

LAPORAN PRAKTIKUM
POSTTEST 1
ALGORITMA PEMROGRAMAN LANJUT



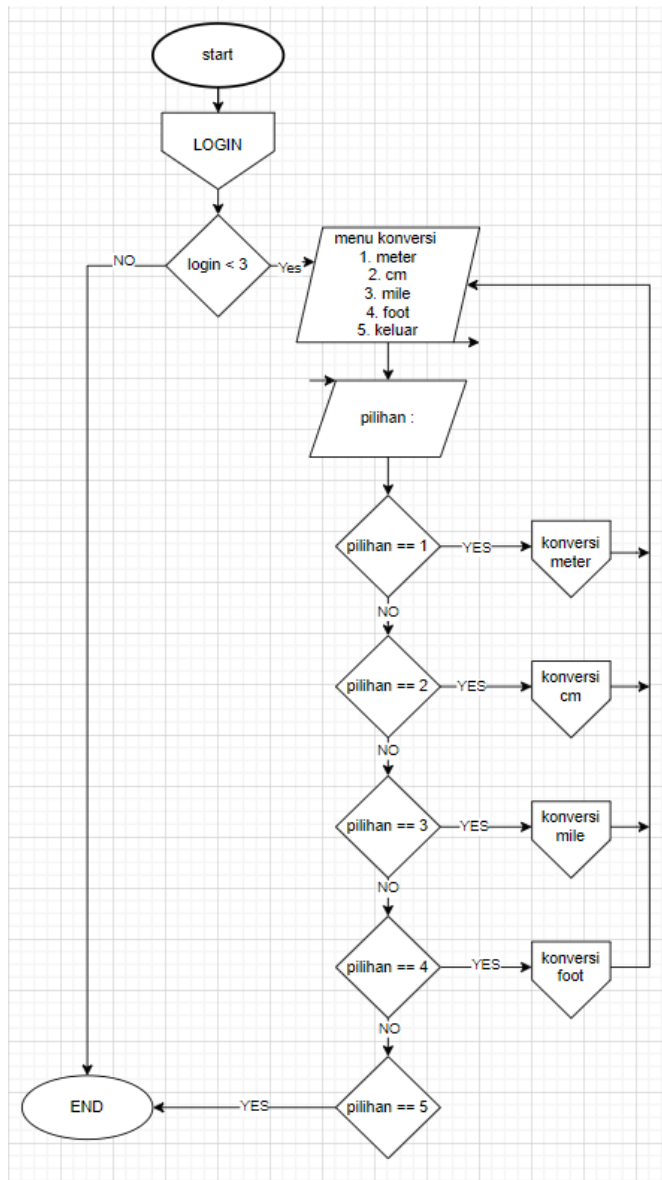
Disusun oleh:

Ayu Azzahrah Alwi (2409106022)

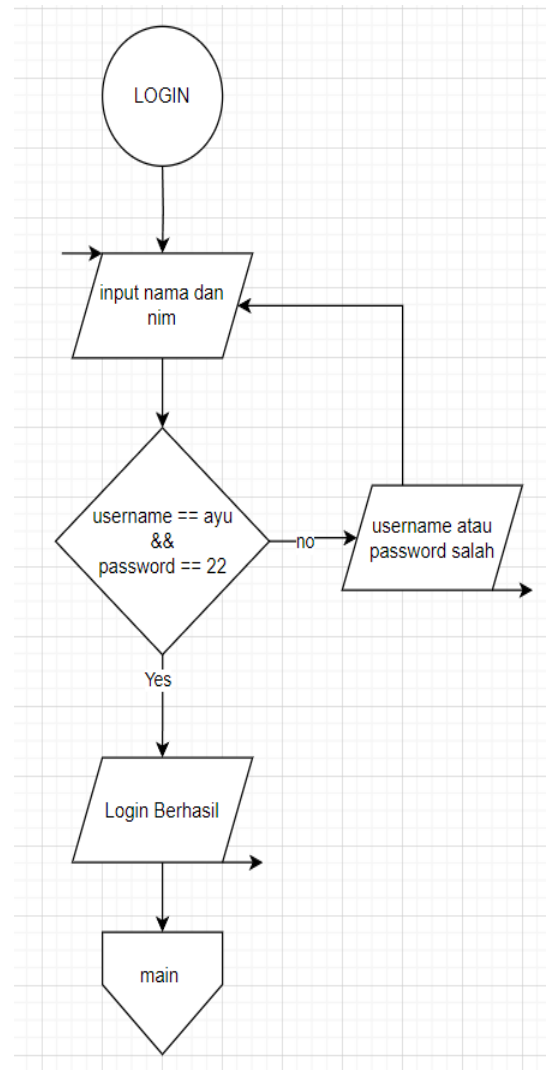
Kelas A1'24

PROGRAM STUDI INFORMATIKA
UNIVERSITAS MULAWARMAN
SAMARINDA
2025

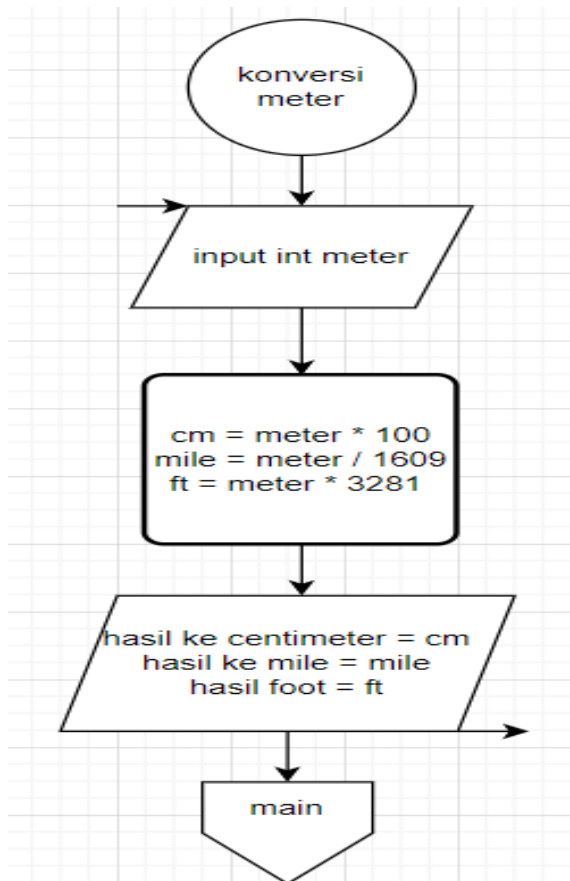
1. Flowchart



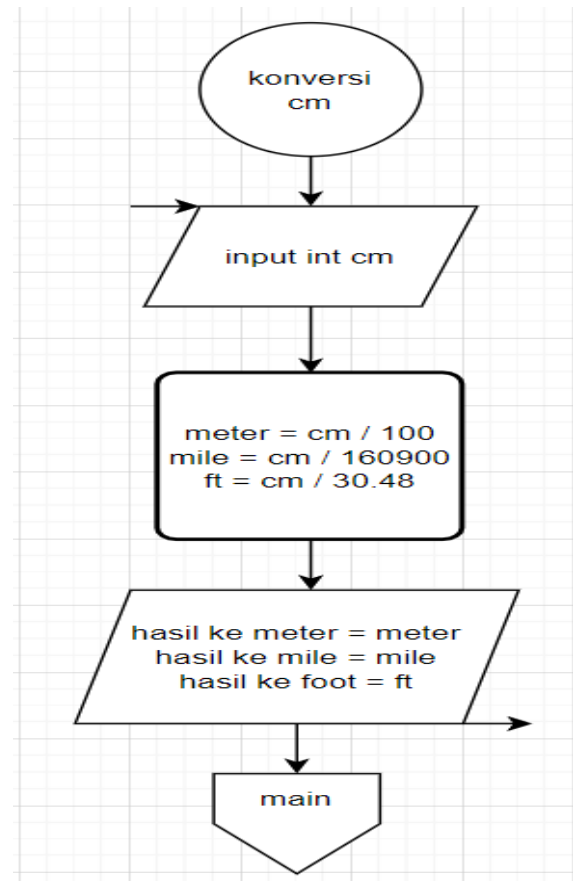
Gambar 1. 2 Main Flow



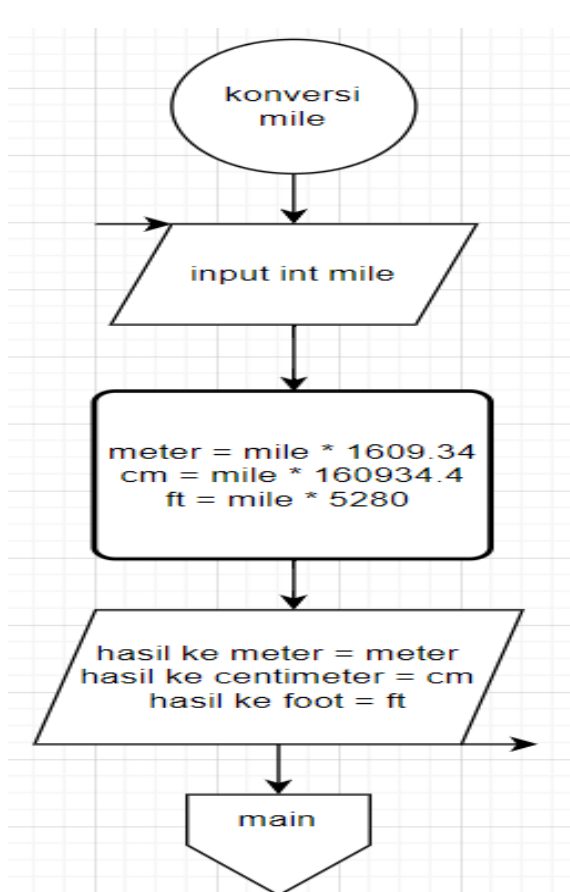
Gambar 1. 1 On-Page Login



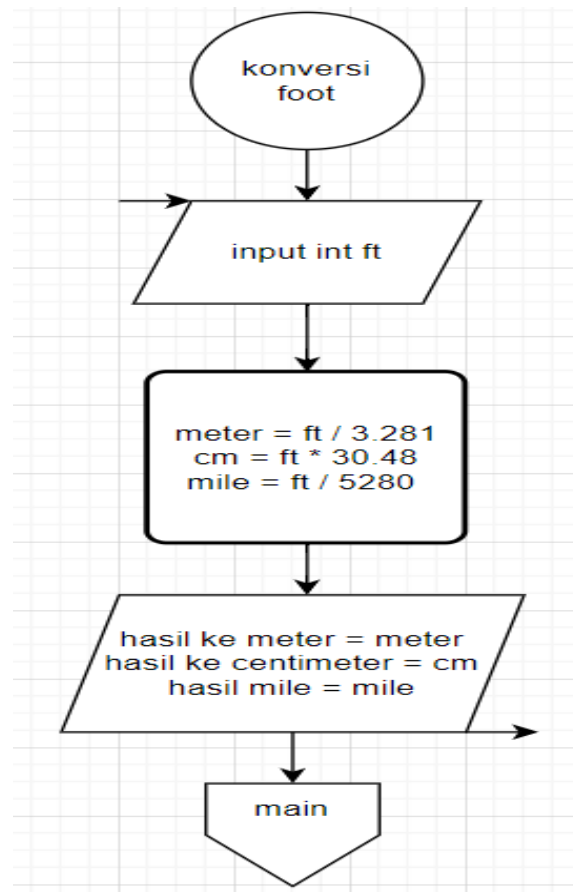
Gambar 1. 4 On-Page Konversi Meter



Gambar 1. 3 On-Page Konversi CentiMeter



Gambar 1. 6 On-Page Konversi Mile



Gambar 1. 5 On-Page Konversi Foot

2. Analisis Program

2.1 Deskripsi Singkat Program

Program ini dirancang untuk melakukan konversi satuan panjang antara meter, centimeter, mile, dan foot. Pengguna harus melakukan login dengan memasukkan nama dan NIM, dengan batas maksimal tiga kali percobaan. Setelah berhasil login, pengguna dapat memilih jenis konversi yang diinginkan dan memasukkan nilai yang akan dikonversi. Program akan terus berjalan hingga pengguna memilih untuk keluar. Program ini bermanfaat untuk mempermudah perhitungan konversi satuan panjang secara cepat dan akurat.

2.2 Penjelasan Alur & Algoritma

Alur Program

- Memasukkan **Nama (username)** dan **NIM (password)**.
- Jika salah 3 kali, program berhenti.
- Pilih jenis konversi satuan panjang.
- Masukkan nilai yang akan dikonversi.
- Program menghitung hasil konversi sesuai rumus.
- Hasil ditampilkan ke pengguna.
- Pengguna dapat memilih untuk melakukan konversi lagi atau keluar dari program.

Algoritma Program

- Inisialisasi attempts = 3
- Selama attempts > 0 lakukan:
 - Minta input Nama dan NIM
 - Jika input benar → lanjut ke menu utama
 - Jika salah → kurangi attempts dan tampilkan pesan kesalahan
- Jika gagal login 3 kali → program berhenti
- Cetak daftar pilihan konversi:
 1. Meter ke (Centimeter, Mile, Foot)
 2. Centimeter ke (Meter, Mile, Foot)
 3. Mile ke (Meter, Centimeter, Foot)
 4. Foot ke (Meter, Centimeter, Mile)
 5. Keluar

2. Minta input pilihan pengguna

- Jika memilih 5 → program berhenti
- Jika memilih selain 5 → Minta input angka yang akan dikonversi

3. Lakukan konversi berdasarkan pilihan

- **Hitung nilai konversi menggunakan rumus:**
 - Meter → Centimeter ($\text{nilai} * 100$), Mile ($\text{nilai} * 0.000621371$), Foot ($\text{nilai} * 3.28084$)
 - Centimeter → Meter ($\text{nilai} / 100$), Mile ($\text{nilai} * 0.0000621371$), Foot ($\text{nilai} * 0.0328084$)
 - Mile → Meter ($\text{nilai} * 1609.34$), Centimeter ($\text{nilai} * 160934$), Foot ($\text{nilai} * 5280$)
 - Foot → Meter ($\text{nilai} * 0.3048$), Centimeter ($\text{nilai} * 30.48$), Mile ($\text{nilai} * 0.000189394$)

4. Tampilkan hasil konversi

5. Tanya apakah pengguna ingin mengulang

- Jika ya → kembali ke langkah 3
- Jika tidak → program berhenti

6. Selesai

3. Source Code

A. Login

Fitur ini digunakan untuk memvalidasi user yang ingin menggunakan aplikasi

Source Code:

```
bool login() {
    string nama = "ayu";
    int nim = 222;
    string username;
    int password;

    for (int i = 0; i < 3; i++) {
        cout << "\n===== SILAHKAN LOGIN =====\n";
        cout << "Masukkan Username: "; cin >> username;
        cout << "Masukkan Password: "; cin >> password;

        if (username == nama && password == nim) {
            return true;
        } else {
            cout << "Salah! Coba lagi.\n";
        }
    }
    return false;
}
```

B. Menu

Terdapat beberapa menu untuk memilih fitur pada aplikasi

Source Code :

```
bool menu() {
    char pilihan;
    cout << " \n===== MENU KONVERSI =====\n";
    cout << "1. meter ke (cm, mile, foot)\n";
    cout << "2. cm ke (m, mile, foot)\n";
    cout << "3. mile ke (m, cm, foot)\n";
    cout << "4. foot ke (m, cm, mile)\n";
    cout << "5. keluar\n";
```

```

cout << "pilih menu :";
cin >> pilihan;
return pilihan;

```

C. Konversi

Fitur ini berfungsi untuk menghitung hasil konversi satuan Meter, CentiMeter, Mile, dan Foot

Source Code:

```

bool KonversiMeter() {
    double meter;
    cout << " \n===== KONVERSI METER =====\n";
    cout << "masukkan nilai meter :";
    cin >> meter ;
    cout << "hasil centimeter = " << meter * 100 << "cm\n" ;
    cout << "hasil mile = " << meter / 1609 << "mile\n";
    cout << "hasil foot = " << meter * 3.281 << "foot\n" ;
    return true ;
}

bool KonversiCenti() {
    double cm;
    cout << " \n===== KONVERSI CENTIMETER =====\n";
    cout << "masukkan nilai centimeter : "; cin >> cm ;
    cout << "hasil meter = " << cm / 100 << "cm\n" ;
    cout << "hasil mile = " << cm / 160900 << "mile\n";
    cout << "hasil foot = " << cm / 30.48 << "foot\n";
    return true;
}

bool KonversiMile() {
    double mile;
    cout << " \n===== KONVERSI MILE =====\n";
    cout << "masukkan nilai mile :"; cin >> mile;
    cout << "hasil meter = " << mile * 1609.34 << "meter\n";
    cout << "hasil Centi meter = " << mile * 160934.4 << "cm\n";
    cout << "hasil foot = " << mile * 5280 << "foot\n";
    return true;
}

```

```
bool KonversiFoot() {  
    double ft;  
    cout << " \n===== KONVERSI FOOT =====\n";  
    cout << "masukkan nilai foot = "; cin >> ft ;  
    cout << "hasil meter = " << ft / 3.281 << "meter\n";  
    cout << "hasil centimeter = " << ft * 30.48 << "cm\n";  
    cout << "hasil mile = " << ft / 5280 << "mile\n";  
    return true;  
}
```


4. Uji Coba dan Hasil Output

4.1 Uji Coba

Pada tahap uji coba ini, program diuji dengan beberapa skenario untuk memastikan bahwa setiap fitur berjalan dengan baik. Uji coba dilakukan dengan berbagai jenis input, termasuk input yang valid dan tidak valid, untuk melihat bagaimana program menangani berbagai kemungkinan kasus.

1. Skenario 1 : Login Berhasil dan melakukan Konversi

Tujuan skenario ini Menguji apakah pengguna dapat login dengan benar dan melakukan konversi satuan panjang.

Langkah Pengujian:

1. Jalankan program.
2. Masukkan **Nama** dan **NIM** yang benar.
3. Pilih menu konversi, misalnya **Meter ke Centimeter**.
4. Masukkan nilai **20 meter**.
5. Program mengonversi **20 meter ke centimeter** dan menampilkan hasilnya.
6. Pilih opsi untuk mengulang atau keluar.
7. Jika memilih mengulang, ulangi konversi lainnya. Jika memilih keluar, program berhenti.

Hasil yang Diharapkan:

- Jika login berhasil, program melanjutkan ke menu utama.
- Konversi dilakukan dengan perhitungan yang benar.
- Program tetap berjalan hingga pengguna memilih keluar.

2. Skenario 2 : Gagal login setelah 3 kali percobaan

Tujuan skenario ini menguji apakah program akan berhenti jika pengguna salah memasukkan Nama dan NIM sebanyak tiga kali.

Langkah Pengujian:

1. Jalankan program.
2. Masukkan **Nama atau NIM yang salah** pada percobaan pertama.
3. Ulangi dengan input yang salah pada percobaan kedua.
4. Ulangi lagi dengan input yang salah pada percobaan ketiga.
5. Program menampilkan pesan bahwa login gagal dan **program berhenti**.

Hasil yang Diharapkan:

- Jika pengguna salah login tiga kali, program langsung berhenti.
- Tidak ada akses ke menu utama setelah gagal login tiga kali.

4.2 Hasil Output

```
===== SILAHKAN LOGIN =====
Masukkan Username: ayu
Masukkan Password: 222

===== SELAMAT ANDA BERHASIL LOGIN =====

===== MENU KONVERSI =====
1. meter ke (cm, mile, foot)
2. cm ke (m, mile, foot)
3. mile ke (m, cm, foot)
4. foot ke (m, cm, mile)
5. keluar
pilih menu : █
```

Gambar 4. 2 login_berhasil & Menu_konversi

```
===== SILAHKAN LOGIN =====
Masukkan Username: ayu
Masukkan Password: 233
Salah! Coba lagi.

===== SILAHKAN LOGIN =====
Masukkan Username: ayu
Masukkan Password: 554
Salah! Coba lagi.

===== SILAHKAN LOGIN =====
Masukkan Username: ayu
Masukkan Password: 443
Salah! Coba lagi.
anda sudah 3 kali salah , silahkan coba lagi.
```

Gambar 4. 1 Gagal_login

```
===== KONVERSI METER =====  
masukkan nilai meter :20  
hasil centimeter = 2000cm  
hasil mile = 0.0124301mile  
hasil foot = 65.62foot
```

Gambar 4. 3 konversi_meter

```
===== KONVERSI CENTIMETER =====  
masukkan nilai centimeter : 20  
hasil meter = 0.2cm  
hasil mile = 0.000124301mile  
hasil foot = 0.656168foot
```

Gambar 4. 4 konversi_centimeter

```
===== KONVERSI FOOT =====  
masukkan nilai foot = 20  
hasil meter = 6.0957meter  
hasil centimeter = 609.6cm  
hasil mile = 0.00378788mile
```

Gambar 4. 5 konversi_Foot

```
===== KONVERSI MILE =====  
masukkan nilai mile :20  
hasil meter = 32186.8meter  
hasil Centi meter = 3.21869e+006cm  
hasil foot = 105600foot
```

Gambar 4. 6 konversi_mile

```
===== MENU KONVERSI =====  
1. meter ke (cm, mile, foot)  
2. cm ke (m, mile, foot)  
3. mile ke (m, cm, foot)  
4. foot ke (m, cm, mile)  
5. keluar  
pilih menu :5  
keluar  
PS C:\Users\ASUS GK\OneDrive\Dokumen\praktikum-apl\post-test\post-test1-1\output>
```

Gambar 4. 7 menu_keluar

5. Langkah-Langkah GIT

1. Git Init

Git init adalah perintah Git yang digunakan untuk menginisialisasi repository Git baru dalam sebuah folder. Cukup ketik “git init” pada terminal vscode

```
PS C:\Users\ASUS GK\OneDrive\Dokumen\praktikum-apl> git init
Initialized empty Git repository in C:/Users/ASUS GK/OneDrive/Dokumen/praktikum-apl/.git/
PS C:\Users\ASUS GK\OneDrive\Dokumen\praktikum-apl> █
```

Gambar 5. 1 git_init

2. Git add

Git add adalah perintah Git yang digunakan untuk menambahkan perubahan pada file ke staging area sebelum dilakukan commit. Perintah ini memungkinkan Git mengetahui file mana saja yang akan dimasukkan dalam commit berikutnya

```
PS C:\Users\ASUS GK\OneDrive\Dokumen\praktikum-apl> git add .
PS C:\Users\ASUS GK\OneDrive\Dokumen\praktikum-apl> █
```

Gambar 5. 2 Git_add

3. Git commit

Git commit adalah perintah Git yang digunakan untuk menyimpan perubahan yang telah ditambahkan ke staging area ke dalam repository. Setiap commit akan memiliki hash unik, pesan commit, dan menyimpan snapshot dari perubahan yang dilakukan.

```
PS C:\Users\ASUS GK\OneDrive\Dokumen\praktikum-apl> git commit -m "finish Post Test 1"
[master 0ecc1ec] finish Post Test 1
1 file changed, 0 insertions(+), 0 deletions(-)
delete mode 100644 post-test/post-test1-1/output/2409106022-AyuAzzahrahAlwi-PT-1.exe
PS C:\Users\ASUS GK\OneDrive\Dokumen\praktikum-apl> █
```

Gambar 5. 3 Git_Commit

4. Git remote

Git remote adalah perintah Git yang digunakan untuk menghubungkan repository lokal dengan repository remote (misalnya di GitHub, GitLab, atau Bitbucket). Dengan perintah ini, kita bisa mengelola koneksi ke repository jarak jauh, memungkinkan push, pull, dan fetch dari repository tersebut.

```
PS C:\Users\ASUS GK\OneDrive\Dokumen\praktikum-apl> git remote add origin https://github.com/ayuaa13/praktikum-apl.git
PS C:\Users\ASUS GK\OneDrive\Dokumen\praktikum-apl> █
```

Gambar 5. 4 Git_remote

5. Git Push

Git push adalah perintah Git yang digunakan untuk mengirim (mengunggah) perubahan dari repository lokal ke repository remote (seperti GitHub, GitLab, atau Bitbucket). Perintah ini memastikan bahwa perubahan yang sudah dikomit di lokal tersedia di repository jarak jauh sehingga bisa diakses oleh orang lain atau untuk cadangan.

```
PS C:\Users\ASUS GK\OneDrive\Dokumen\praktikum-ap1> git push -u origin main
```

Gambar 5. 5 Git_Push