

**LAPORAN PRAKTIKUM**  
**POSTTEST 3**  
**ALGORITMA PEMROGRAMAN DASAR**



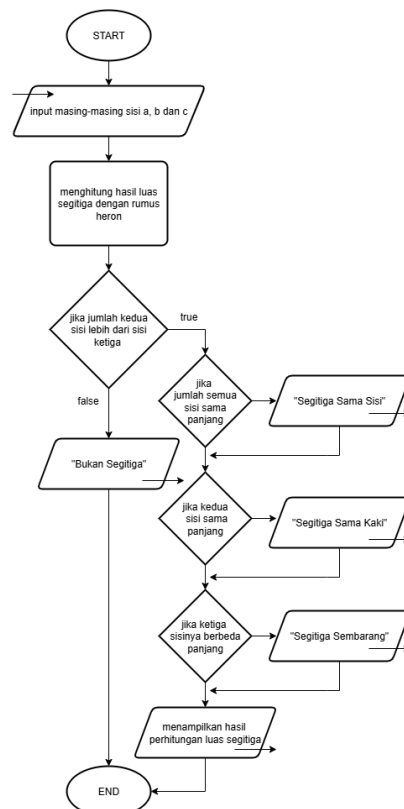
**Disusun oleh:**  
**Muhammad Adrian (2509106032)**  
**Kelas (A2 '25)**

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

## 1. Flowchart

Program ini dibuat untuk mengidentifikasi segitiga dan hasil penjumlahan luasnya setelah menginput masing-masing sisi yaitu a, b, dan c. Adapun alur logika programnya yaitu:

- Seseorang yang menjalankan program akan disuruh memasukkan atau menginput masing-masing sisi yaitu sisi a, b dan c.
- Program akan memproses dan menghitung hasil luas segitiga tersebut yang akan ditampilkan nantinya.
- Jika jumlah kedua sisi lebih dari sisi ketiga maka masuk ke Decision selanjutnya dan jika tidak memenuhi fungsi tersebut, program akan menampilkan “Bukan Segitiga.”
- Dalam Decision percabangan selanjutnya, jika semua sisinya sama panjang maka menampilkan output “Segitiga Sama Sisi”.
- Lalu berikutnya jika hanya dua sisinya yang sama panjang maka menampilkan output “Segitiga Sama Kaki”.
- Terakhir jika ketiga sisinya berbeda panjang maka menampilkan output “Segitiga Sembarang”.
- Juga dengan menampilkan hasil perhitungan luas segitiganya



## 2. Deskripsi Singkat Program

Program ini bertujuan untuk mengidentifikasi segitiga berdasarkan input masing-masing sisi serta menghitung dan menampilkan hasil penjumlahan luas pada segitiga tersebut.

## 3. Source Code

Source Code:

```
print("=== Mengidentifikasi Segitiga dan luasnya ===")
print(" ")
sisi_a = int(input("Masukkan Sisi A: "))
sisi_b = int(input("Masukkan Sisi B: "))
sisi_c = int(input("Masukkan Sisi C: "))

print(" ")
# MENGHITUNG LUAS DENGAN RUMUS HERON
setengah_luas = ((sisi_a + sisi_b) + sisi_c) / 2
luas = (setengah_luas * (setengah_luas - sisi_a) * (setengah_luas - sisi_b)
* (setengah_luas - sisi_c)) ** 0.5

if (sisi_a + sisi_b > sisi_c) and (sisi_b + sisi_c > sisi_a) and (sisi_c +
sisi_a > sisi_b) :
    if sisi_a == sisi_b == sisi_c:
        print("Segitiga Sama Sisi")
    elif (sisi_a == sisi_b) or (sisi_a == sisi_c) or (sisi_b == sisi_c):
        print("Segitiga Sama Kaki")
    elif sisi_a != sisi_b != sisi_c:
        print("Segitiga Sembarang")
    print(f"Hasil luasnya adalah: {luas}")
else:
    print("Bukan Segitiga.")
```

## 4. Hasil Output

```
PS C:\Users\ASUS\Desktop\Praktikum-APD\post-test\post-test-apd-3> py .\main.py
=== Mengidentifikasi Segitiga dan luasnya ===

Masukkan Sisi A: 3
Masukkan Sisi B: 4
Masukkan Sisi C: 5

Segitiga Sembarang
Hasil luasnya adalah: 6.0
PS C:\Users\ASUS\Desktop\Praktikum-APD\post-test\post-test-apd-3> py .\main.py
=== Mengidentifikasi Segitiga dan luasnya ===

Masukkan Sisi A: 4
Masukkan Sisi B: 4
Masukkan Sisi C: 4

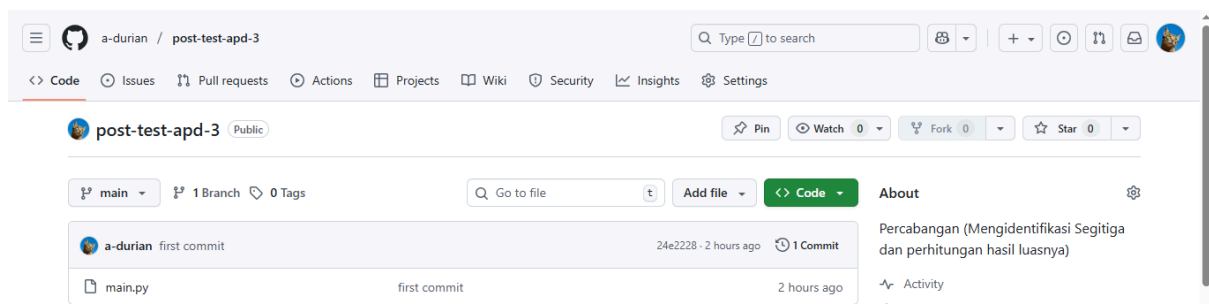
Segitiga Sama Sisi
Hasil luasnya adalah: 6.928203230275509
PS C:\Users\ASUS\Desktop\Praktikum-APD\post-test\post-test-apd-3> py .\main.py
=== Mengidentifikasi Segitiga dan luasnya ===

Masukkan Sisi A: 5
Masukkan Sisi B: 5
Masukkan Sisi C: 4

Segitiga Sama Kaki
Hasil luasnya adalah: 9.16515138991168
```

Gambar 4.1 Hasil Output

## 5. Langkah-langkah GIT



### 5.1 GIT Init

```
PS C:\Users\ASUS\Desktop\Praktikum-APD\post-test\post-test-apd-3> git init
Initialized empty Git repository in C:/Users/ASUS/Desktop/Praktikum-APD/post-test/post-test-apd-3/.git/
```

### 5.2 GIT Add

```
PS C:\Users\ASUS\Desktop\Praktikum-APD\post-test\post-test-apd-3> git add .
```

## 5.3 GIT Commit

```
PS C:\Users\ASUS\Desktop\Praktikum-APD\post-test\post-test-apd-3> git commit -m "first commit"
[main (root-commit) 24e2228] first commit
1 file changed, 21 insertions(+)
create mode 100644 main.py
```

## 5.4 GIT Remote

```
PS C:\Users\ASUS\Desktop\Praktikum-APD\post-test\post-test-apd-3> git remote add origin https://github.com/a-durian/post-test-apd-3.git
```

## 5.5 GIT Push

```
PS C:\Users\ASUS\Desktop\Praktikum-APD\post-test\post-test-apd-3> git push -u origin main
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Delta compression using up to 2 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 536 bytes | 53.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/a-durian/post-test-apd-3.git
 * [new branch]      main -> main
branch 'main' set up to track 'origin/main'.
```