

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

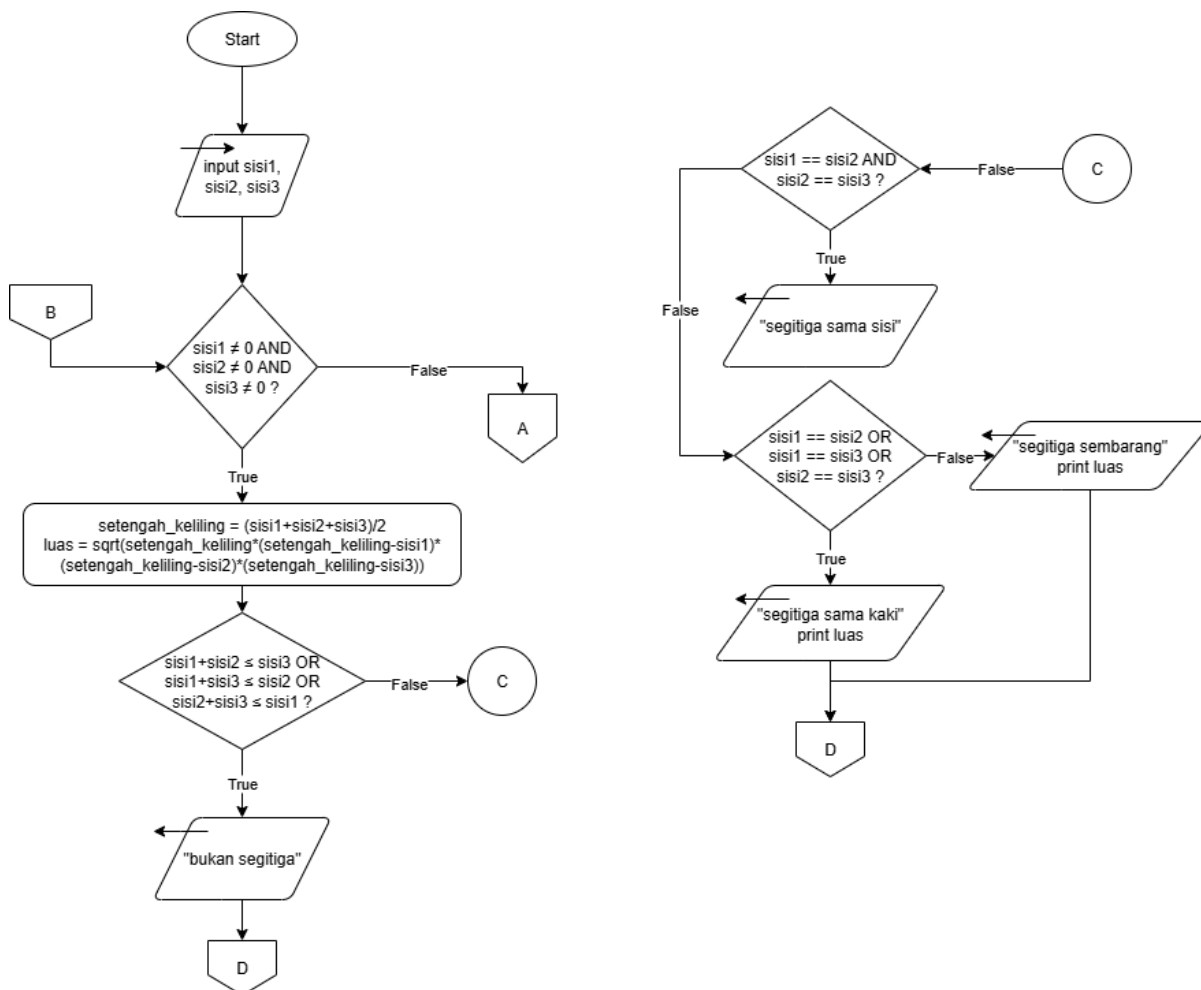


**Disusun oleh:**  
**Nama (2509106038)**  
**Kelas (A2 '25)**

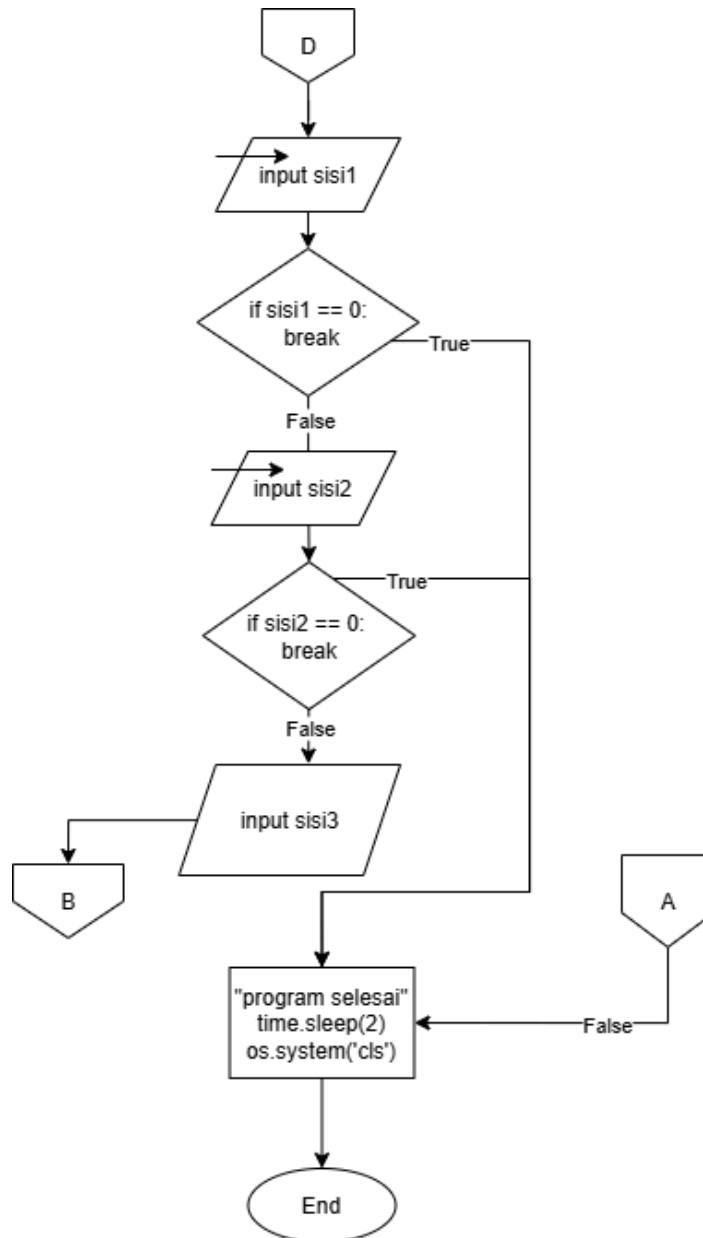
**PROGRAM STUDI INFORMATIKA**  
**UNIVERSITAS MULAWARMAN**  
**SAMARINDA**  
**2025**

## 1. Flowchart

pada flowchart ini sama dimulai dengan input sisi-sisi oleh user, setelah user selesai memasukkan sisi, akan di cek apakah semua sisi tidak bernilai 0, jika benar, flowchart akan melakukan perulangan dan menentukan jenis segitiga apa berdasarkan input oleh user, jika salah, flowchart akan selesai dan membersihkan terminal, setelah ditentukan segitiganya oleh flowchart, akan terjadi perulangan sampai user menginput nomor untuk berhenti (0) pada setiap input atau salah satu saja.



gambar 1.1 flowchart kondisi program dan penentuan jenis segitiga



flowchart 1.2 flowchart input ulang sisi dengan opsi berhenti

## 2. Deskripsi Singkat Program

sama seperti posttest sebelumnya, program ini berfungsi untuk menentukan jenis segitiga berdasarkan user input, hanya saja terdapat looping, sehingga program akan terus berjalan hingga user memilih opsi untuk berhenti, yaitu input nomor 0 pada salah satu atau setiap input

### 3. Source Code

#### A. program utama

disini terdapat looping dan pengkategorian jenis segitiga

Source Code:

```
while sisi1 and sisi2 and sisi3 != 0 :
    setengah_keliling = (sisi1 + sisi2 + sisi3) / 2
    luas = (setengah_keliling * (setengah_keliling - sisi1) *
(setengah_keliling - sisi2) * (setengah_keliling - sisi3)) ** 0.5

    if sisi1 + sisi2 <= sisi3 or sisi1 + sisi3 <= sisi2 or sisi2 + sisi3 <=
sisi1 :
        print("bukan segitiga")
        print("Note : masukkan angka 0 untuk berhenti")

        sisi1 = int(input("masukkan sisi pertama : "))
        if sisi1 == 0 :
            break

        sisi2 = int(input("masukkan sisi kedua : "))
        if sisi2 == 0 :
            break

        sisi3 = int(input("masukkan sisi ketiga : "))

    elif sisi1 == sisi2 and sisi2 == sisi3 :
        print("segitiga sama sisi")
        print(f"luas : {luas}")

        sisi1 = int(input("masukkan sisi pertama : "))
        if sisi1 == 0 :
            break

        sisi2 = int(input("masukkan sisi kedua : "))
```

```

if sisi2 == 0 :
    break

sisi3 = int(input("masukkan sisi ketiga : "))

elif sisi1 == sisi2 or sisi1 == sisi3 or sisi2 == sisi3 :
    print("segitiga siku-suku")
    print(f"luas : {luas}")

sisi1 = int(input("masukkan sisi pertama : "))
if sisi1 == 0 :
    break

sisi2 = int(input("masukkan sisi kedua : "))
if sisi2 == 0 :
    break

sisi3 = int(input("masukkan sisi ketiga : "))

else :
    print("segitiga sembarang")
    print(f"luas : {luas}")

sisi1 = int(input("masukkan sisi pertama : "))
if sisi1 == 0 :
    break

sisi2 = int(input("masukkan sisi kedua : "))
if sisi2 == 0 :
    break

sisi3 = int(input("masukkan sisi ketiga : "))

```

## B. Fitur membersihkan terminal

setelah user memilih opsi berhenti, program akan membersihkan terminal, lalu program selesai

```
print("program selesai, membersihkan terminal....")  
  
time.sleep(2)  
  
os.system('cls')
```

## 4. Hasil Output

```
st-test/post-test-apd-4/2509106038-Febrianno-Ozora-Alinanto-PI-4.py  
masukkan sisi pertama : 12  
masukkan sisi kedua : 12  
masukkan sisi ketiga : 12  
segitiga sama sisi  
luas : 62.353829072479584  
masukkan sisi pertama : 0  
program selesai, membersihkan terminal....
```

Gambar 4.1 hasil output

```
PS G:\praktikum\APD\A2-25\post-test\post-test-apd-4>
```

Gambar 4.2 setelah program selesai terjalan

## 5. Langkah-langkah GIT

### 5.1 GIT Init

```
ozora@ozra MINGW64 /g/praktikum/APD (main)  
$ git init  
Reinitialized existing Git repository in G:/praktikum/APD/.git/
```

gambar 5.1 inisiasi git

### 5.2 GIT Add

```
ozora@ozra MINGW64 /g/praktikum/APD (main)  
$ git add .
```

gambar 5.2 menambahkan file baru ke git

## 5.3 GIT Commit

```
ozora@ozra MINGW64 /g/praktikum/APD (main)
$ git commit -m "submit possttest 4"
[main 5fe0c0d] submit possttest 4
3 files changed, 168 insertions(+)
create mode 100644 A2-25/pertemuan-4/looping.py
create mode 100644 A2-25/post-test/post-test-apd-4/2509106038-Febrianno-Ozora-Alinanto-PT-4 .pdf
create mode 100644 A2-25/post-test/post-test-apd-4/2509106038-Febrianno-Ozora-Alinanto-PT-4.py
```

gambar 5.3 commit git local

## 5.4 GIT Remote



```
ozora@ozra MINGW64 /g/praktikum/APD (main)
$ git remote add origin https://github.com/ojola444/praktikum-apd
error: remote origin already exists.
```

gambar 5.4 git remote

## 5.5 GIT Push

```
ozora@ozra MINGW64 /g/praktikum/APD (main)
$ git push -f origin main
Enumerating objects: 35, done.
Counting objects: 100% (35/35), done.
Delta compression using up to 16 threads
Compressing objects: 100% (27/27), done.
Writing objects: 100% (35/35), 1.14 MiB | 279.00 KiB/s, done.
Total 35 (delta 5), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (5/5), done.
To https://github.com/ojola444/praktikum-apd
+ 22cdc85...5fe0c0d main -> main (forced update)
```

gambar 5.5 push commit local ke repository main

 2509106038-Febrianno-Ozora-Alinanto-PT-4 .pdf	submit possttest 4	30 minutes ago
 2509106038-Febrianno-Ozora-Alinanto-PT-4.py	submit possttest 4	30 minutes ago

gambar 5.6 hasil push github