

**LAPORAN PRAKTIKUM ALGORITMA
DAN PEMROGRAMAN 1**

**MODUL 4
I/O, TIPE DATA & VARIABLE**



Disusun oleh:

FIRDAUS RAMADHANA

109082500151

S1IF-13-07

Asisten Praktikum

Adithana dharma putra

Apri pandu wicaksono

PROGRAM STUDI S1 INFORMATIKA

FAKULTAS INFORMATIKA

TELKOM UNIVERSITY PURWOKERTO

2025

LATIHAN KELAS – GUIDED

1. Guided 1 Source Code

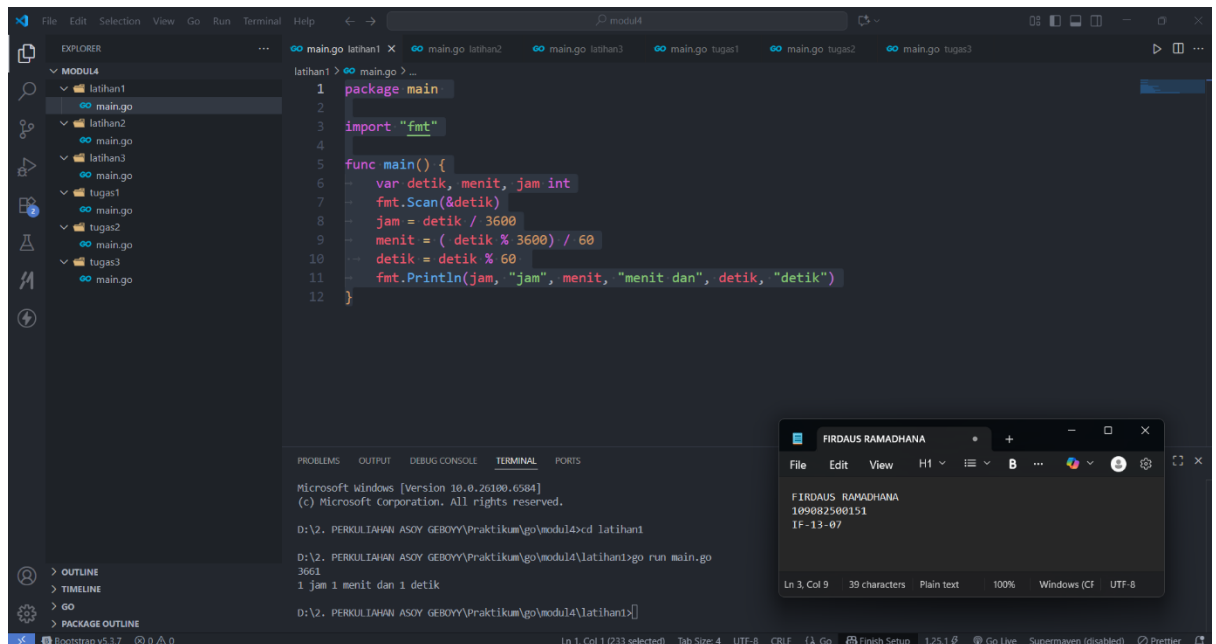
```
package main

import "fmt"

func main() {
    var detik, menit, jam int
    fmt.Scan(&detik)
    jam = detik / 3600
    menit = ( detik % 3600) / 60
    detik = detik % 60
    fmt.Println(jam, "jam", menit, "menit dan", detik,
"detik")
}
```

Screenshoot program

//tambahkan tangkapan layar dari program (boleh lebih dari 1 jika diperlukan)
CONTOH TANGKAPAN LAYAR:



Deskripsi program

Program ini menerima input berupa jumlah detik dari user, kemudian mengkonversinya menjadi format waktu yang lebih mudah dibaca yaitu jam, menit, dan detik. Program akan menghitung berapa jam, menit, dan detik yang setara dengan jumlah detik yang dimasukkan, lalu menampilkan hasil konversinya.

Cara kerjanya:

1. Meminta user memasukkan jumlah detik
2. Mengkonversi detik menjadi jam dengan membagi 3600 (karena 1 jam = 3600 detik)
3. Mengkonversi sisa detik setelah dikurangi jam menjadi menit dengan membagi 60
4. Mengambil sisa detik yang tersisa setelah dikonversi ke jam dan menit
5. Menampilkan hasil dalam format yang mudah dipahami

Contoh: Jika dimasukkan 3665 detik, program akan menampilkan "1 jam, 1 menit, dan 5 detik".

2. Guided 2

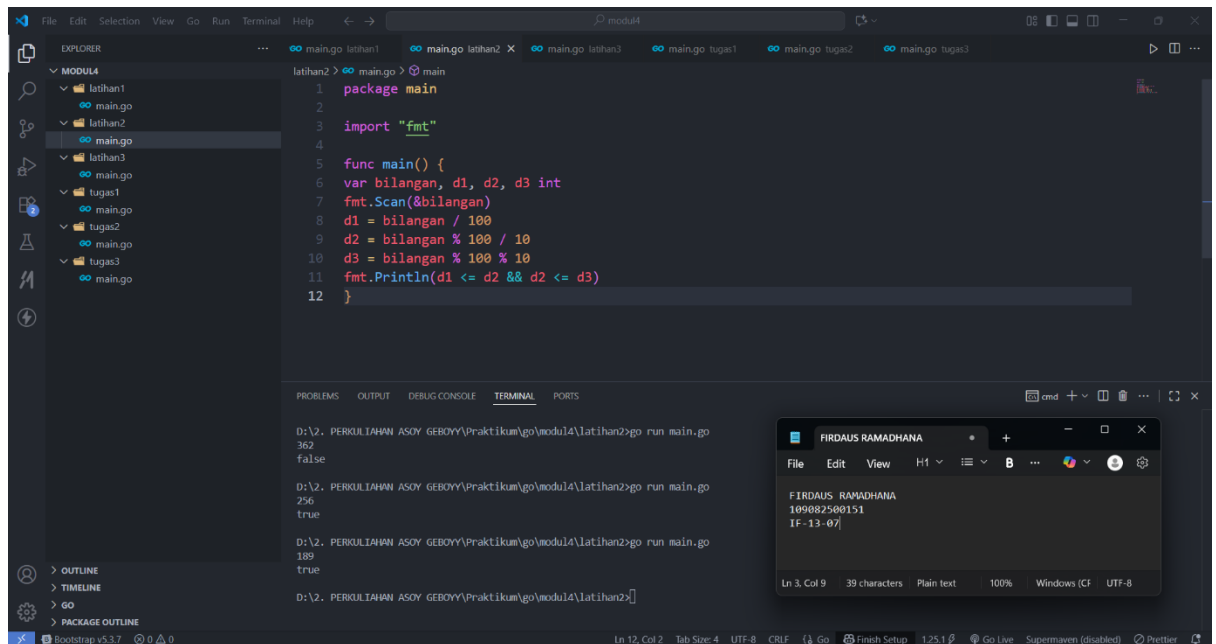
Source Code

```
package main

import "fmt"

func main() {
    var bilangan, d1, d2, d3 int
    fmt.Scan(&bilangan)
    d1 = bilangan / 100
    d2 = bilangan % 100 / 10
    d3 = bilangan % 100 % 10
    fmt.Println(d1 <= d2 && d2 <= d3)
}
```

Screenshoot program



Deskripsi program

Program ini menerima input sebuah bilangan tiga digit dari user, kemudian memeriksa apakah digit-digitnya tersusun dalam urutan yang tidak menurun (digit pertama lebih kecil atau sama dengan digit kedua, dan digit kedua lebih kecil atau sama dengan digit ketiga). Program akan menampilkan nilai true jika digit-digitnya terurut, dan false jika tidak terurut.

Cara kerjanya:

1. Meminta user memasukkan sebuah bilangan tiga digit
2. Memisahkan bilangan menjadi tiga digit terpisah:
 - d1 = digit pertama (ratusan) diperoleh dengan membagi 100
 - d2 = digit kedua (puluhan) diperoleh dengan sisa bagi 100 lalu dibagi 10
 - d3 = digit ketiga (satuan) diperoleh dengan sisa bagi 100 lalu sisa bagi 10
3. Memeriksa apakah $d1 \leq d2 \leq d3$
4. Menampilkan hasil pemeriksaan (true atau false)

Contoh:

- Input 123 => true (karena $1 \leq 2 \leq 3$)
- Input 321 => false (karena $3 > 2 > 1$)
- Input 112 => true (karena $1 \leq 1 \leq 2$)
- Input 111 => true (karena $1 = 1 = 1$)

3. Guided 3

Source Code

```
package main
```

```
import "fmt"

func main() {

    var beratBadan, tinggiBadan, bmi float64

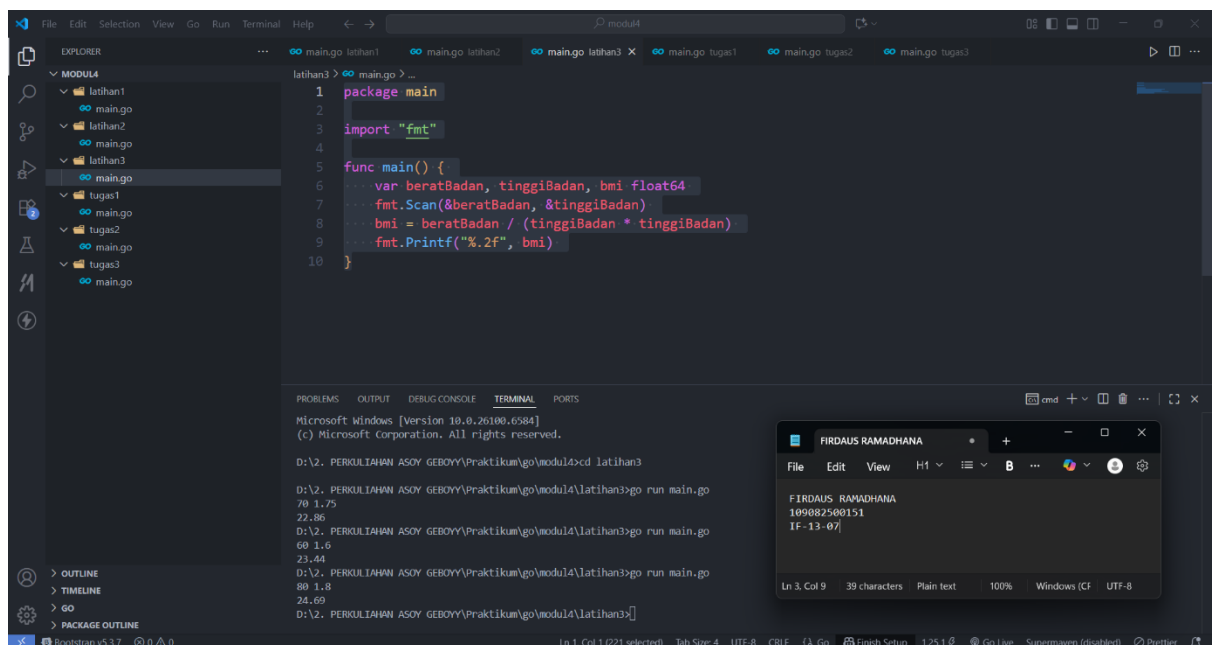
    fmt.Scan(&beratBadan, &tinggiBadan)

    bmi = beratBadan / (tinggiBadan * tinggiBadan)

    fmt.Printf("%.2f", bmi)

}
```

Screenshoot program



Deskripsi program

Program ini menerima input berat badan (dalam kilogram) dan tinggi badan (dalam meter) dari user, kemudian menghitung nilai Indeks Massa Tubuh (BMI) dengan rumus berat badan dibagi kuadrat tinggi badan. Program kemudian menampilkan hasil perhitungan BMI dengan format dua angka di belakang koma.

Cara kerjanya:

1. Meminta pengguna memasukkan dua nilai: berat badan dan tinggi badan
2. Menghitung BMI menggunakan rumus: $BMI = \text{berat badan} / (\text{tinggi badan} \times \text{tinggi badan})$
3. Menampilkan hasil perhitungan BMI dengan presisi dua desimal

Contoh:

Jika dimasukkan berat badan 70 kg dan tinggi badan 1.75 m

Program menghitung: $70 / (1.75 \times 1.75) = 70 / 3.0625 = 22.86$

Program menampilkan: "22.86"

TUGAS

1. Tugas 1

Source code

```
import "fmt"

func main() {

    var totalbelanja, diskon int

    fmt.Scan(&totalbelanja)

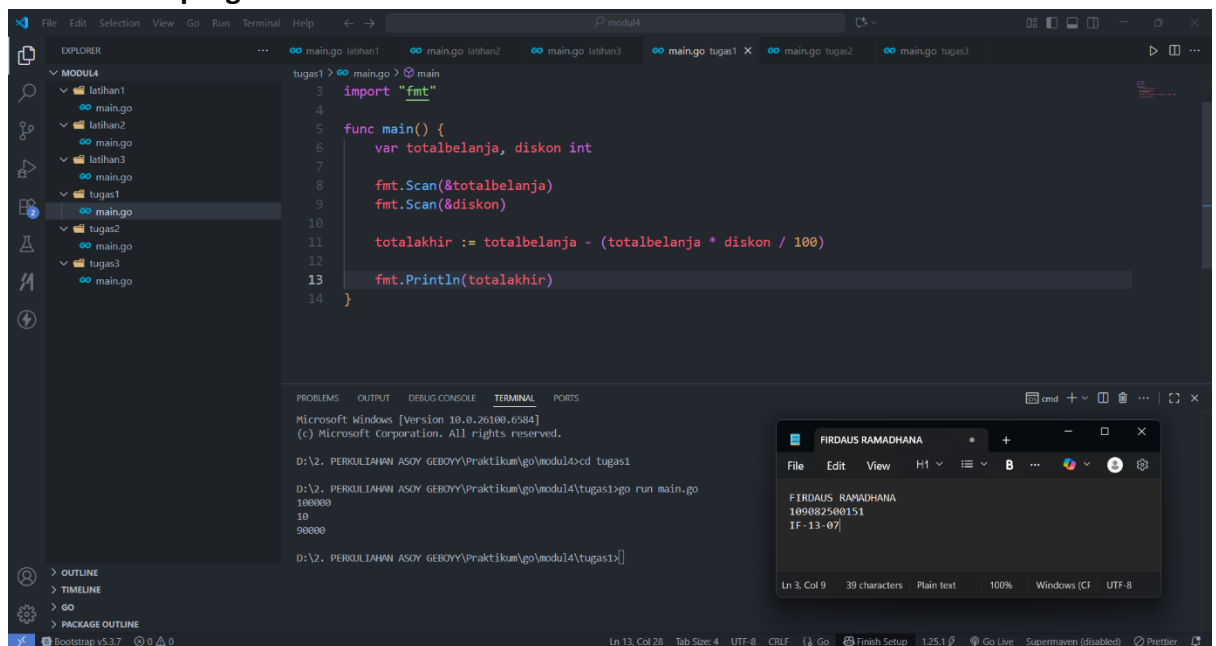
    fmt.Scan(&diskon)

    totalakhir := totalbelanja - (totalbelanja * diskon
/ 100)

    fmt.Println(totalakhir)

}
```

Screenshoot program



Deskripsi program

Program ini menerima input total belanja dan persentase diskon dari user, kemudian menghitung jumlah uang yang harus dibayar setelah dipotong diskon. Program menampilkan total akhir hasil perhitungan tersebut.

Cara kerjanya:

1. Meminta user memasukkan dua nilai: total belanja dan persentase diskon
2. Menghitung besarnya potongan harga: $\text{total belanja} \times \text{diskon} / 100$
3. Menghitung total akhir: $\text{total belanja} - \text{potongan harga}$
4. Menampilkan total akhir yang harus dibayar

Contoh dengan input yang diberikan:

- Input total belanja: 100000
- Input diskon: 10
- Perhitungan:
 - Potongan harga = $100000 \times 10 / 100 = 10000$
 - Total akhir = $100000 - 10000 = 90000$
- Program menampilkan: 90000

Jadi, untuk total belanja Rp 100.000 dengan diskon 10%, program akan menghitung dan menampilkan jumlah yang harus dibayar sebesar Rp 90.000.

2. Tugas 2

Source code

```
package main

import "fmt"

func main() {

    var bmi, tinggi float64

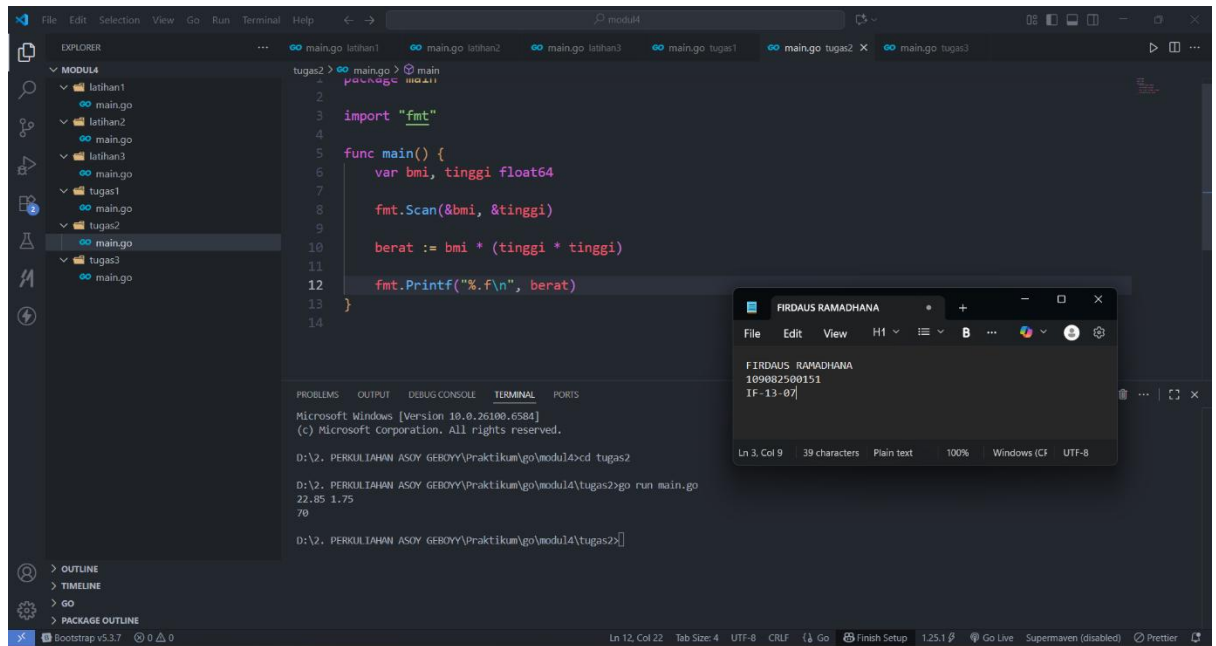
    fmt.Scan(&bmi, &tinggi)

    berat := bmi * (tinggi * tinggi)

    fmt.Printf("%.f\n", berat)

}
```

Screenshoot program



Deskripsi program

Program ini menerima input nilai BMI dan tinggi badan (dalam meter) dari user, kemudian menghitung berat badan yang sesuai menggunakan rumus kebalikan dari perhitungan BMI. Program menampilkan hasil perhitungan berat badan dalam bentuk bilangan bulat.

Cara kerjanya:

1. Meminta pengguna memasukkan dua nilai: BMI dan tinggi badan
2. Menghitung berat badan menggunakan rumus: berat = BMI × (tinggi badan × tinggi badan)
3. Menampilkan hasil perhitungan berat badan tanpa desimal (bilangan bulat)

Contoh dengan input yang diberikan:

- Input BMI: 22.85
- Input tinggi badan: 1.75 m
- Perhitungan:

$$\text{Berat} = 22.85 \times (1.75 \times 1.75)$$

$$= 22.85 \times 3.0625$$

$$= 69.978125$$
- Program menampilkan: 70 (karena dibulatkan menjadi bilangan bulat)

Jadi, untuk seseorang dengan BMI 22.85 dan tinggi badan 1.75 m, program menghitung bahwa berat badannya sekitar 70 kg.

3. Tugas 3

Source code

```
package main
```



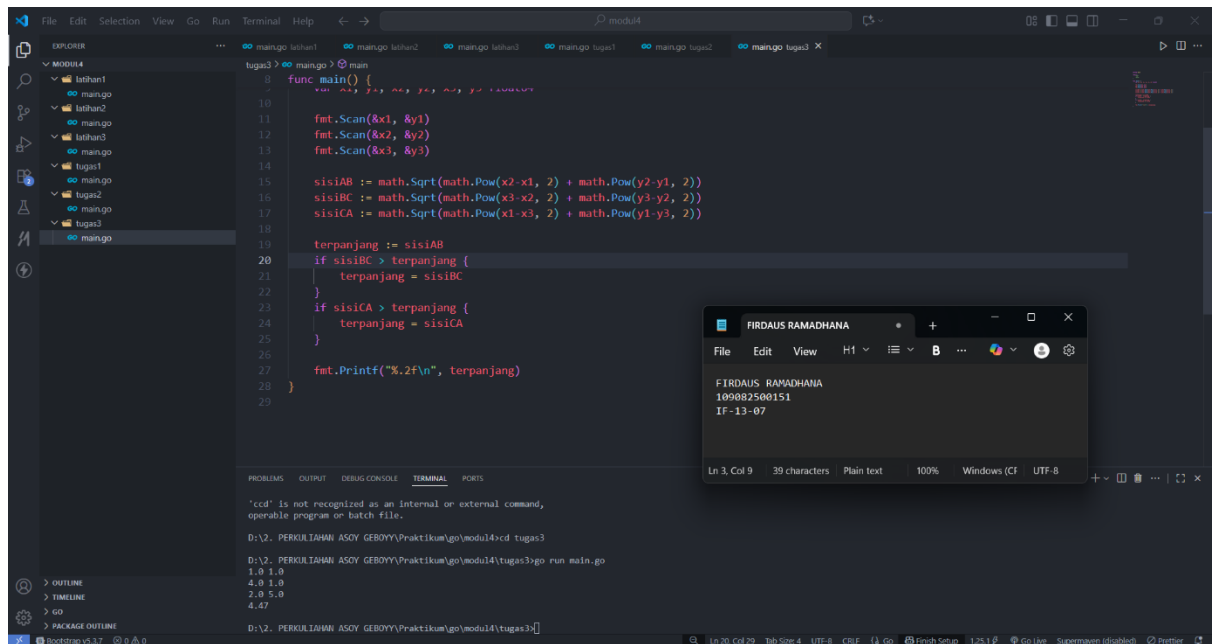
```

import (
    "fmt"
    "math"
)

func main() {
    var x1, y1, x2, y2, x3, y3 float64
    fmt.Scan(&x1, &y1)
    fmt.Scan(&x2, &y2)
    fmt.Scan(&x3, &y3)
    sisiAB := math.Sqrt(math.Pow(x2-x1, 2) +
math.Pow(y2-y1, 2))
    sisiBC := math.Sqrt(math.Pow(x3-x2, 2) +
math.Pow(y3-y2, 2))
    sisiCA := math.Sqrt(math.Pow(x1-x3, 2) +
math.Pow(y1-y3, 2))
    terpanjang := sisiAB
    if sisiBC > terpanjang {
        terpanjang = sisiBC
    }
    if sisiCA > terpanjang {
        terpanjang = sisiCA
    }
    fmt.Printf("%.2f\n", terpanjang)
}

```

Screenshoot program



```
func main() {  
    // x1, y1, x2, y2, x3, y3  
    10  
    11     fmt.Scan(&x1, &y1)  
    12     fmt.Scan(&x2, &y2)  
    13     fmt.Scan(&x3, &y3)  
    14  
    15     sisiAB := math.Sqrt(math.Pow(x2-x1, 2) + math.Pow(y2-y1, 2))  
    16     sisiBC := math.Sqrt(math.Pow(x3-x2, 2) + math.Pow(y3-y2, 2))  
    17     sisiCA := math.Sqrt(math.Pow(x1-x3, 2) + math.Pow(y1-y3, 2))  
    18  
    19     terpanjang := sisiAB  
    20     if sisiBC > terpanjang {  
    21         terpanjang = sisiBC  
    22     }  
    23     if sisiCA > terpanjang {  
    24         terpanjang = sisiCA  
    25     }  
    26  
    27     fmt.Printf("%.2f\n", terpanjang)  
    28 }  
    29
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

'cmd' is not recognized as an internal or external command,
operable program or batch file.

D:\2. PERKULIAHAN ASOY GEBOYYY\Praktikum\go\modul4\tugas3
D:\2. PERKULIAHAN ASOY GEBOYYY\Praktikum\go\modul4\tugas3>go run main.go
1.0 1.0
4.0 1.0
2.0 5.0
4.47

Deskripsi program

Program ini menerima input tiga pasang koordinat titik (x,y) yang membentuk sebuah segitiga, kemudian menghitung panjang masing-masing sisi segitiga menggunakan rumus jarak Euclidean antara titik-titik tersebut, dan akhirnya menentukan serta menampilkan panjang sisi terpanjang dari segitiga tersebut dengan format dua angka di belakang koma.

Cara kerjanya:

1. Meminta user memasukkan tiga titik koordinat: (x1,y1), (x2,y2), (x3,y3)
2. Menghitung panjang ketiga sisi segitiga:
sisi AB = jarak antara titik A(x1,y1) dan B(x2,y2)
sisi BC = jarak antara titik B(x2,y2) dan C(x3,y3)
sisi CA = jarak antara titik C(x3,y3) dan A(x1,y1)
3. Mencari sisi terpanjang dengan membandingkan ketiga sisi
4. Menampilkan panjang sisi terpanjang dengan format dua desimal

Contoh dengan input yang diberikan:

Titik A: (1.0, 1.0)

Titik B: (4.0, 1.0)

Titik C: (2.0, 5.0)

Perhitungan:

- sisi AB = $\sqrt{(4-1)^2 + (1-1)^2} = \sqrt{9 + 0} = 3.00$
- sisi BC = $\sqrt{(2-4)^2 + (5-1)^2} = \sqrt{4 + 16} = \sqrt{20} \approx 4.47$
- sisi CA = $\sqrt{(1-2)^2 + (1-5)^2} = \sqrt{1 + 16} = \sqrt{17} \approx 4.12$

Sisi terpanjang adalah BC ≈ 4.47

Program akan menampilkan: 4.47