

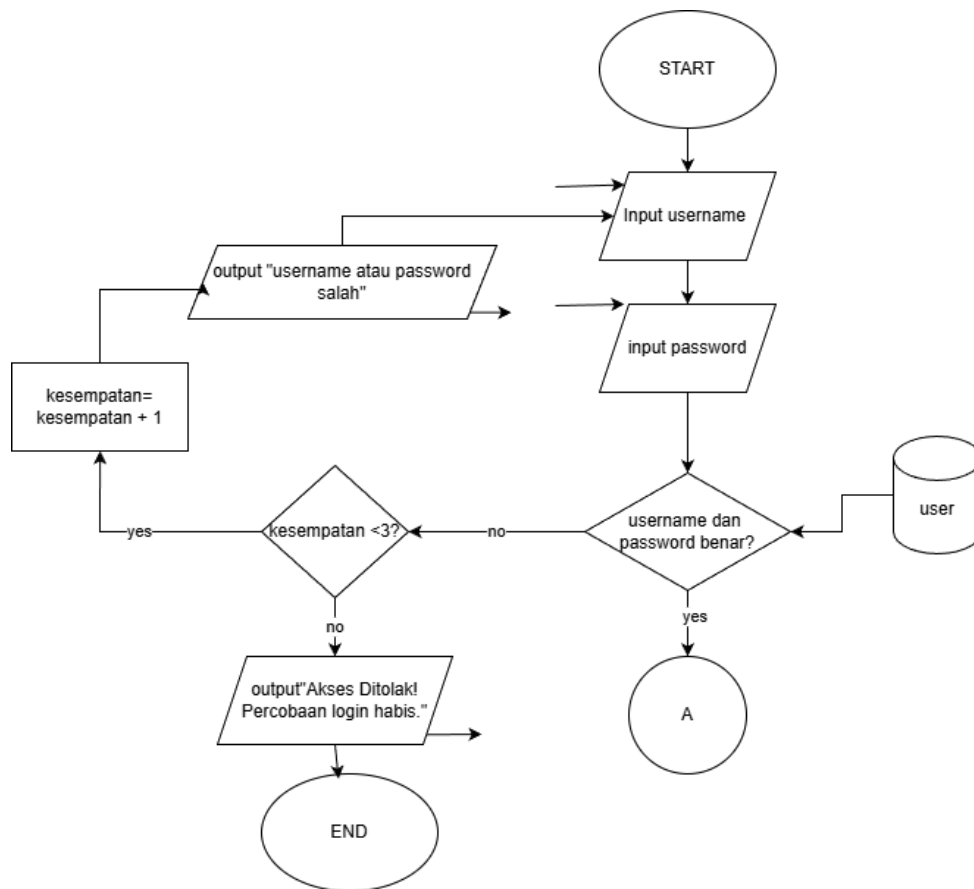
**LAPORAN PRAKTIKUM**  
**POSTTEST 1**  
**ALGORITMA PEMROGRAMAN LANJUT**



**Disusun oleh:**  
**K DESTIANA (2509106001)**  
**Kelas (A1 '25)**

**PROGRAM STUDI INFORMATIKA**  
**UNIVERSITAS MULAWARMAN**  
**SAMARINDA**  
**2026**

## 1. Flowchart



*Gambar 1.1 Flowchart awal program dan login*

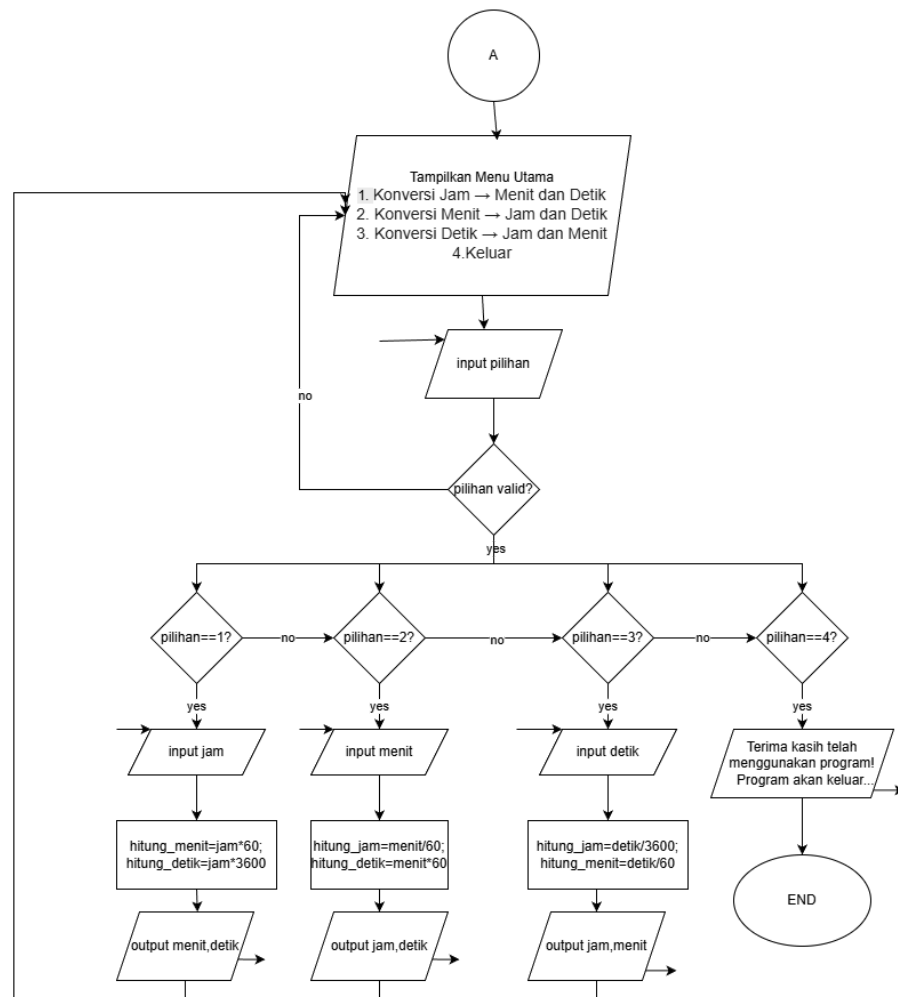
Program dimulai dari simbol Terminal START yang menandakan awal eksekusi program. Setelah itu, program akan melanjutkan ke proses input. Program menampilkan permintaan input username dan password melalui simbol Input/Output berbentuk jajaran genjang. Pengguna memasukkan nilai username dan password yang kemudian disimpan ke dalam variabel namaInput dan passwordInput menggunakan perintah cin,

Setelah input diterima, program masuk ke simbol Decision untuk memvalidasi apakah namaInput sama dengan USER\_DEFAULT ("desti") dan passwordInput sama dengan PASS\_DEFAULT ("001"). Proses validasi ini menggunakan operator perbandingan == dan operator logika &&.

Jika hasil validasi bernilai ya (Yes), artinya login berhasil, maka program akan menampilkan pesan "Login Berhasil! Selamat datang!" melalui simbol Input/Output, lalu melanjutkan ke connector A yang mengarah ke bagian menu utama.

Namun, jika hasil validasi bernilai tidak (No), artinya login gagal, maka program masuk ke simbol Process untuk menambah nilai counter dengan percobaan++. Setelah itu, program menampilkan pesan error "Username atau Password salah!" dan sisa percobaan melalui simbol Input/Output, lalu alur kembali ke keputusan percobaan < 3 untuk mengulangi proses login.

Selanjutnya, program masuk ke simbol Decision untuk memeriksa kondisi percobaan  $< 3$ . Jika kondisi bernilai tidak (No), artinya pengguna sudah gagal login sebanyak 3 kali, maka program akan menampilkan pesan "Akses Ditolak! Percobaan login habis." melalui simbol Input/Output, lalu program berakhir pada simbol Terminal END.



*Gambar 1.2 Flowchart menu utama konversi*

Setelah login berhasil, program masuk ke connector A yang menghubungkan ke bagian menu utama. Program menampilkan menu pilihan melalui simbol Input/Output yang berisi empat opsi: (1) Konversi Jam ke Menit & Detik, (2) Konversi Menit ke Jam & Detik, (3) Konversi Detik ke Jam & Menit, dan (4) Keluar.

Selanjutnya, program meminta pengguna memasukkan pilihan menu melalui simbol Input/Output, lalu masuk ke serangkaian simbol Decision untuk memeriksa nilai pilihan tersebut. Pertama, program memeriksa apakah pilihan  $= 4$ . Jika ya, program menampilkan pesan "Terima kasih telah menggunakan program!" dan berakhir pada simbol Terminal END.

Jika pilihan  $\neq 4$ , program melanjutkan pemeriksaan ke pilihan  $= 1$ . Jika ya, program masuk ke proses konversi Jam: meminta input nilai jam, menghitung menit = jam \* 60 dan detik = jam \* 3600 menggunakan operator aritmatika perkalian sesuai MODUL 1 Hal. 9, lalu menampilkan hasil melalui simbol Input/Output. Setelah itu, alur kembali ke tampilan menu.

Jika pilihan  $\neq 1$ , program memeriksa pilihan  $= 2$ . Jika ya, program melakukan konversi Menit: meminta input menit, menghitung jam = menit / 60, sisa\_menit = menit % 60, dan detik = menit \* 60 menggunakan operator pembagian dan modulus sesuai MODUL 1 Hal. 9-10, lalu menampilkan hasil dan kembali ke menu.

Jika pilihan  $\neq 2$ , program memeriksa pilihan  $= 3$ . Jika ya, program melakukan konversi Detik: meminta input detik, menghitung jam = detik / 3600, sisa = detik % 3600, menit = sisa / 60, dan detik = sisa % 60, lalu menampilkan hasil dan kembali ke menu.

Jika semua kondisi pilihan  $= 1/2/3/4$  bernilai tidak, maka program masuk ke simbol Input/Output untuk menampilkan pesan "Pilihan tidak valid! Masukkan 1-4.", lalu kembali ke tampilan menu.

## 2. Deskripsi Singkat Program

Program ini dirancang untuk mengkonversi satuan waktu (Jam, Menit, dan Detik). Program dilengkapi dengan sistem keamanan sederhana berupa login username dan password sebelum pengguna dapat mengakses menu. Diberikan batas maksimal 3 kali percobaan. Jika gagal sebanyak 3 kali, program akan otomatis berhenti. dengan menu konversi waktu:

- Opsi 1: Konversi Jam  $\rightarrow$  Menit dan Detik.
- Opsi 2: Konversi Menit  $\rightarrow$  Jam dan Detik.
- Opsi 3: Konversi Detik  $\rightarrow$  Jam dan Menit.
- Opsi 4: Keluar dari program.

## 3. Source Code

- SISTEM LOGIN (3 Percobaan)

```
const int BATAS LOGIN = 3;
```

```

const string USER_DEFAULT = "desti";
const string PASS_DEFAULT = "001";

bool prosesLogin() {
    string nama Input, password Input;
    int percobaan = 0;
    bool loginBerhasil = false;

    while (percobaan < BATAS LOGIN) {
        cout << "Masukkan Username : ";
        cin >> namaInput;
        cout << "Masukkan Password : ";
        cin >> passwordInput;

        if (namaInput == USER_DEFAULT && passwordInput == PASS_DEFAULT) {
            loginBerhasil = true;
            break;
        } else {
            percobaan;
            if (percobaan < BATAS LOGIN) {
                cout << "Username atau Password salah!" << endl;
                cout << "Sisa percobaan: " << (BATAS_LOGIN - percobaan) << endl;
            }
        }
    }
    return loginBerhasil;
}

```

- konversi waktu: Jam → Menit & Detik

```

void konversiJamKeMenitDetik() {
    int inputJam, outputMenit, outputDetik;

    cout << "Masukkan jumlah Jam : ";
    cin >> inputJam;

    outputMenit = inputJam * 60;
    outputDetik = inputJam * 3600;

    cout << inputJam << " Jam = " << outputMenit << " Menit" << endl;
    cout << inputJam << " Jam = " << outputDetik << " Detik" << endl;
}

```

#### 4. Hasil Output

```
=====
PROGRAM KONVERSI WAKTU SEDERHANA
=====

--- LOGIN ---
Masukkan Username : desta
Masukkan Password : 002
Username atau Password salah!
Sisa percobaan: 2

--- LOGIN ---
Masukkan Username : esti
Masukkan Password : 002
Username atau Password salah!
Sisa percobaan: 1

--- LOGIN ---
Masukkan Username : desta
Masukkan Password : 003

Akses Ditolak! Percobaan login habis.
Program akan berhenti...
```

*Gambar 4.1 output login gagal sebanyak 3 kali*

```
=====
PROGRAM KONVERSI WAKTU SEDERHANA
=====

--- LOGIN ---
Masukkan Username : desti
Masukkan Password : 001

Login Berhasil! Selamat datang, desti!
```

*Gambar 4.2 output login berhasil*

```
=====
MENU UTAMA
=====
1. Konversi Jam -> Menit dan Detik
2. Konversi Menit -> Jam dan Detik
3. Konversi Detik -> Jam dan Menit
4. Keluar
=====
Masukkan pilihan menu (1-4): █
```

*Gambar 4.3 output menu utama*

```
Masukkan pilihan menu (1-4): 1

--- Konversi: Jam -> Menit & Detik ---
Masukkan jumlah Jam : 2

>>> HASIL KONVERSI <<<
2 Jam = 120 Menit
2 Jam = 7200 Detik
```

*Gambar 4.5 output pilihan 1*

```
Masukkan pilihan menu (1-4): 2

--- Konversi: Menit -> Jam & Detik ---
Masukkan jumlah Menit : 3

>>> HASIL KONVERSI <<<
3 Menit = 0 Jam 3 Menit
3 Menit = 180 Detik
```

*Gambar 4.6 output pilihan 2*

```
Masukkan pilihan menu (1-4): 3

--- Konversi: Detik -> Jam & Menit ---
Masukkan jumlah Detik : 3600

>>> HASIL KONVERSI <<<
3600 Detik = 1 Jam 0 Menit 0 Detik
```

*Gambar 4.7 output pilihan 3*

```
Masukkan pilihan menu (1-4): 4

=====

Terima kasih telah menggunakan program!
Program akan keluar...

=====
```

*Gambar 4.8 output pilihan 4*

## 5. Langkah-langkah GIT

### 5.1 GIT Add

```
PS C:\Users\LENOVO\OneDrive\Desktop\praktikum-apl\praktikum-apl\post-test\post-test-1> git add .
PS C:\Users\LENOVO\OneDrive\Desktop\praktikum-apl\praktikum-apl\post-test\post-test-1> git status
On branch main

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
        new file:   2509106001-KDESTIANA-PT-1.cpp
        new file:   2509106001-KDESTIANA-PT-1.exe
```

Berfungsi untuk menandai file yang akan dimasukan ke commit.

### 5.2 GIT Commit

```
PS C:\Users\LENOVO\OneDrive\Desktop\praktikum-apl\praktikum-apl\post-test\post-test-1> git commit -m "Finish Post Test 1"
[main (root-commit) 7d41ccb] Finish Post Test 1
 2 files changed, 140 insertions(+)
 create mode 100644 post-test/post-test-1/2509106001-KDESTIANA-PT-1.cpp
 create mode 100644 post-test/post-test-1/2509106001-KDESTIANA-PT-1.exe
```

Berfungsi menyimpan atau stage yang telah berubah kedalam riwayat versi repository, serta pesan “,”.

### 5.3 GIT Push

```
PS C:\Users\LENOVO\OneDrive\Desktop\praktikum-apl\praktikum-apl\post-test\post-test-1> git add .
PS C:\Users\LENOVO\OneDrive\Desktop\praktikum-apl\praktikum-apl\post-test\post-test-1> git status
On branch main

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
        new file:   2509106001-KDESTIANA-PT-1.cpp
        new file:   2509106001-KDESTIANA-PT-1.exe
```

Berfungsi mengirim atau mengupload commit yang ada di lokal repository remote, agar tersimpan di github.