

## **LAPORAN TUGAS AKHIR**

### **PERANCANGAN DAN IMPLEMENTASI PROGRAM PRESENSI DAN PENILAIAN MAHASISWA BERBASIS *COMMAND LINE INTERFACE (CLI)* MENGGUNAKAN BAHASA PYTHON**

**Dosen Pengampu: Ni Luh Putu Ika Candrawengi S.Stat., M.Stat**



**Oleh:**

**LADHISYA IZYAFA FAIQATUZHNI**

**(42530032)**

**ZINTIA ADELLA**

**(42530036)**

**PROGRAM STUDI TEKNOLOGI INFORMASI  
FAKULTAS TEKNIK DAN INFORMATIKA  
UNIVERSITAS PENDIDIKAN NASIONAL**

**2025**

## 1. Penjelasan Struktur Program

### A) Pembagian Fungsi

Program dibagi menjadi beberapa fungsi seperti *tambah\_mahasiswa()*, *input\_nilai()*, *input\_presensi()*, *tampilkan\_data()*, *save\_data()*, dan *load\_data()*.

Pembagian fungsi dilakukan karena:

- Satu fungsi hanya mengerjakan satu tugas (misalnya: hanya input nilai, atau hanya simpan data).
- Mencegah kode menjadi terlalu panjang dan sulit dibaca.
- Memudahkan pengembangan dan perawatan. Jika terjadi kesalahan pada fitur input nilai, kita hanya perlu memperbaiki pada fungsi tersebut, bukan seluruh program.
- Menu navigasi lebih jelas, karena tiap pilihan menu memanggil satu fungsi. Seperti: Menu → pilih 1 → panggil *tambah\_mahasiswa()*

### B) Struktur Class Mahasiswa

Class Mahasiswa bertujuan untuk menyimpan data setiap mahasiswa dalam satu wadah. Atribut dalam class yaitu **NIM, Nama, Tugas, UTS, UAS, Hadir**, dan ***total\_pertemuan***.

Method dalam class:

- *nilai\_akhir()* untuk menghitung skor akhir mahasiswa.
- *persentase\_hadir()* untuk menghitung persentase kehadiran.

Alasan menggunakan Class:

- Data tiap mahasiswa saling berkaitan dan lebih rapi disatukan dalam satu objek.
- Jika hanya memakai variabel bebas, program akan menjadi berantakan.
- Penggunaan class mendukung prinsip OOP: mengelompokkan data + perilaku.

## 2. Penjelasan Logika Khusus

### A) Advance Function (Lambda)

Di dalam method *nilai\_akhir()* digunakan lambda:

*hitung = lambda t, u1, u2: 0.3\*t + 0.35\*u1 + 0.35\*u2*

```
def nilai_akhir(self):  
    hitung = lambda t, u1, u2: 0.3*t + 0.35*u1 + 0.35*u2  
    return hitung(self.tugas, self.uts, self.uas)
```

Alasan memakai lambda:

- Operasi hanya satu baris sehingga tidak perlu fungsi terpisah dengan def.

- Lambda lebih sederhana untuk perhitungan matematis.
- Kode menjadi ringkas namun tetap jelas.

Lambda lain dipakai untuk konversi nilai menjadi grade:

*konversi\_grade = lambda n: ...*

```
konversi_grade = lambda n: (
    "A" if n >= 85 else
    "B" if n >= 70 else
    "C" if n >= 55 else
    "D" if n >= 40 else "E"
)
```

Lambda dipakai karena :

- Fungsi konversi nilai ke grade sederhana, satu baris, dan langsung dipakai.
- Ini membuat kode lebih ringkas dan mudah dibaca tanpa membuat fungsi terpisah.

Jadi program memakai lambda untuk perhitungan nilai akhir dan penentuan grade. Keduanya termasuk kategori advance function.

## B) Logik Modulus pada Presensi

Dalam fitur presensi:

*jenis = "Teori" if pertemuan % 2 == 1 else "Praktikum"*

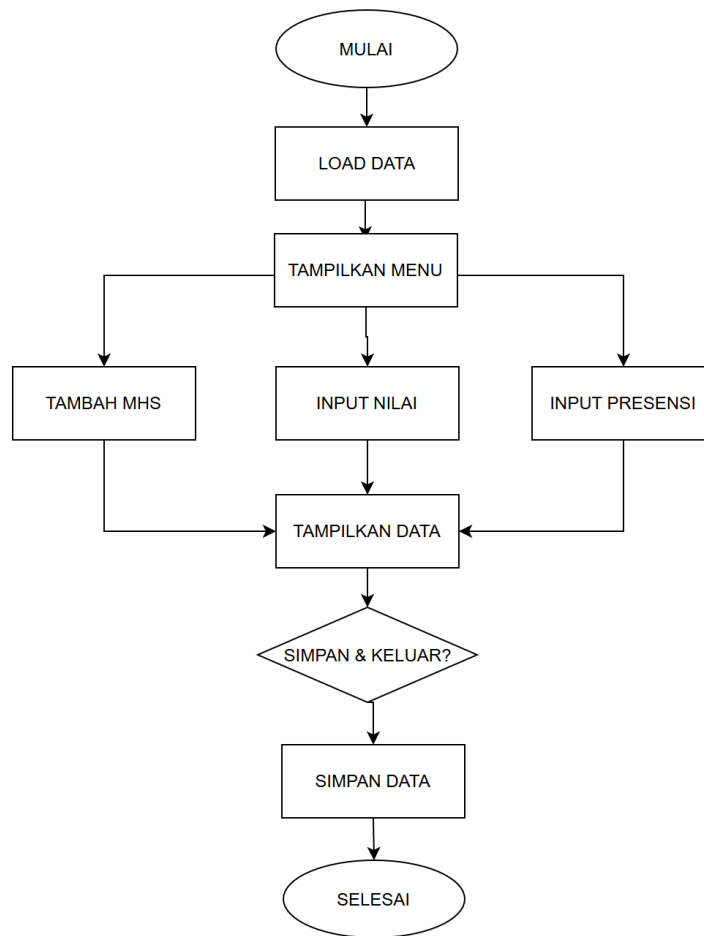
Logika modulus akan bekerja seperti ini:

- jika pertemuan ganjil  $\rightarrow \% 2 == 1$  maka akan masuk ke jenis teori
- jika pertemuan genap  $\rightarrow \% 2 == 0$  maka akan masuk ke jenis praktikum

Alasannya:

- Program tidak perlu menanyakan jenis pertemuan manual.
- Cukup input nomor pertemuan saja.
- Sistem otomatis menentukan teori atau praktikum.

## 3. Flowchart



#### 4. Screenshot Output

```

PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

1
PS C:\Users\Ladhisya\PRESENSI> & C:/Users/Ladhisya/AppData/Local/Programs/Python/Python314/python.exe "c:/CODINGS/uas_pmr/mahasiswa.py"

===== MENU =====
1. Tambah Mahasiswa
2. Input Nilai
3. Input Presensi
4. Tampilkan Data
5. Simpan & Keluar

Pilih menu: 1
Masukkan NIM: 42530070
Masukkan Nama: Keano Zachary
Mahasiswa berhasil ditambahkan.

PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

===== MENU =====
1. Tambah Mahasiswa
2. Input Nilai
3. Input Presensi
4. Tampilkan Data
5. Simpan & Keluar

Pilih menu: 2
Masukkan NIM: 42530070
Nilai Tugas (0-100): 80
Nilai UTS (0-100): 89
Nilai UAS (0-100): 96
Nilai berhasil diinput.
  
```

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

===== MENU =====
1. Tambah Mahasiswa
2. Input Nilai
3. Input Presensi
4. Tampilkan Data
5. Simpan & Keluar

Pilih menu: 3
Pertemuan ke-: 1
Jenis pertemuan: Teori
Keano Zachary (Hadir/Alpha/Izin): Hadir

PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

===== MENU =====
1. Tambah Mahasiswa
2. Input Nilai
3. Input Presensi
4. Tampilkan Data
5. Simpan & Keluar

Pilih menu: 3
Pertemuan ke-: 2
Jenis pertemuan: Praktikum
Keano Zachary (Hadir/Alpha/Izin): Izin

PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

1. Tambah Mahasiswa
2. Input Nilai
3. Input Presensi
4. Tampilkan Data
5. Simpan & Keluar

Pilih menu: 4

===== DATA MAHASISWA =====
NIM | Nama | Nilai Akhir | Grade | Kehadiran
-----
42530070 | Keano Zachary | 88.75 | A | 50.00%
=====

PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

NIM | Nama | Nilai Akhir | Grade | Kehadiran
-----
42530070 | Keano Zachary | 88.75 | A | 50.00%
=====

===== MENU =====
1. Tambah Mahasiswa
2. Input Nilai
3. Input Presensi
4. Tampilkan Data
5. Simpan & Keluar

Pilih menu: 5
Data disimpan. Program selesai.
PS C:\Users\Ladhisva\PRESENST>
```

- **Contoh output di Excel:**

