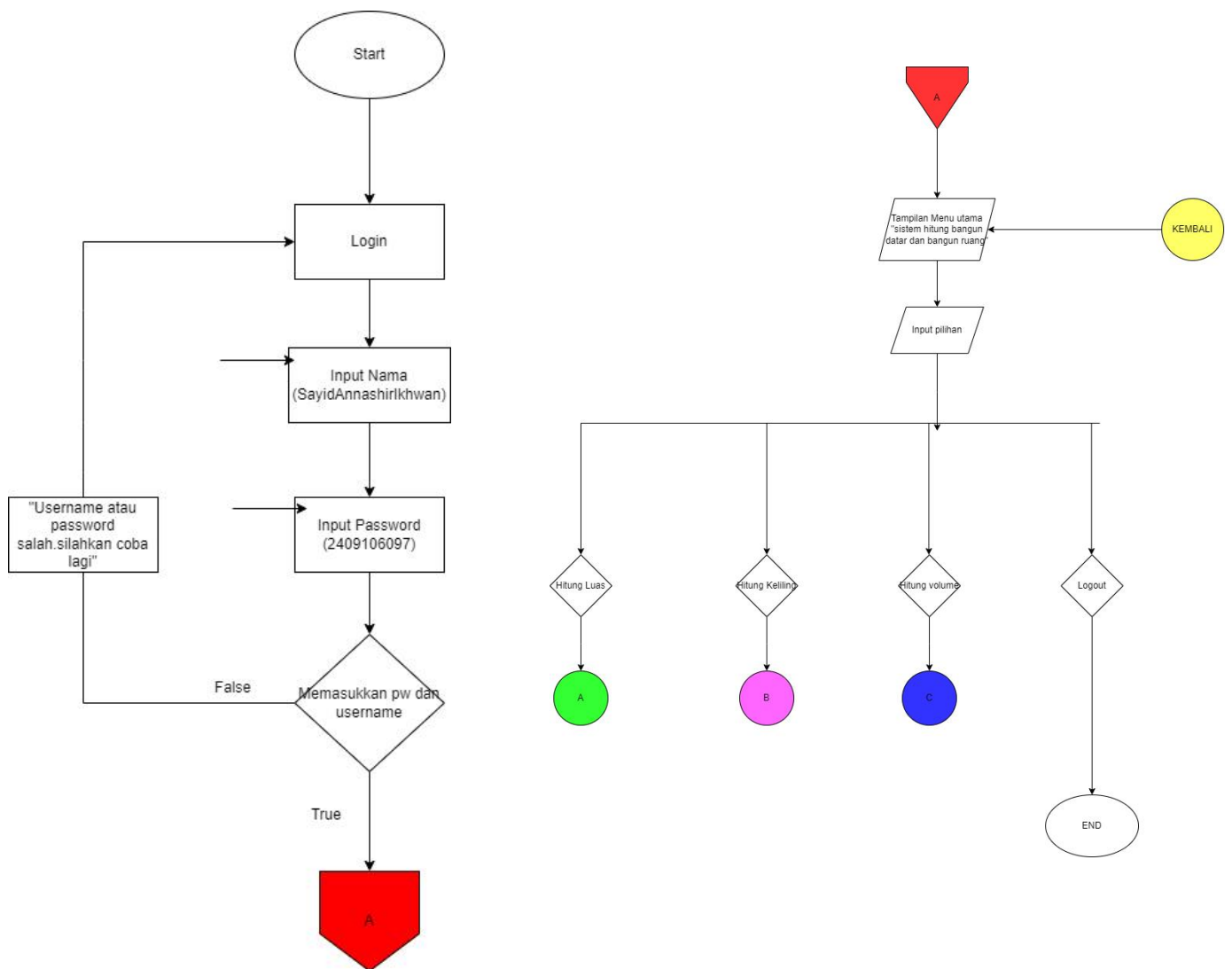
**LAPORAN PRAKTIKUM**  
**POSTTEST 1**  
**ALGORITMA PEMROGRAMAN LANJUT**



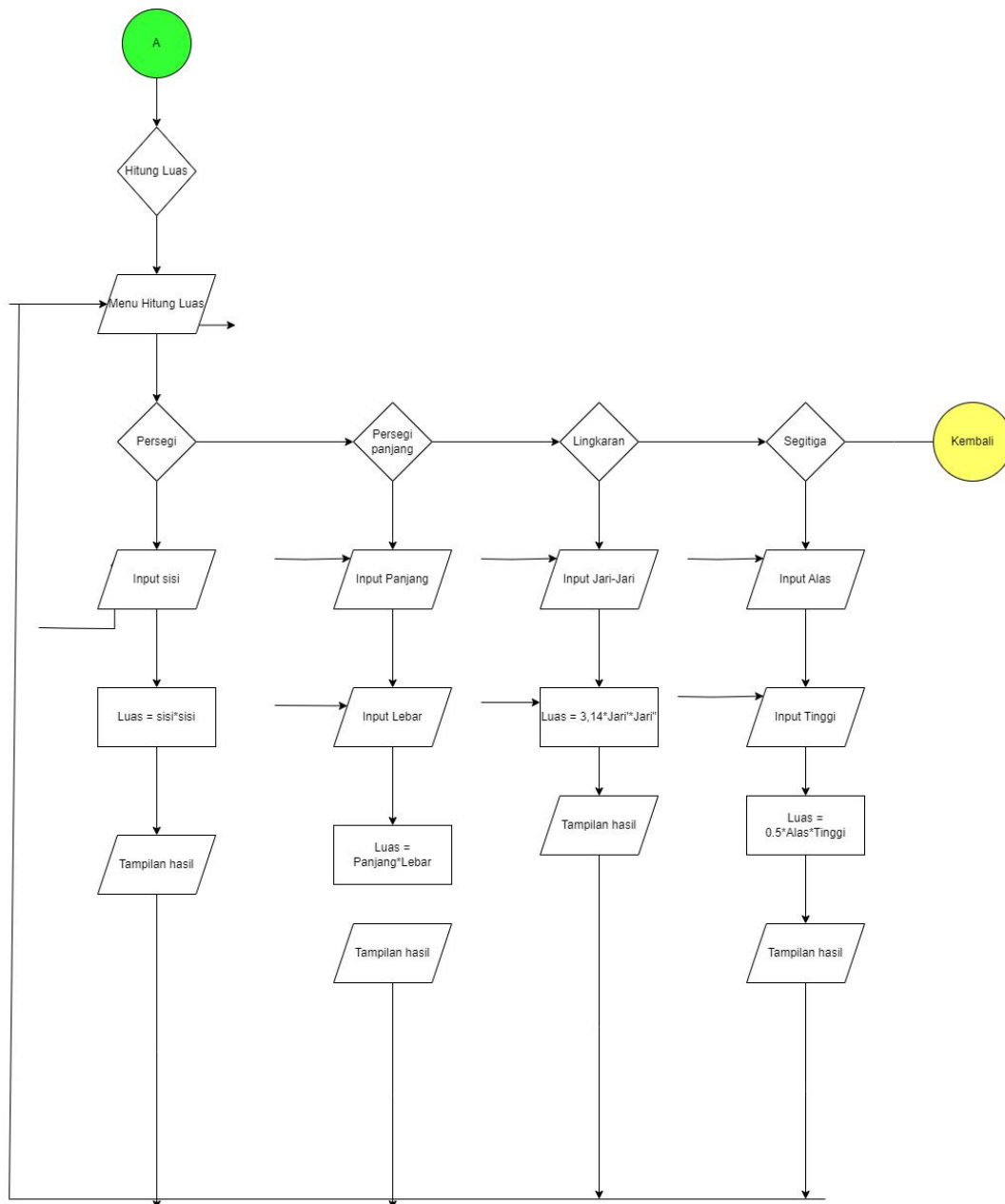
**Disusun oleh:**  
**Sayid Annashir Ikhwan (2409106097)**  
**Kelas (C '24)**

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

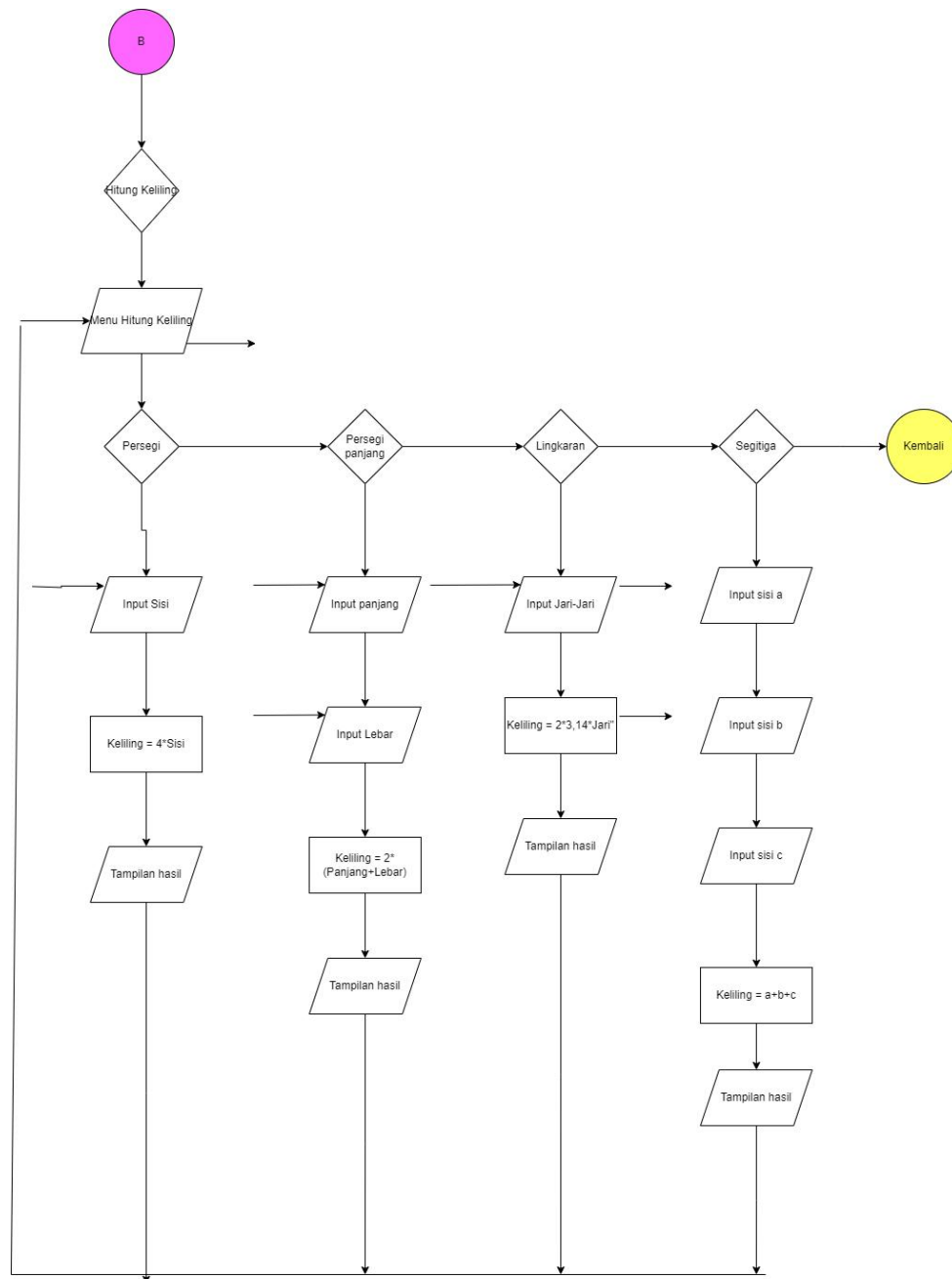
## 1. Flowchart



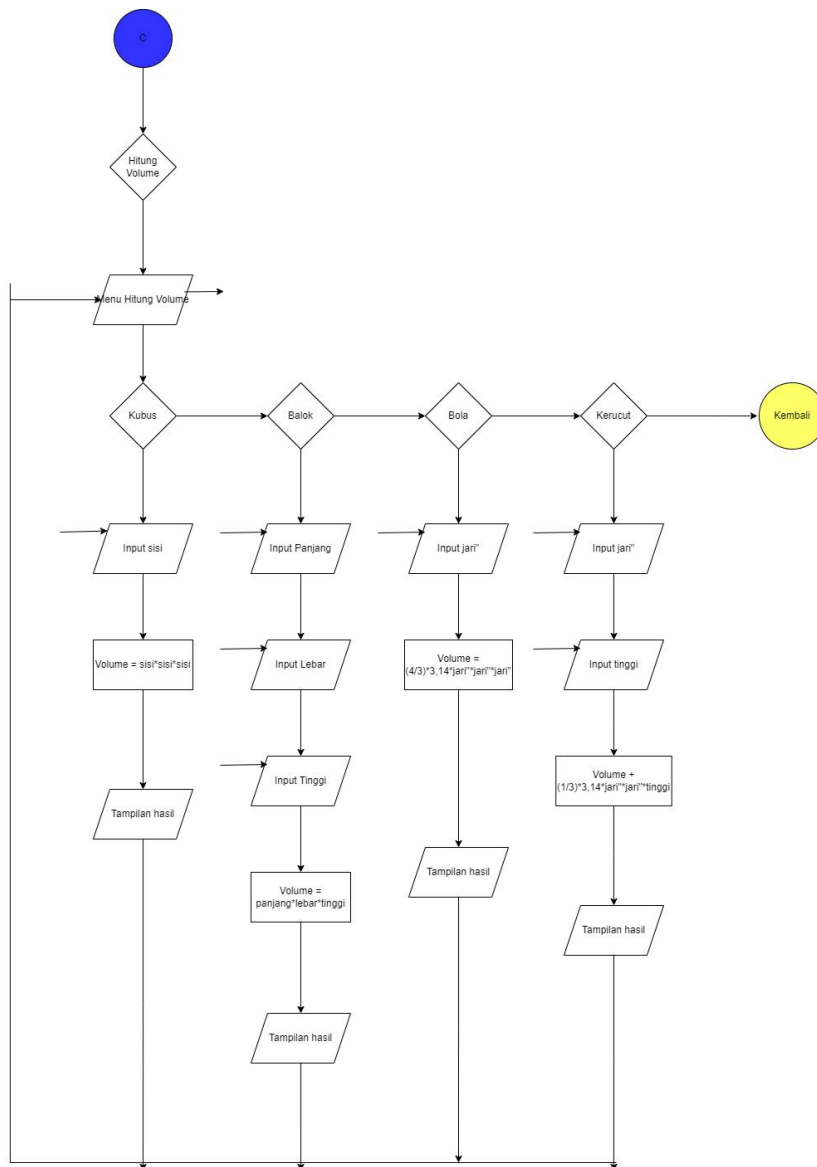
Gambar 1.1. Flowchart login,menu utama



Gambar 1.2. <flowchart hitung luas>



Gambar 1.3. <flowchart hitung keliling>



Gambar 1.4. <flowchart hitung volume>

## 2. Analisis Program

### 2.1 Deskripsi Singkat Program

Program ini dirancang untuk menghitung berbagai parameter geometri, seperti luas, keliling, dan volume dari bangun datar dan bangun ruang. Program dilengkapi dengan sistem login yang membatasi akses hanya kepada pengguna benar. Setelah login, pengguna dapat memilih berbagai operasi perhitungan hingga akhirnya keluar dari program.

### 2.2 Penjelasan Alur & Algoritma

- **Login:** Pengguna harus memasukkan username (SayidAnnashirIkhwan) dan password (2409106097) agar bisa masuk. jika salah maka program tersebut akan berhenti

- Jika username dan password benar, program masuk ke menu utama.
- Jika salah, program akan menampilkan pesan error

- **Menu Utama:** Pengguna dapat memilih

- Hitung luas
- Hitung keliling
- Hitung volume
- Logout

- **Submenu**

- hitung luas: Pengguna memilih bangun datar (persegi, persegi panjang, lingkaran, segitiga), lalu memasukkan input yang diperlukan. Program menghitung luas dan menampilkan hasilnya.
- Hitung keliling: Pengguna memilih bangun datar (persegi, persegi panjang, lingkaran, segitiga), lalu memasukkan input yang diperlukan. Program menghitung keliling dan menampilkan hasilnya

- Hitung Volume: Pengguna memilih bangun ruang (kubus,balok,bola,prisma segitiga), lalu memasukkan input yang diperlukan.Program menghitung volume dan menampilkan hasilnya

#### - Logout:

Yang terakhir yaitu logout.Jika pengguna memilih Logout, program akan berhenti.

### 3. Source Code

#### A. Fitur login.

Program akan dimulai dengan login. memasukkan username dan password yang harus sesuai dengan yang ada di database yaitu “Sayid Annashir Ikhwan” untuk Username dan “2409106097” untuk Passwordnya. Jika user menginput yang tidak sesuai dengan keterangan maka program akan terhenti dikarenakan tidak sesuai input login nya.

#### Source Code :

```
// Login
while (!loggedIn) {
    cout << "Selamat datang...silahkan login ><" << endl;
    cout << "Username: ";
    cin >> username;
    cout << "Password: ";
    cin >> password;
    if (username == "Sayid Annashir Ikhwan" && password == "2409106097") { //memasukkan nim dan
nama saya
        loggedIn = true;
        cout << "Login berhasil!" << endl;
    } else {
        cout << "Username atau passwordkamu salah. Silakan coba lagi." << endl;
    }
}
```

Gambar 3,1

#### B. Fitur Menu Utama

bagian pertama yang muncul setelah pengguna berhasil login. Di sini, pengguna diberikan beberapa pilihan untuk melakukan perhitungan atau keluar dari program. Pilihan yang tersedia adalah:

Hitung Luas: Untuk menghitung luas bangun datar.

Hitung Keliling: Untuk menghitung keliling bangun datar.

Hitung Volume: Untuk menghitung volume bangun ruang.

Logout: Untuk keluar dari program.

Dan juga pengguna bisa memilih opsi “5” untuk kembali ke menu utama, pada saat ingin mengubah dari hitung luas ke hitung keliling, begitu juga dengan hitung volume.

```
// Main Menu
while (loggedIn) {
    cout << "\n== SISTEM HITUNG BANGUN DATAR DAN BANGUN RUANG ==> << endl;
    cout << "1. Hitung Luas" << endl;
    cout << "2. Hitung Keliling" << endl;
    cout << "3. Hitung Volume" << endl;
    cout << "4. Logout" << endl;
    cout << "Pilih menu: ";
    int mainMenuChoice;
    cin >> mainMenuChoice;
```

Gambar 3,2

### C. Fitur Sistem hitung bangun datar dan bangun ruang

Fitur ini digunakan untuk menghitung luas bangun datar dan bangun ruang yang dipilih oleh pengguna.

#### 1. Hitung Luas

- Persegi

Rumus:  $\text{Luas} = \text{sisi} \times \text{sisi}$

Jika pengguna memilih opsi 1 (Persegi), program akan meminta input sisi dan menghitung luasnya.

```
if (luasChoice == 1) {
    double sisi;
    cout << "Masukkan panjang sisi persegi: ";
    cin >> sisi;
    cout << "Luas persegi: " << sisi * sisi << endl;
```

Gambar 3,3

- Persegi Panjang

Rumus:  $\text{Luas} = \text{panjang} \times \text{lebar}$

Jika pengguna memilih opsi 2 (Persegi Panjang), program akan meminta input panjang dan lebar, lalu menghitung luasnya

```
} else if (luasChoice == 2) {
    double panjang, lebar;
    cout << "Masukkan panjang persegi panjang: ";
    cin >> panjang;
    cout << "Masukkan lebar persegi panjang: ";
    cin >> lebar;
    cout << "Luas persegi panjang: " << panjang * lebar << endl;
```



Gambar 3,4

#### - Lingkaran

Rumus:  $\text{Luas} = \pi \times r^2$  ( $\pi = 3.14$ )

Jika pengguna memilih opsi 3 (Lingkaran), program akan meminta input jari-jari dan menghitung luasnya.

```
} else if (luasChoice == 3) {  
    double jariJari;  
    cout << "Masukkan jari-jari lingkaran: ";  
    cin >> jariJari;  
    cout << "Luas lingkaran: " << 3.14 * jariJari * jariJari << endl;
```

Gambar 3,5

#### - Segitiga

Rumus:  $\text{Luas} = 0.5 \times \text{alas} \times \text{tinggi}$

Jika pengguna memilih opsi 4 (Segitiga), program akan meminta input alas dan tinggi, lalu menghitung luasnya.

```
} else if (luasChoice == 4) {  
    double alas, tinggi;  
    cout << "Masukkan alas segitiga: ";  
    cin >> alas;  
    cout << "Masukkan tinggi segitiga: ";  
    cin >> tinggi;  
    cout << "Luas segitiga: " << 0.5 * alas * tinggi << endl;
```

Gambar 3,6

## 2. Hitung Keliling

#### - Persegi

Rumus:  $\text{keliling} = 4 \times \text{sisi}$

Jika pengguna memilih opsi 1 (Persegi), program akan meminta input sisi dan menghitung kelilingnya.

```
if (kelilingChoice == 1) {  
    double sisi;  
    cout << "Masukkan panjang sisi persegi: ";  
    cin >> sisi;
```

```
cout << "Keliling persegi: " << 4 * sisi << endl;
```

Gambar 3,7

#### - Persegi panjang

Rumus: Keliling =  $2 \times (\text{panjang} + \text{lebar})$

Jika pengguna memilih opsi 2 (Persegi Panjang), program akan meminta input panjang dan lebar, lalu menghitung kelilingnya.

```
} else if (kelilingChoice == 2) {  
    double panjang, lebar;  
    cout << "Masukkan panjang persegi panjang: ";  
    cin >> panjang;  
    cout << "Masukkan lebar persegi panjang: ";  
    cin >> lebar;  
    cout << "Keliling persegi panjang: " << 2 * (panjang + lebar) << endl;
```

Gambar 3,8

#### - Lingkaran

Rumus: Keliling =  $2 \times \pi \times r$  ( $\pi = 3.14$ )

Jika pengguna memilih opsi 3 (Lingkaran), program akan meminta input jari-jari dan menghitung kelilingnya.

```
} else if (kelilingChoice == 3) {  
    double jariJari;  
    cout << "Masukkan jari-jari lingkaran: ";  
    cin >> jariJari;  
    cout << "Keliling lingkaran: " << 2 * 3.14 * jariJari << endl;
```

Gambar 3,9

#### - Segitiga

Rumus: Keliling = sisi1 + sisi2 + sisi3

Jika pengguna memilih opsi 4 (Segitiga), program akan meminta input ketiga sisi dan menghitung kelilingnya.

```
double sisi1, sisi2, sisi3;  
cout << "Masukkan panjang sisi 1 segitiga: ";  
cin >> sisi1;
```

```

cout << "Masukkan panjang sisi 2 segitiga: ";
cin >> sisi2;
cout << "Masukkan panjang sisi 3 segitiga: ";
cin >> sisi3;
cout << "Keliling segitiga: " << sisi1 + sisi2 + sisi3 << endl;

```

Gambar 3,10

### 3. Hitung Volume

#### - Kubus

Rumus:  $\text{Volume} = \text{sisi} \times \text{sisi} \times \text{sisi}$

Jika pengguna memilih opsi 1 (Kubus), program akan meminta input sisi dan menghitung volumenya.

```

if (volumeChoice == 1) {
    double sisi;
    cout << "Masukkan panjang sisi kubus: ";
    cin >> sisi;
    cout << "Volume kubus: " << sisi * sisi * sisi << endl;
}

```

Gambar 3,11

#### - Balok

Rumus:  $\text{Volume} = \text{panjang} \times \text{lebar} \times \text{tinggi}$

Jika pengguna memilih opsi 2 (Balok), program akan meminta input panjang, lebar, dan tinggi, lalu menghitung volumenya.

```

else if (volumeChoice == 2) {
    double panjang, lebar, tinggi;
    cout << "Masukkan panjang balok: ";
    cin >> panjang;
    cout << "Masukkan lebar balok: ";
    cin >> lebar;
    cout << "Masukkan tinggi balok: ";
    cin >> tinggi;
    cout << "Volume balok: " << panjang * lebar * tinggi << endl;
}

```

Gambar 3,12

#### - Bola

Rumus:  $\text{Volume} = \frac{4}{3} \times \pi \times r^3$  ( $\pi = 3.14$ )

Jika pengguna memilih opsi 3 (Bola), program akan meminta input jari-jari dan menghitung volumenya.

```
else if (volumeChoice == 3) {
    double jariJari;
    cout << "Masukkan jari-jari bola: ";
    cin >> jariJari;
    cout << "Volume bola: " << (4.0 / 3.0) * 3.14 * jariJari * jariJari * jariJari <<
endl;
} else if (volumeChoice == 4) {
```

Gambar 3,13

- Kerucut

Rumus:  $\text{Volume} = (1/3) \times \pi \times r^2 \times \text{tinggi}$  ( $\pi = 3.14$ )

Jika pengguna memilih opsi 4 (Kerucut), program akan meminta input jari-jari dan tinggi, lalu menghitung volumenya.

```
else if (volumeChoice == 4) {
    double jariJari, tinggi;
    cout << "Masukkan jari-jari kerucut: ";
    cin >> jariJari;
    cout << "Masukkan tinggi kerucut: ";
    cin >> tinggi;
    cout << "Volume kerucut: " << (1.0 / 3.0) * 3.14 * jariJari * jariJari * tinggi <<
endl;
} else {
```

Gambar 3,14

## D. Fitur Logout

Fitur logout adalah opsi yang memungkinkan pengguna untuk keluar dari program. Ketika pengguna memilih opsi logout, program akan menampilkan pesan Logout berhasil. “Terima kasih! sampai jumpa kembali..” dan menghentikan eksekusi program.

```
} else if (mainMenuChoice == 4) {
    // Logout
    loggedIn = false;
    cout << "Logout berhasil. Terima kasih! sampai jumpa kembali.." << endl;
```

Gambar 3,15

## 4. Uji Coba dan Hasil Output

### 4.1 Uji Coba

*(Jelaskan skenario yang digunakan untuk menguji program, misalnya dengan berbagai jenis input.)*

#### 1. Skenario 1

Pada skenario ini program menjalankan suatu sistem menghitung volume kubus, untuk menghitung volume kubus disini kita memasukkan panjang sisi kubus 10

```
Selamat Datang Silahkan Login >_<
Username: SayidAnnashirIkhwan
Password: 2409106097
Login berhasil!

== SISTEM HITUNG BANGUN DATAR DAN BANGUN RUANG ==
1. Hitung Luas
2. Hitung Keliling
3. Hitung Volume
4. Logout
Pilih menu: 3

== HITUNG VOLUME ==
1. Kubus
2. Balok
3. Bola
4. Kerucut
5. Kembali
Pilih bangun: 1
Masukkan panjang sisi kubus: 10
Volume kubus: 1000
```

Gambar 4,1 Hasil-Uji-Coba-Berhasil

#### 2. Skenario 2

Pada skenario ini jika user menginput nomor yang tidak sesuai di menu maka program akan menjadi berpesan “mohon masukkan angka dengan benar”, dan akan kembali ke menu awal untuk memilih lagi input yang benar

```

== SISTEM HITUNG BANGUN DATAR DAN BANGUN RUANG ==
1. Hitung Luas
2. Hitung Keliling
3. Hitung Volume
4. Logout
Pilih menu: 5
Mohon masukkan angka dengan benar.

== SISTEM HITUNG BANGUN DATAR DAN BANGUN RUANG ==
1. Hitung Luas
2. Hitung Keliling
3. Hitung Volume
4. Logout
Pilih menu: |

```

Gambar 4,2 Hasil-Uji-Coba-Gagal

## 4.2 Hasil Output

Contoh hasil Output dari perhitungan keliling segitiga

```

Selamat Datang Silahkan Login >_<
Username: SayidAnnashirIkhwan
Password: 2409106097
Login berhasil!

== SISTEM HITUNG BANGUN DATAR DAN BANGUN RUANG ==
1. Hitung Luas
2. Hitung Keliling
3. Hitung Volume
4. Logout
Pilih menu: 2

== HITUNG KELILING ==
1. Persegi
2. Persegi Panjang
3. Lingkaran
4. Segitiga
5. Kembali
Pilih bangun: 4
Masukkan panjang sisi 1 segitiga: 3
Masukkan panjang sisi 2 segitiga: 6
Masukkan panjang sisi 3 segitiga: 8
Keliling segitiga: 17

```

Gambar 4,3 Hasil-Output

## 5. GIT

```
MINGW64/c/Users/LOQ/Desktop/GITHUB/Praktikum-Apl

LOQ@LAPTOP-4B4VUDGG MINGW64 ~/Desktop/GITHUB/Praktikum-Apl
$ git confif --global user.email "sayidannasir@gmail.com"
git: 'confif' is not a git command. See 'git --help'.

The most similar command is
    config

LOQ@LAPTOP-4B4VUDGG MINGW64 ~/Desktop/GITHUB/Praktikum-Apl
$ AC

LOQ@LAPTOP-4B4VUDGG MINGW64 ~/Desktop/GITHUB/Praktikum-Apl
$ git config --global user.email "sayidannasir@gmail.com"

LOQ@LAPTOP-4B4VUDGG MINGW64 ~/Desktop/GITHUB/Praktikum-Apl
$ git init
Initialized empty Git repository in C:/Users/LOQ/Desktop/GITHUB/Praktikum-Apl/.git/

LOQ@LAPTOP-4B4VUDGG MINGW64 ~/Desktop/GITHUB/Praktikum-Apl (master)
$ git add .

LOQ@LAPTOP-4B4VUDGG MINGW64 ~/Desktop/GITHUB/Praktikum-Apl (master)
$ git commit -m "Sayid code"
[master (root-commit) be12ed1] Sayid code
 3 files changed, 121 insertions(+)
   create mode 100644 Post-test/Post-test-1/2409106097-SayidAnnashirIkhwan-PT-1.cp
p
   create mode 100644 Post-test/Post-test-1/2409106097-SayidAnnashirIkhwan-PT-1.do
cx
   create mode 100644 Post-test/Post-test-1/2409106097-SayidAnnashirIkhwan-PT-1.ex
e

LOQ@LAPTOP-4B4VUDGG MINGW64 ~/Desktop/GITHUB/Praktikum-Apl (master)
$ git branch -M main

LOQ@LAPTOP-4B4VUDGG MINGW64 ~/Desktop/GITHUB/Praktikum-Apl (main)
$ git remote add origin https://github.com/wizker12/Praktikum-Apl.git

LOQ@LAPTOP-4B4VUDGG MINGW64 ~/Desktop/GITHUB/Praktikum-Apl (main)
$ git push -u origin main
info: please complete authentication in your browser...
Enumerating objects: 7, done.
Counting objects: 100% (7/7), done.
Delta compression using up to 20 threads
Compressing objects: 100% (5/5), done.
Writing objects: 100% (7/7), 1.01 MiB | 749.00 KiB/s, done.
Total 7 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/wizker12/Praktikum-Apl.git
 * [new branch]    main -> main
branch 'main' set up to track 'origin/main'.

LOQ@LAPTOP-4B4VUDGG MINGW64 ~/Desktop/GITHUB/Praktikum-Apl (main)
$ |
```

### 1. Konfigurasi Git User Email

\$ git config --global user.email "[sayidannasri@gmail.com](mailto:sayidannasri@gmail.com)"

Perintah ini digunakan untuk mengatur email global yang akan digunakan untuk setiap commit dalam repositori Git. Namun, terjadi kesalahan karena perintah yang dimasukkan adalah git conf, yang bukan perintah yang valid.

Kemudian, kesalahan diperbaiki dengan menjalankan: \$ git config --global user.email "[sayidannasri@gmail.com](mailto:sayidannasri@gmail.com)"

Sekarang Git sudah memiliki konfigurasi email pengguna.

## **2. Inisialisasi Repositori Git**

“\$ git init”

Perintah ini digunakan untuk menginisialisasi repository Git baru di dalam direktori Praktikum-Apl. Dengan ini, direktori sekarang sudah menjadi repository Git yang bisa digunakan untuk versi kontrol.

## **3. Menambahkan File ke Staging Area**

“\$ git add .”

Perintah ini digunakan untuk menambahkan semua file yang ada di direktori ke dalam staging area, sehingga siap untuk di-commit.

## **4. Commit Perubahan**

\$ git commit -m "Sayid code"

Commit dilakukan dengan pesan "Sayid code", yang berarti semua perubahan yang sudah ditambahkan ke staging area disimpan dalam repository lokal dengan deskripsi tersebut.

## **5. Mengubah Nama Branch ke main**

\$ git branch -m main

Perintah ini digunakan untuk mengganti nama branch default dari master menjadi main. Ini penting karena banyak platform Git seperti GitHub sekarang menggunakan main sebagai branch utama secara default.

## **6. Menambahkan Remote Repository**

\$ git remote add origin <https://github.com/wizker12/Praktikum-Apl.git>

Dengan perintah ini, repository lokal dikaitkan dengan repository di GitHub yang memiliki URL <https://github.com/wizker12/Praktikum-Apl.git>. Ini memungkinkan repository lokal untuk didorong (push) ke GitHub.

## **7. Mendorong (Push) Repository ke GitHub**

\$ git push -u origin main



Perintah ini digunakan untuk mengunggah semua commit dari branch main di repository lokal ke repository remote (origin). Hasil output menunjukkan bahwa operasi push berhasil dilakukan dan repository sudah disinkronkan dengan GitHub.