

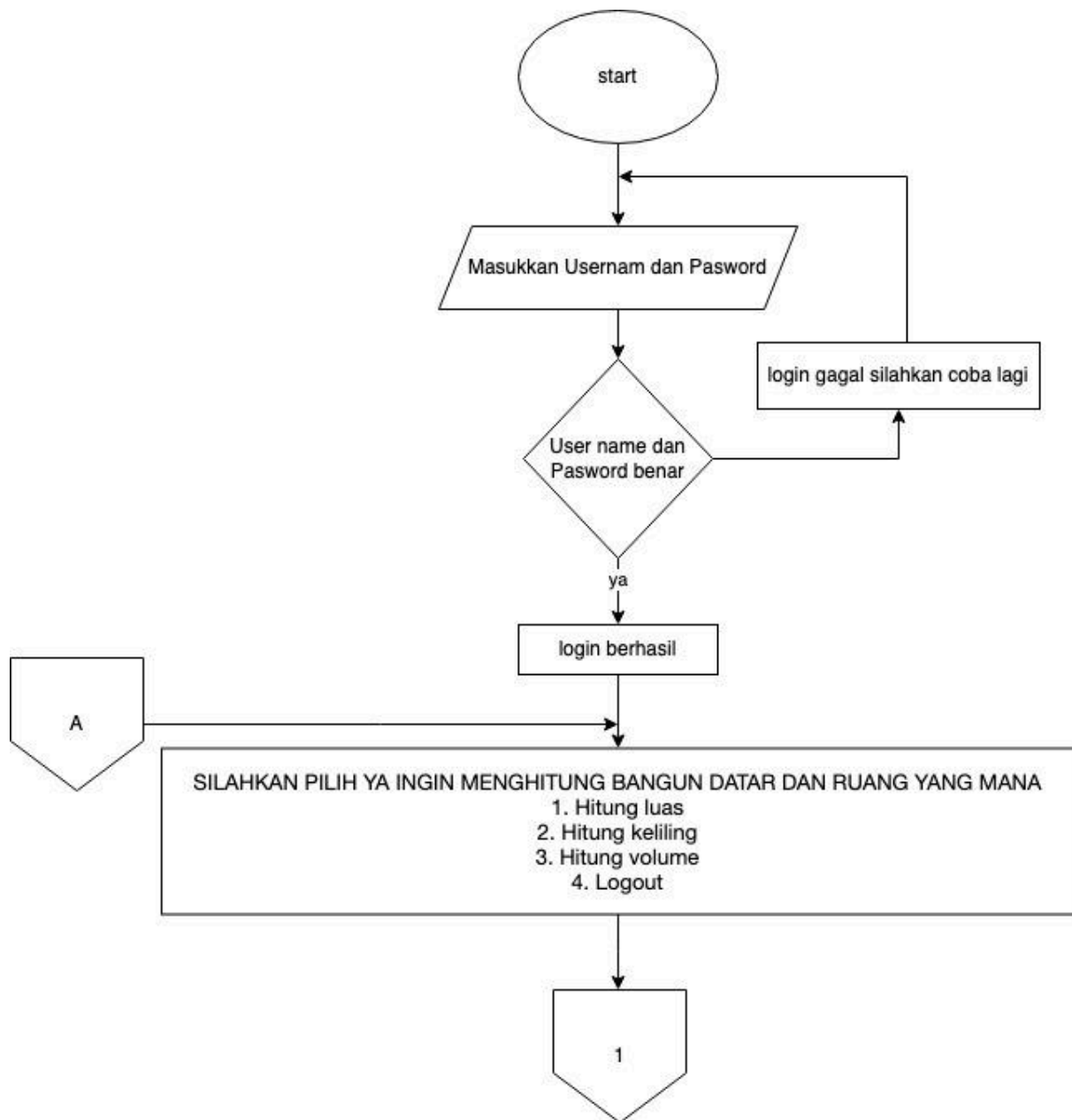
**LAPORAN PRAKTIKUM**  
**POSTTEST 1**  
**ALGORITMA PEMROGRAMAN LANJUT**



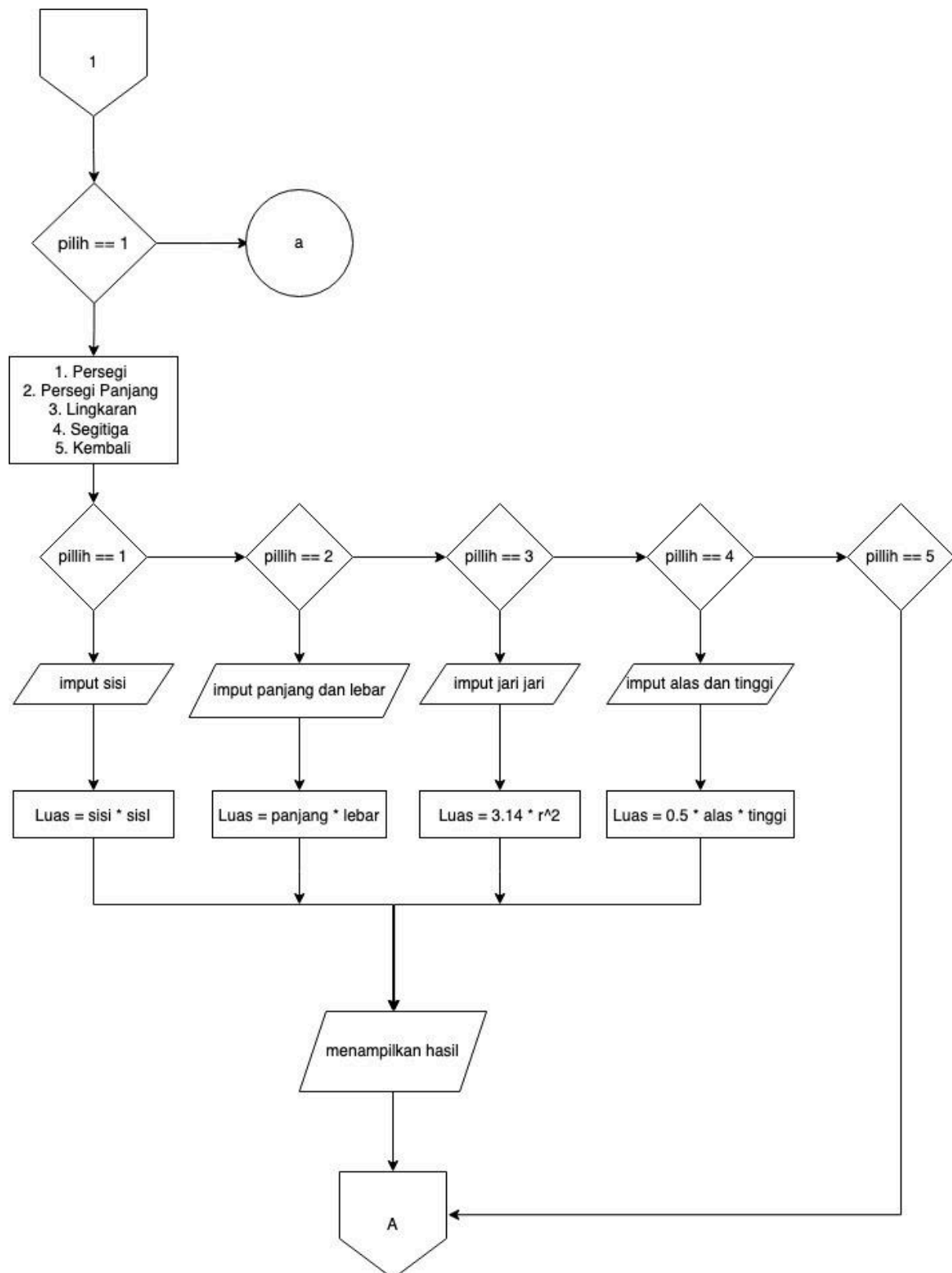
**Disusun oleh:**  
**Nama (2409106094)**  
**Kelas (c'1)**

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

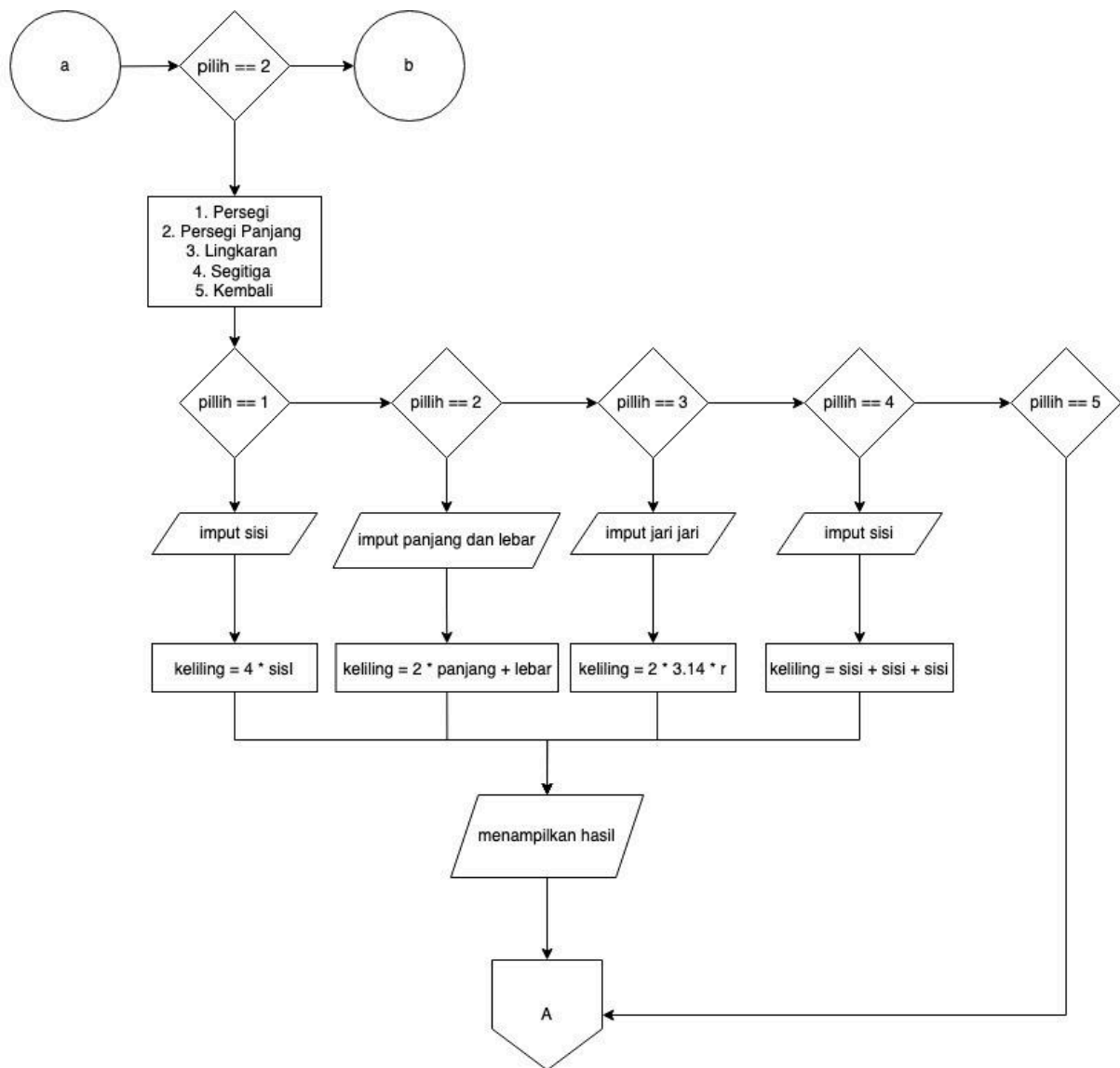
## 1. Flowchart

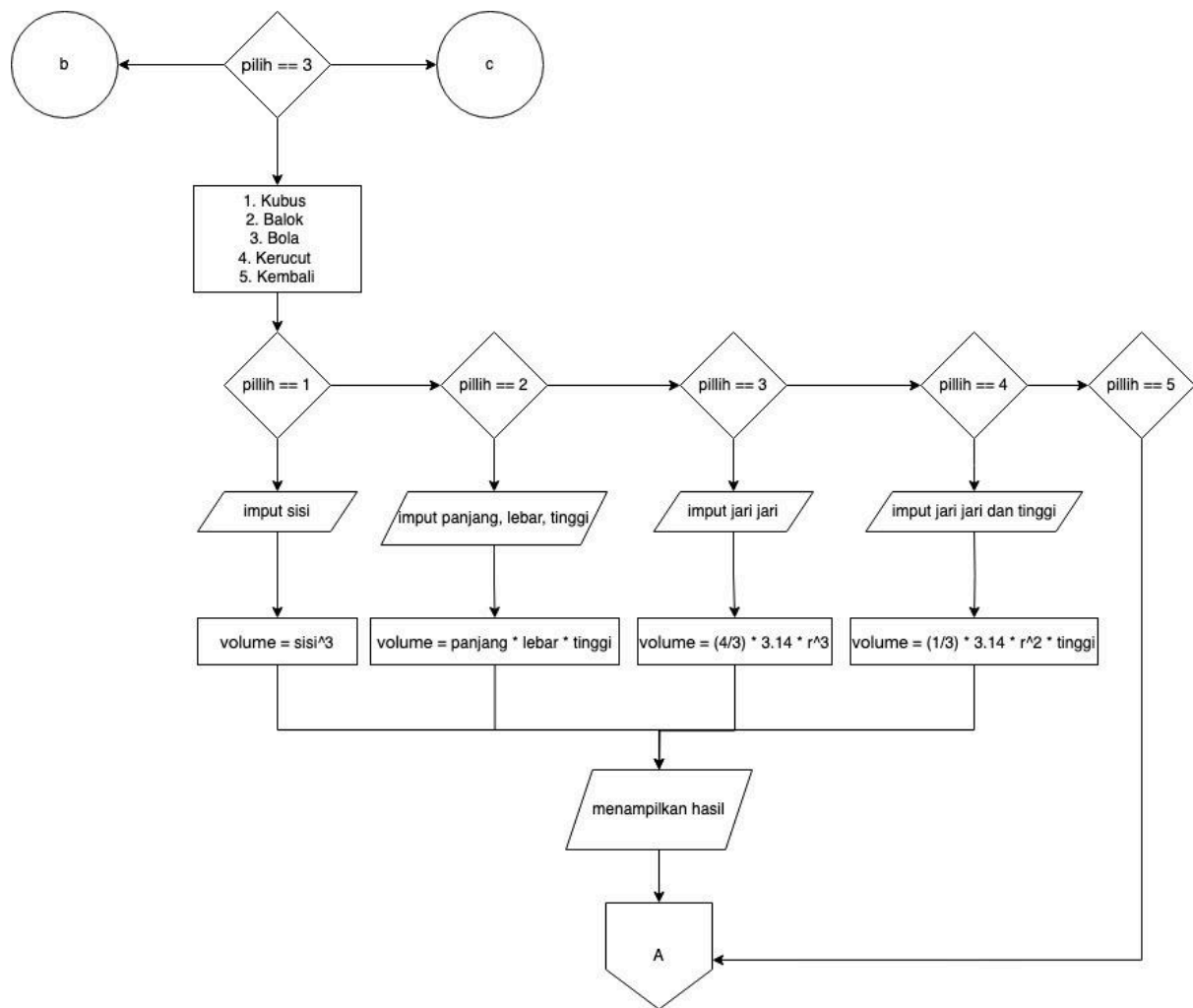


jadi ini adalah menu awal dari program nya, tapi sebelum masuk ke program nya kita harus terlebih dahulu

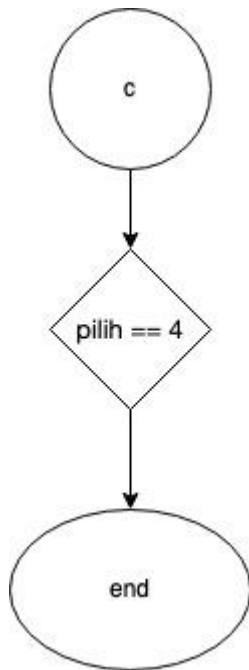


selanjut nya ini adalah jika kamu memilih opsi hitung luas, kamu bisa memilih salah satu opsi nya di sini saya akan menjelaskan salah satunya semisa 1 (persegi) jika dia ada di opsi yang kamu pilih dia akan menyuruh menginput seperti yang saya pilih tadi adalah persegi saya di suruh menginput sisi, setelah memasukan sisi program akan menghitung lalu menampilkannya dan akan kembali ke menu utama.





nah untuk dua flowchart di atas kurang lebih seperti penjelasan sebelum bisa memilih dan menginput bilangan yang di punya.



untuk di atas ini fungsinya untuk menutup program atau memberhentikan program nya

## 2. Analisis Program

### 2.1 Deskripsi Singkat Program

program ini dibuat untuk menghitung sebuah bangun datar dan ruang yang ingin di hitung.

### 2.2 Penjelasan Alur & Algoritma

#### 1. Login User

```
bool Login() {
    string username, password;
    do {
        cout << "🔑 masukan 🔑\n";
        cout << "👤 Username: ";
        getline(cin, username);
        cin.ignore();
        cout << "🔑 Password: ";
        cin >> password;

        if (username == "kelvin alviansyah" && password ==
"2409106094") {
            return true;
        } else {
            cout << "Login gagal! Silakan coba lagi.\n";
        }
    }
```

```

    } while (true);
}

```

Jadi gini, programnya bakal minta username sama password ke pengguna. Kalau bener, langsung masuk ke menu utama. Tapi kalau salah, bakal muncul pesan "Login gagal", terus disuruh coba lagi. Nah, di sini pakai looping do-while, biar programnya nggak berhenti sampai user masukin username dan password yang benar.

## 2. Menu Utama

```

do {
    cout <<
    ".-----\n";
    cout << "|  SILAHKAN PILIH YA INGIN MENGHITUNG BANGUN DATAR
DAN RUANG YANG MANA  |\n";
    cout << "| 1. Hitung luas
|\n";
    cout << "| 2. Hitung keliling
|\n";
    cout << "| 3. Hitung volume
|\n";
    cout << "| 4. Logout
|\n";
    cout <<
    "| _____
_____| \n";
    cout << "\n";
    cout << "Pilih: ";
    cin >> pilihan;
}

```

Setelah berhasil login, program bakal nunjukin menu utama. Di sini, pengguna bisa milih mau hitung luas, keliling, volume, atau langsung logout. Pilihan yang dipilih user bakal disimpan di variabel pilihan.

## 3. Menghitung Luas

```

double hitungLuas() {
    int pilihan;
    double sisi, panjang, lebar, jariJari, alas, tinggi;
}

```

```

do {
    cout << "\n-- HITUNG LUAS --\n";
    cout << "1. Persegi\n";
    cout << "2. Persegi Panjang\n";
    cout << "3. Lingkaran\n";
    cout << "4. Segitiga\n";
    cout << "5. Kembali\n";
    cout << "Pilih: ";
    cin >> pilihan;

    switch (pilihan) {
        case 1:
            cout << "Masukkan sisi: ";
            cin >> sisi;
            return sisi * sisi;
        case 2:
            cout << "Masukkan panjang dan lebar: ";
            cin >> panjang >> lebar;
            return panjang * lebar;
        case 3:
            cout << "Masukkan jari-jari: ";
            cin >> jariJari;
            return 3.14 * jariJari * jariJari;
        case 4:
            cout << "Masukkan alas dan tinggi: ";
            cin >> alas >> tinggi;
            return 0.5 * alas * tinggi;
    }
} while (pilihan != 5);
return 0;
}

```

Pengguna bakal milih **bangun datar** yang mau dihitung luasnya. Setelah itu, mereka masukin **input** sesuai bentuk yang dipilih, terus program langsung ngitung luasnya pake **rumus yang sesuai**.

Kalau misalnya pengguna milih **"5. Kembali"**, program bakal **balik ke menu sebelumnya** dengan nge-return nilai **0**.

#### 4. Menghitung Keliling

```

double hitungKeliling() {
    int pilihan;
    double sisi, panjang, lebar, jariJari, sisi1, sisi2, sisi3;

```



```

do {
    cout << "\n-- HITUNG KELILING --\n";
    cout << "1. Persegi\n";
    cout << "2. Persegi Panjang\n";
    cout << "3. Lingkaran\n";
    cout << "4. Segitiga\n";
    cout << "5. Kembali\n";
    cout << "Pilih: ";
    cin >> pilihan;

    switch (pilihan) {
        case 1:
            cout << "Masukkan sisi: ";
            cin >> sisi;
            return 4 * sisi;
        case 2:
            cout << "Masukkan panjang dan lebar: ";
            cin >> panjang >> lebar;
            return 2 * (panjang + lebar);
        case 3:
            cout << "Masukkan jari-jari: ";
            cin >> jariJari;
            return 2 * 3.14 * jariJari;
        case 4:
            cout << "Masukkan tiga sisi segitiga: ";
            cin >> sisi1 >> sisi2 >> sisi3;
            return sisi1 + sisi2 + sisi3;
    }
} while (pilihan != 5);
return 0;
}

```

Prosesnya kurang lebih sama kayak di `hitungLuas()`, tapi kali ini buat hitung keliling bangun datar. Pengguna masukan input yang dibutuhkan, terus program bakal ngitung kelilingnya pake rumus yang sesuai buat tiap bentuk.

## 5. Menghitung Volume

```

double hitungVolume() {
    int pilihan;
    double sisi, panjang, lebar, tinggi, jariJari;

    do {
        cout << "\n-- HITUNG VOLUME --\n";

```

```

        cout << "1. Kubus\n";
        cout << "2. Balok\n";
        cout << "3. Bola\n";
        cout << "4. Kerucut\n";
        cout << "5. Kembali\n";
        cout << "Pilih: ";
        cin >> pilihan;

        switch (pilihan) {
            case 1:
                cout << "Masukkan sisi: ";
                cin >> sisi;
                return sisi * sisi * sisi;
            case 2:
                cout << "Masukkan panjang, lebar, dan tinggi: ";
                cin >> panjang >> lebar >> tinggi;
                return panjang * lebar * tinggi;
            case 3:
                cout << "Masukkan jari-jari: ";
                cin >> jariJari;
                return (4.0 / 3.0) * 3.14 * jariJari * jariJari *
jariJari;
            case 4:
                cout << "Masukkan jari-jari dan tinggi: ";
                cin >> jariJari >> tinggi;
                return (1.0 / 3.0) * 3.14 * jariJari * jariJari *
tinggi;
        }
    } while (pilihan != 5);
    return 0;
}

```

Jadi, program bakal minta input sesuai dengan bangun ruang yang dipilih. Setelah itu, program bakal ngitung volume bangun ruang itu pake rumus volume yang sesuai buat masing-masing bentuk. Misalnya, kalau milih kubus, program bakal pakai rumus volume kubus.

## 6. Logout & Program Berhenti

```

case 4:
    cout << "Logout berhasil.\n";
    break;

```

Kalau pengguna milih opsi 4 di menu utama, program bakal langsung keluar dan ngebroadcast pesan "Logout berhasil." jadi user tahu kalau mereka udah keluar dari aplikasi.

### 3. Source Code

```
#include <iostream>
#include <cmath>
using namespace std;
// Fungsi login
bool login() {
    string username, password;
    do {
        cout << "🔒 masukan 🔒\n";
        cout << "👤 Username: ";
        getline(cin, username);
        cin.ignore();
        cout << "🔑 Password: ";
        cin >> password;

        if (username == "kelvin alviansyah" && password == "2409106094") {
            return true;
        } else {
            cout << "Login gagal! Silakan coba lagi.\n";
        }
    } while (true);
}
// Fungsi menghitung luas
double hitungLuas() {
    int pilihan;
    double sisi, panjang, lebar, jariJari, alas, tinggi;
    do {
        cout << "\n-- HITUNG LUAS --\n";
        cout << "1. Persegi\n";
        cout << "2. Persegi Panjang\n";
        cout << "3. Lingkaran\n";
        cout << "4. Segitiga\n";
        cout << "5. Kembali\n";
        cout << "Pilih: ";
        cin >> pilihan;
        switch (pilihan) {
            case 1:
                cout << "Masukkan sisi: ";
                cin >> sisi;
                return sisi * sisi;
            case 2:
                cout << "Masukkan panjang dan lebar: ";
                cin >> panjang >> lebar;
                return panjang * lebar;
            case 3:
                cout << "Masukkan jari-jari: ";
                cin >> jariJari;
                return 3.14 * jariJari * jariJari;
            case 4:
                cout << "Masukkan alas dan tinggi: ";
```

```

        cin >> alas >> tinggi;
        return 0.5 * alas * tinggi;
    }
} while (pilihan != 5);
return 0; // Kembali 0 jika tidak ada pilihan yang valid
}

// Fungsi menghitung keliling
double hitungKeliling() {
    int pilihan;
    double sisi, panjang, lebar, jariJari, sisi1, sisi2, sisi3;
    do {
        cout << "\n-- HITUNG KELILING --\n";
        cout << "1. Persegi\n";
        cout << "2. Persegi Panjang\n";
        cout << "3. Lingkaran\n";
        cout << "4. Segitiga\n";
        cout << "5. Kembali\n";
        cout << "Pilih: ";
        cin >> pilihan;
        switch (pilihan) {
            case 1:
                cout << "Masukkan sisi: ";
                cin >> sisi;
                return 4 * sisi;
            case 2:
                cout << "Masukkan panjang dan lebar: ";
                cin >> panjang >> lebar;
                return 2 * (panjang + lebar);
            case 3:
                cout << "Masukkan jari-jari: ";
                cin >> jariJari;
                return 2 * 3.14 * jariJari;
            case 4:
                cout << "Masukkan tiga sisi segitiga: ";
                cin >> sisi1 >> sisi2 >> sisi3;
                return sisi1 + sisi2 + sisi3;
        }
    } while (pilihan != 5);
    return 0; // Kembali 0 jika tidak ada pilihan yang valid
}

// Fungsi menghitung volume
double hitungVolume() {
    int pilihan;
    double sisi, panjang, lebar, tinggi, jariJari;
    do {
        cout << "\n-- HITUNG VOLUME --\n";
        cout << "1. Kubus\n";
        cout << "2. Balok\n";
        cout << "3. Bola\n";
        cout << "4. Kerucut\n";
        cout << "5. Kembali\n";
        cout << "Pilih: ";
        cin >> pilihan;
        switch (pilihan) {
            case 1:
                cout << "Masukkan sisi: ";

```

```

        cin >> sisi;
        return sisi * sisi * sisi;
    case 2:
        cout << "Masukkan panjang, lebar, dan tinggi: ";
        cin >> panjang >> lebar >> tinggi;
        return panjang * lebar * tinggi;
    case 3:
        cout << "Masukkan jari-jari: ";
        cin >> jariJari;
        return (4.0 / 3.0) * 3.14 * jariJari * jariJari * jariJari;
    case 4:
        cout << "Masukkan jari-jari dan tinggi: ";
        cin >> jariJari >> tinggi;
        return (1.0 / 3.0) * 3.14 * jariJari * jariJari * tinggi;
    }
} while (pilihan != 5);
return 0; // Kembali 0 jika tidak ada pilihan yang valid
}

// Fungsi utama program
int main() {
    login();
    int pilihan;
    do {
        cout <<
        ".-----.\n";
        cout << "|  SILAHKAN PILIH YA INGIN MENGHITUNG BANGUN DATAR DAN RUANG YANG MANA\n";
        cout << "| 1. Hitung luas\n";
        cout << "| 2. Hitung keliling\n";
        cout << "| 3. Hitung volume\n";
        cout << "| 4. Logout\n";
        cout <<
        "|_____|\n";
        cout << "\n";
        cout << "Pilih: ";
        cin >> pilihan;
        switch (pilihan) {
            case 1: {
                double luas = hitungLuas();
                cout << "Hasil Luas: " << luas << endl;
                break;
            }
            case 2: {
                double keliling = hitungKeliling();
                cout << "Hasil Keliling: " << keliling << endl;
                break;
            }
            case 3: {
                double volume = hitungVolume();
                cout << "Hasil Volume: " << volume << endl;
                break;
            }
        }
    }
}

```

```

        case 4:
            cout << "Logout berhasil.\n";
            break;
        default:
            cout << "Pilihan tidak valid.\n";
    }
} while (pilihan != 4);
return 0;
}



```

## 4. Uji Coba dan Hasil Output


### 4.1 Uji Coba

#### A. Pengujian Login

##### Skenario 1.1: Login Berhasil

 masukan 

 Username: kelvin alviansyah

 Password: 2409106094

```

-----
| SILAHKAN PILIH YA INGIN MENGHITUNG BANGUN DATAR DAN RUANG YANG MANA |
| 1. Hitung luas                                                         |
| 2. Hitung keliling                                                     |
| 3. Hitung volume                                                       |
| 4. Logout                                                              |
|_____
Pilih:


```



##### Skenario 1.2: Login Gagal

 masukan 

 Username: 1

 Password: 1

 Login gagal! Silakan coba lagi.

 masukan 

 Username:

#### B. Pengujian Input Menu Utama

##### Skenario 2.1: Memilih Opsi "Logout"

```

-----
| SILAHKAN PILIH YA INGIN MENGHITUNG BANGUN DATAR DAN RUANG YANG MANA |
| 1. Hitung luas                                                         |
| 2. Hitung keliling                                                     |
| 3. Hitung volume                                                       |
| 4. Logout                                                              |
|_____

```

Pilih: 4

Logout berhasil.

Program ended with exit code: 0

## Skenario 2.2: Memilih Opsi Tidak Valid

Pilih: 5

Pilihan tidak valid.

```
-----
| SILAHKAN PILIH YA INGIN MENGHITUNG BANGUN DATAR DAN RUANG YANG MANA |
| 1. Hitung luas |
| 2. Hitung keliling |
| 3. Hitung volume |
| 4. Logout |
|_____|
Pilih:
```

## C. Pengujian Perhitungan Luas

### Skenario 3.1: Menghitung Luas Persegi

Pilih: 1

-- HITUNG LUAS --

1. Persegi
2. Persegi Panjang
3. Lingkaran
4. Segitiga
5. Kembali

Pilih: 1

Masukkan sisi: 1

Hasil Luas: 1

### Skenario 3.2: Menghitung Luas Segitiga

-- HITUNG LUAS --

1. Persegi
2. Persegi Panjang
3. Lingkaran
4. Segitiga
5. Kembali

Pilih: 4

Masukkan alas dan tinggi: 1

1

Hasil Luas: 0.5

## D. Pengujian Perhitungan Keliling

### Skenario 4.1: Menghitung Keliling Persegi

-- HITUNG KELILING --

1. Persegi
2. Persegi Panjang
3. Lingkaran
4. Segitiga
5. Kembali

Pilih: 1

Masukkan sisi: 1

Hasil Keliling: 4

## E. Pengujian Perhitungan Volume

### Skenario 5.1: Menghitung Volume balok

-- HITUNG VOLUME --

1. Kubus
2. Balok
3. Bola
4. Kerucut
5. Kembali

Pilih: 2

Masukkan panjang, lebar, dan tinggi: 2

2

2

Hasil Volume: 8

## 4.2 Hasil Output

### A. Pengujian Login

```
🔒 masukan 🔒
👤 Username: kelvin alviansyah
🔑 Password: 2409106094
✅ Login berhasil!

.
|  SILAHKAN PILIH YA INGIN MENGHITUNG BANGUN DATAR DAN RUANG YANG MANA  |
|  1. Hitung luas                                                            |
|  2. Hitung keliling                                                       |
|  3. Hitung volume                                                         |
|  4. Logout                                                                |
|  .-----|
Pilih: |
```



## B. Pengujian Input Menu Utama

```
.-----.  
|  SILAHKAN PILIH YA INGIN MENGHITUNG BANGUN DATAR DAN RUANG YANG MANA  |  
|  1. Hitung luas  |  
|  2. Hitung keliling  |  
|  3. Hitung volume  |  
|  4. Logout  |  
|-----|  
  
Pilih: 1  
  
-- HITUNG LUAS --  
1. Persegi  
2. Persegi Panjang  
3. Lingkaran  
4. Segitiga  
5. Kembali  
Pilih: |
```

## C. Pengujian Input Hitung Luas

```
-- HITUNG LUAS --  
1. Persegi  
2. Persegi Panjang  
3. Lingkaran  
4. Segitiga  
5. Kembali  
Pilih: 1  
Masukkan sisi: 1  
Hasil Luas: 1
```

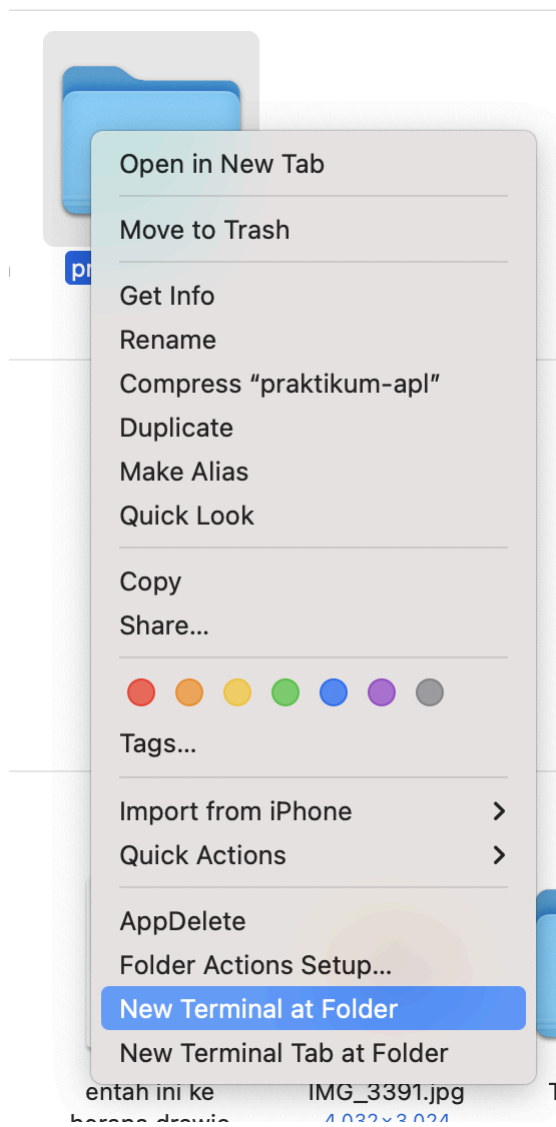
#### D. Pengujian Input Hitung Keliling

```
-- HITUNG KELILING --  
1. Persegi  
2. Persegi Panjang  
3. Lingkaran  
4. Segitiga  
5. Kembali  
Pilih: 1  
Masukkan sisi: 1  
Hasil Keliling: 4
```

#### E. Pengujian Input Hitung Volume

```
-- HITUNG VOLUME --  
1. Kubus  
2. Balok  
3. Bola  
4. Kerucut  
5. Kembali  
Pilih: 1  
Masukkan sisi: 1  
Hasil Volume: 1
```

## 5. GIT



Petama klik kanan di folder lalu pilih seperti di gambar

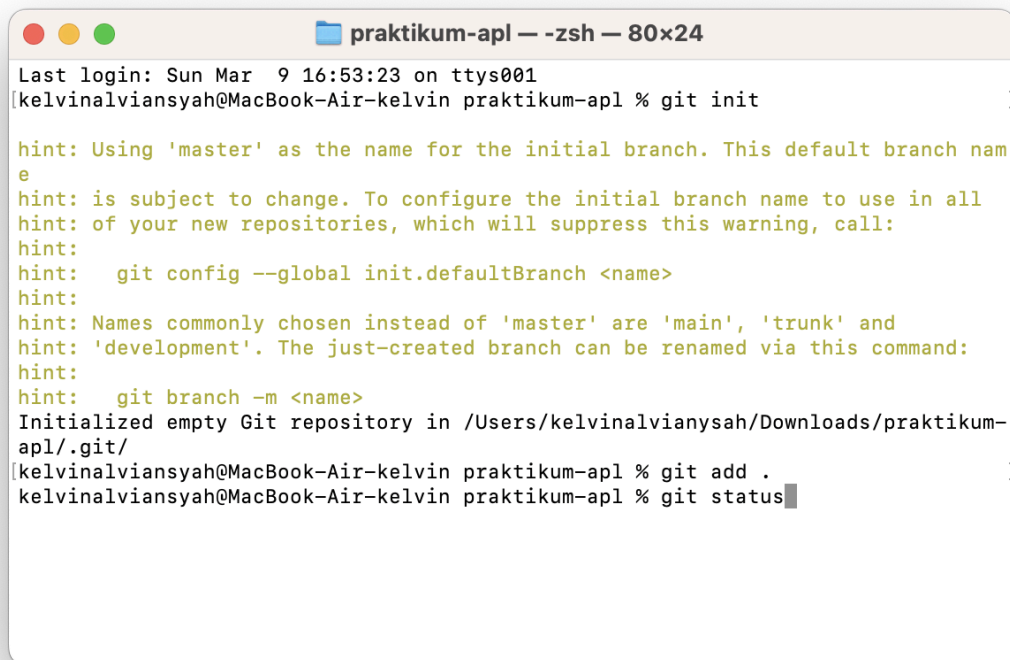
```
praktikum-apl — -zsh — 80x24
Last login: Sun Mar  9 16:53:23 on ttys001
kelvinalviansyah@MacBook-Air-kelvin praktikum-apl % git init
```

lalu ketik `git init`

```
praktikum-apl — -zsh — 80x24
Last login: Sun Mar  9 16:53:23 on ttys001
kelvinalviansyah@MacBook-Air-kelvin praktikum-apl % git init

hint: Using 'master' as the name for the initial branch. This default branch name
hint: is subject to change. To configure the initial branch name to use in all
hint: of your new repositories, which will suppress this warning, call:
hint:
hint:   git config --global init.defaultBranch <name>
hint:
hint: Names commonly chosen instead of 'master' are 'main', 'trunk' and
hint: 'development'. The just-created branch can be renamed via this command:
hint:
hint:   git branch -m <name>
Initialized empty Git repository in /Users/kelvinalvianysah/Downloads/praktikum-
apl/.git/
kelvinalviansyah@MacBook-Air-kelvin praktikum-apl % git add .
```

lalu ketikkan `git add .`

A terminal window titled "praktikum-apl — -zsh — 80x24" showing the execution of git commands. The output includes a login message, a successful git init command with several hints about the default branch name, and the execution of git add . and git status commands.

```
praktikum-apl — -zsh — 80x24
Last login: Sun Mar  9 16:53:23 on ttys001
[kelvinalviansyah@MacBook-Air-kelvin praktikum-apl % git init

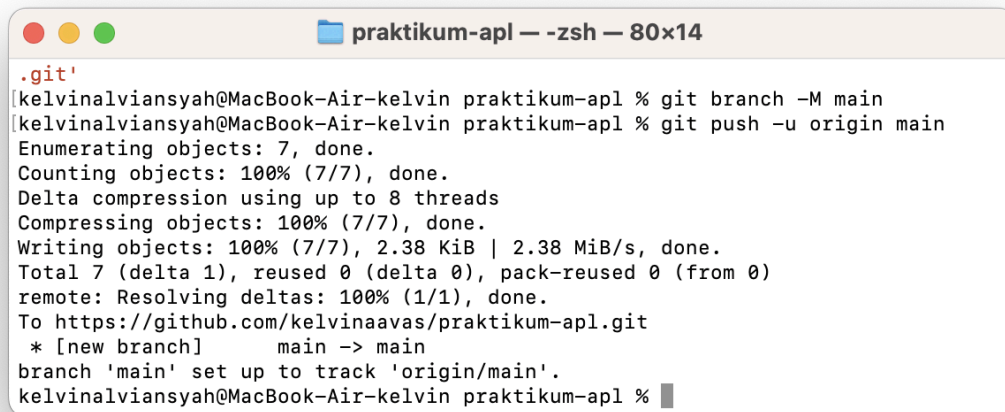
hint: Using 'master' as the name for the initial branch. This default branch nam
e
hint: is subject to change. To configure the initial branch name to use in all
hint: of your new repositories, which will suppress this warning, call:
hint:
hint:   git config --global init.defaultBranch <name>
hint:
hint: Names commonly chosen instead of 'master' are 'main', 'trunk' and
hint: 'development'. The just-created branch can be renamed via this command:
hint:
hint:   git branch -m <name>
Initialized empty Git repository in /Users/kelvinalviansyah/Downloads/praktikum-
apl/.git/
[kelvinalviansyah@MacBook-Air-kelvin praktikum-apl % git add .
[kelvinalviansyah@MacBook-Air-kelvin praktikum-apl % git status
```

selanjutnya ketikkan `git status`

```
[kelvinalviansyah@MacBook-Air-kelvin praktikum-apl % git remote add origin https:]
//github.com/kelvinaavas/praktikum-apl.git
```

lalu masukan code yang ada di github contohnya

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

A terminal window titled "praktikum-apl — -zsh — 80x14" with standard macOS window controls (red, yellow, green buttons). The terminal shows the execution of two git commands. The first command, "git branch -M main", is followed by the second command, "git push -u origin main". The output of the push command shows the process of enumerating, counting, compressing, and writing objects, followed by the creation of the 'main' branch and its tracking of 'origin/main'.

```
.git'  
[kelvinalviansyah@MacBook-Air-kelvin praktikum-apl % git branch -M main  
[kelvinalviansyah@MacBook-Air-kelvin praktikum-apl % git push -u origin main  
Enumerating objects: 7, done.  
Counting objects: 100% (7/7), done.  
Delta compression using up to 8 threads  
Compressing objects: 100% (7/7), done.  
Writing objects: 100% (7/7), 2.38 KiB | 2.38 MiB/s, done.  
Total 7 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)  
remote: Resolving deltas: 100% (1/1), done.  
To https://github.com/kelvinaavas/praktikum-apl.git  
* [new branch]      main -> main  
branch 'main' set up to track 'origin/main'.  
kelvinalviansyah@MacBook-Air-kelvin praktikum-apl %
```

lalu ketikan `git branch -M main` enter dan ketikan lagi `git push -u origin main` dan file sudah masuk ke github