

**LAPORAN PRAKTIKUM**  
**POSTTEST 4**  
**ALGORITMA PEMROGRAMAN DASAR**

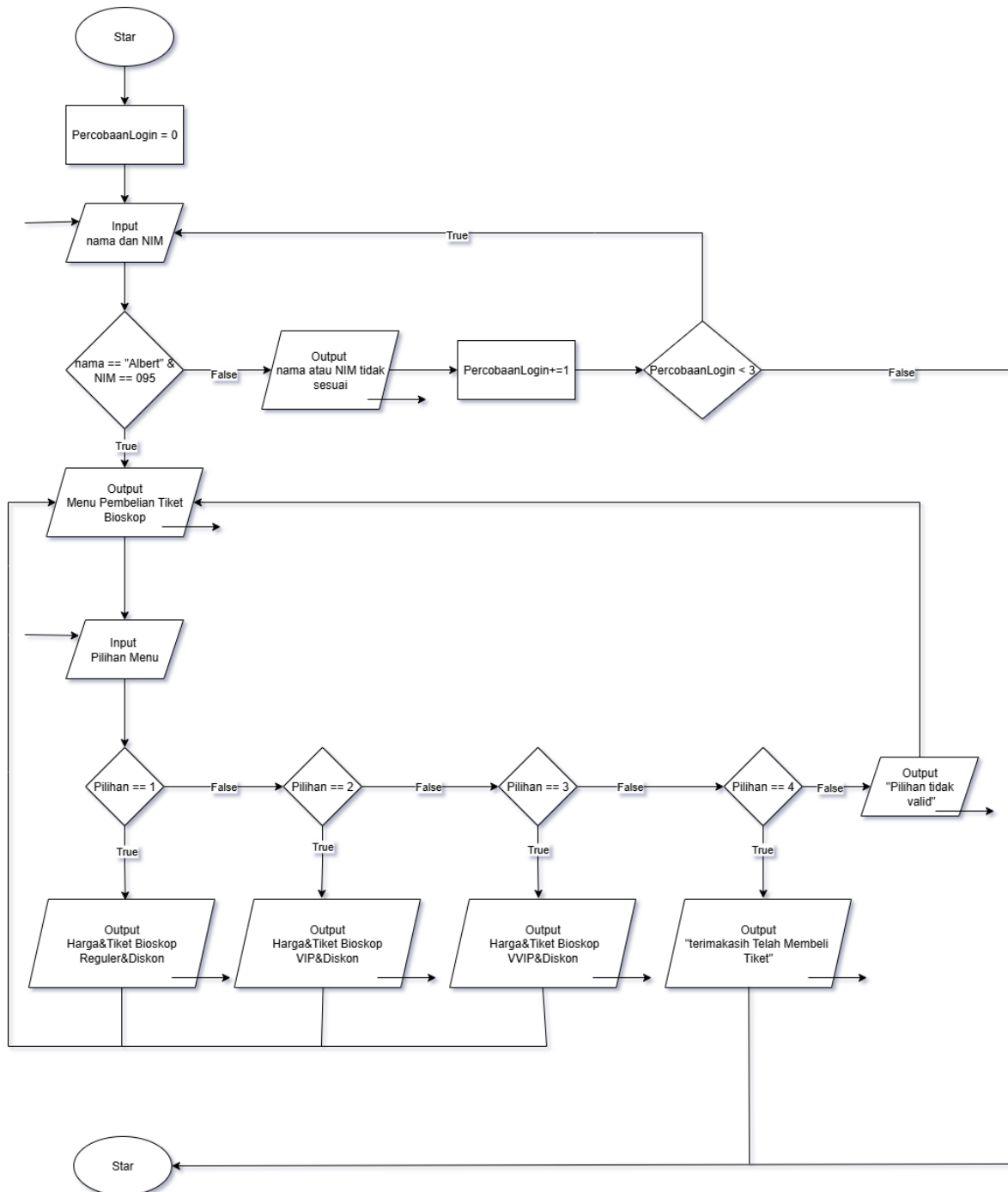


**Disusun oleh:**  
**ALBERT EINSTEIN LIEM (2509106095)**  
**Kelas (C1 '25)**

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

## 1. Flowchart Pembelian Tiket Bioskop

Pertama kita tambahkan simbol *Process* untuk Login kemudian masukan *Input Username* dan *Password* “nama dan NIM” lalu *Decision* di bagian *False* jika nama dan NIM tidak sesuai ia akan mengulang nya sampai 3 kali jika lebih dari itu anda akan otomatis *Exit*, di bagian *True* jika nama dan NIM sesuai, ia akan menampilkan pilihan tiket yang ada dan jika pelanggan memilih tiket 1-3 programnya akan terulang terus sampai kita memilih pilihan ke 4 program nya akan otomatis berhenti.



Gambar 1.1 Flowchart

## 2. Deskripsi Singkat Program

Agar pelanggan dapat melihat diskon dan harga yang harus di bayar dan memudahkan pelanggan untuk membeli tiket.

## 3. Source Code

- Menyimpan data login (nama & NIM) yang benar.
- Menggunakan perulangan *for* sebanyak 3 kali untuk memberikan kesempatan login.
- Perulangan *while True* agar menu muncul berulang kali sampai pengguna memilih keluar (*break*).
- Jika input tidak termasuk “1”, “2”, atau “3”, maka dianggap salah dan diulang.
- try-except digunakan agar tidak error jika pengguna salah mengetik (misalnya huruf).
- Mengambil nama kategori tiket dan harga berdasarkan pilihan pengguna.

```
nama, NIM = "Albert Einstein Liem", "2509106095"
harga = {"1": ("Reguler", 50000), "2": ("VIP", 100000), "3": ("VVIP", 150000)}

for i in range(3):
    print("== SELAMAT DATANG DI BIOSKOP SCP ==")
    if input("Username: ") == nama and input("Password: ") == NIM:
        print(f"Login berhasil! Selamat datang {nama}"); break
    else:
        print(f"Login gagal! Percobaan ke-{i+1}")
else:
    print("Percobaan habis! Silakan coba lagi."); exit()

while True:
    print("\n=== MENU PEMBELIAN TIKET ===")
    print("1.Reguler(50k) 2.VIP(100k) 3.VVIP(150k) 4.Keluar")
    p = input("Pilih (1-4): ")
    if p == "4": print("Terima kasih!"); break
    if p not in harga: print("Pilihan salah!"); continue

    try:
        jml = int(input("Jumlah tiket: "))
        if jml <= 0: print("Jumlah harus >0"); continue
    except:
        print("Input harus angka!"); continue

    kategori, hrg = harga[p]
    total = jml * hrg
```

Gambar 2.1 Python

```
if total >= 300000:
    pot = total * 0.12; bonus = f"Diskon 12% (Rp {int(pot):,})"
elif total >= 200000:
    pot = total * 0.08; bonus = f"Diskon 8% (Rp {int(pot):,})"
elif total >= 150000:
    pot = 0; bonus = "Bonus Poster Film Eksklusif"
else:
    pot = 0; bonus = "Tidak ada bonus/diskon"

total_bayar = total - pot
print(f"\n=== STRUK PEMBELIAN ===\nJenis: {kategori}\nJumlah: {jml}\nTotal: Rp {total:,}\nBonus: {bonus}\nBayar: Rp {int(total_bayar):,}")
```

Gambar 2.1 python

#### 4. Hasil Output

```
== SELAMAT DATANG DI BIOSKOP SCP ==
Username: Albert Einstein Liem
Password: 2509106095
Login berhasil! Selamat datang, Albert Einstein Liem silahkan pilih tiket anda!
=== MENU PEMBELIAN TIKET BIOSKOP MALL SCP ===
1. Tiket Reguler : Rp 50.000
2. Tiket VIP : Rp 100.000
3. Tiket WVIP : Rp 150.000
4. Keluar
Pilih jenis tiket (1-4): 3
Jumlah tiket: 1

=== HASIL PEMBELIAN TIKET ANDA ===
Jenis Tiket : WVIP
Jumlah Tiket: 1
Total Harga : Rp 150,000
Bonus/Promo : Bonus Poster Film Eksklusif
Total bayar anda : Rp 150,000
=== MENU PEMBELIAN TIKET BIOSKOP MALL SCP ===
1. Tiket Reguler : Rp 50.000
2. Tiket VIP : Rp 100.000
3. Tiket WVIP : Rp 150.000
4. Keluar
Pilih jenis tiket (1-4): 4
Terima kasih!
```

Gambar 3.1 Hasil

```
== SELAMAT DATANG DI BIOSKOP SCP ==
Username: Albert Einstein Liem
Password: 25091060925
Login gagal! nama dan NIM anda tidak sesuai, percobaan ke - 1
== SELAMAT DATANG DI BIOSKOP SCP ==
Username: albert
Password: 209373645201
Login gagal! nama dan NIM anda tidak sesuai, percobaan ke - 2
== SELAMAT DATANG DI BIOSKOP SCP ==
Username: albert einstein
Password: 162534171092
Login gagal! nama dan NIM anda tidak sesuai, percobaan ke - 3
Percobaan habis! silahkan coba beberapa saat lagi!
```

Gambar 3.2 Hasil

## 5. Langkah-langkah GIT

### 5.1 GIT Add

memilih file yang mau dicatat sebelum disimpan.

```
PS C:\Kenzyy\kuliah\github\praktikum-apd\Post-Test\Post-test-apd-4> git add .
```

### 5.2 GIT Commit

menyimpan perubahan itu secara permanen di *repository lokal* (di komputer kita), lengkap dengan pesan.

```
PS C:\Kenzyy\kuliah\github\praktikum-apd\Post-Test\Post-test-apd-4> git commit -m "posttest 4"
[main 51e7e07] posttest 4
1 file changed, 103 insertions(+)
create mode 100644 Post-Test/Post-test-apd-4/25091060-AlbertEinsteinLiem-PT-4.py
```

### 5.3 GIT Push

mengirim commit dari komputer kita ke *repository GitHub* supaya tersimpan online dan bisa dilihat orang lain.

```
PS C:\Kenzyy\kuliah\github> git push -u origin main
Enumerating objects: 9, done.
Counting objects: 100% (9/9), done.
Delta compression using up to 16 threads
Compressing objects: 100% (5/5), done.
Writing objects: 100% (6/6), 1.41 KiB | 1.41 MiB/s, done.
Total 6 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/alberteinsteinliem-dev/praktikum-apd.git
  539f159..a724ba5  main -> main
branch 'main' set up to track 'origin/main'.
```