

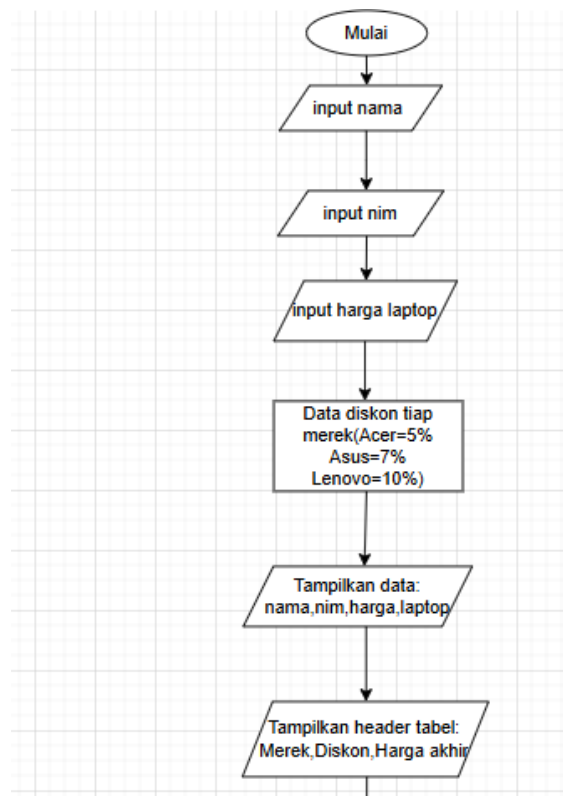
**LAPORAN PRAKTIKUM**  
**POSTTEST 2**  
**ALGORITMA PEMROGRAMAN DASAR**



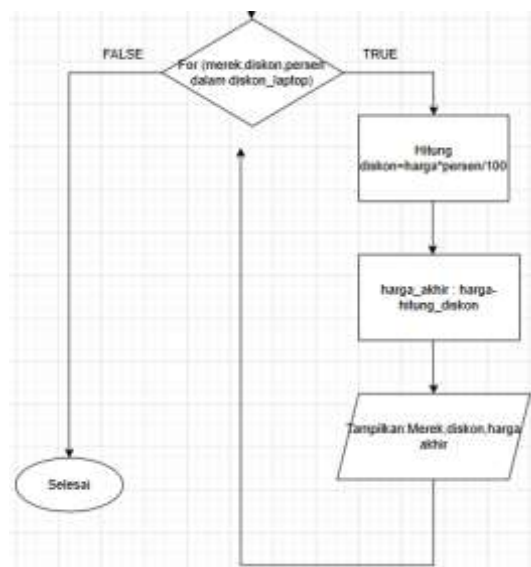
**Disusun oleh:**  
**Meyssa Reguel Sitompul (2509106108)**  
**Kelas (C1 '25)**

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

## 1. Flowchart



Gambar 1.1 Flowchart



Gambar 1.2 Flowchart

Penjelasan Flowchart :

1. Mulai
2. input Nama
3. Input NIM
4. Input Harga Laptop
5. Data diskon tiap merek(Acer=5%,Asus=7%,Lenovo=10%)
6. Tampilkan Data: Nama, NIM, Harga Laptop
7. Tampilkan Header Tabel: Merek, Diskon, Harga Akhir
8. For (Merek, Diskon, Persen Dalam Diskon \_Laptop
9. Hitung Diskon = Hitung Persen/100
10. Harga\_Akhir : Harga -Hitung \_Diskon
11. Tampilkan: Merek, Diskon, Harga Akhir
12. Kembali ke nomor delapan, sampai merek akhir
13. Selesai

## 2. Deskripsi Singkat Program

Program ini menghitung harga laptop setelah diskon berdasarkan merek yang dipilih, lalu menampilkan ringkasan pesanan berisi nama, NIM, merek, dan total harga.

## 3. Source Code

```
#input pilih merek laptop
merek = input("Pilih merek laptop (Acer/Asus/Lenovo):")
# Hitung harga akhir
harga_akhir= harga - (harga*diskon.get(merek,0))
print(f"Harga{merek}setelah diskon:Rp{harga_akhir:,.0f}")
```

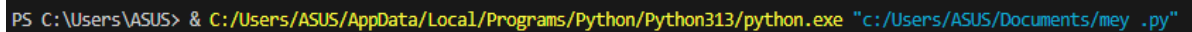
## 4. Hasil Output

```
Masukan Nama Lengkap:mei
Masukan Nim:0000
Hargaaussetelah diskon:Rp12,000,000
=====
PEMBELIAN LEPTOP
=====
NAMA          : mei
NIM           : 0000
MERK          : asus
=====
TOTAL KESELURUHAN : 12000000
=====
PS C:\Users\ASUS> & C:/Users/ASUS/AppData/Local/Programs/Python/Python313/python.exe "c:/Users/ASUS/Documents/mei .py"
Masukan Nama Lengkap:& C:/Users/ASUS/AppData/Local/Programs/Python/Python313/python.exe "c:/Users/ASUS/Documents/mei .py"
Masukan Nim:0000
Pilih merek laptop (Acer/Asus/Lenovo):asus
Hargaaussetelah diskon:Rp12,000,000
```

Gambar 4.1 Output

## 5. Langkah-langkah GIT

### 5.1 GIT Init

A screenshot of a Windows command prompt window. The text displayed is: PS C:\Users\ASUS> & C:/Users/ASUS/AppData/Local/Programs/Python/Python313/python.exe "c:/Users/ASUS/Documents/mey .py". The prompt is green, and the command and its output are white on a black background.

```
PS C:\Users\ASUS> & C:/Users/ASUS/AppData/Local/Programs/Python/Python313/python.exe "c:/Users/ASUS/Documents/mey .py"
```

Gambar 5.5.1 Langkah Git

Perintah git init digunakan untuk membuat sebuah repository Git baru di dalam folder yang dipilih. Dengan kata lain, perintah ini mengubah sebuah folder biasa menjadi folder yang dapat dikelola oleh Git. Setelah menjalankan git init, Git akan membuat sebuah sub-folder tersembunyi bernama .git. Folder inilah yang menyimpan semua informasi penting terkait version control, seperti riwayat perubahan, konfigurasi, dan metadata repository.

Git itu alat buat ngatur versi kerjaan kita, biasanya dipakai sama programmer. Bayangin aja kayak mesin waktu buat code — bisa liat perubahan sebelumnya, balikin file, dan kerja bareng tim.

## 5.2 GIT Add

```
PS C:\partikum-apd\pratikum-apd> git push -u origin main
```

Gambar 5.5.2 Langkah Git

Perintah `git add` digunakan untuk menambahkan semua perubahan file yang ada di dalam folder proyek ke dalam staging area Git. Staging area adalah tempat sementara di mana perubahan file disiapkan sebelum benar-benar disimpan ke dalam riwayat repository melalui perintah `git commit`.

## 5.3 GIT Commit

```
PS D:\Muhammad Fajar\Kuliah\APD\Pratikum\Posstest-2> git commit -m "Memasukkan Ke Github"
[master (root-commit) fafc951] Memasukkan Ke Github
 2 files changed, 68 insertions(+)
 create mode 100644 kelas/pertemuan-2/pertemuan-2.py
 create mode 100644 post-test/post-test-apd-2/2509106117-MuhammadFajar-PT-2.py
```

Gambar 5.5.3 Langkah Git

Commit dalam Git dapat diibaratkan seperti menyimpan catatan atau rekaman atas perubahan yang telah dilakukan pada proyek. `Git commit` berfungsi untuk menyimpan (merekam) snapshot atau perubahan pada kode atau file di repository Git. Saat kamu melakukan commit, Git akan menyimpan semua perubahan yang sudah kamu staging (dimasukkan ke area staging) dalam sebuah commit object.

## 5.4 GIT Remote

```
PS D:\Muhammad Fajar\Kuliah\APD\Pratikum\Posstest-2> git remote add origin https://github.com/Celrik08/praktikum-apd.git
```

Gambar 5.5.4 Langkah Git

Git remote adalah jembatan antara repository lokal kamu dengan repository di server, supaya kamu bisa berbagi kode, berkolaborasi, dan sinkronisasi perubahan dengan tim. Git remote adalah fitur di Git yang berfungsi untuk menghubungkan repository lokal kamu dengan repository yang berada di server atau tempat lain (remote repository), misalnya di GitHub, GitLab, Bitbucket, atau server Git lainnya.

## 5.5 GIT Push

```
PS D:\Muhammad Fajar\Kuliah\APD\Pratikum\Posstest-2> git push -u origin main
Enumerating objects: 8, done.
Counting objects: 100% (8/8), done.
Delta compression using up to 8 threads
Compressing objects: 100% (4/4), done.
Writing objects: 100% (8/8), 1.26 KiB | 184.00 KiB/s, done.
Total 8 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/Celrik08/praktikum-apd.git
 * [new branch]      main -> main
branch 'main' set up to track 'origin/main'.
```

Gambar 5.5.5 Langkah Git

Git push adalah perintah yang digunakan untuk mengirimkan perubahan atau hasil kerja dari komputer Anda (repository lokal) ke penyimpanan Git yang ada di internet atau server (repository remote), seperti GitHub atau GitLab. Setelah Anda melakukan perubahan dan menyimpannya secara lokal, perintah ini berfungsi agar perubahan tersebut dapat tersimpan secara online dan dapat diakses oleh orang lain yang memiliki akses ke repository tersebut.