

**LAPORAN PRAKTIKUM ALGORITMA
DAN PEMROGRAMAN 1**

MODUL 04

I/O, TIPE DATA & VARIABEL (LATIHAN 2)



Disusun oleh:

RAFFI YULIANSYAH

109082500101

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)

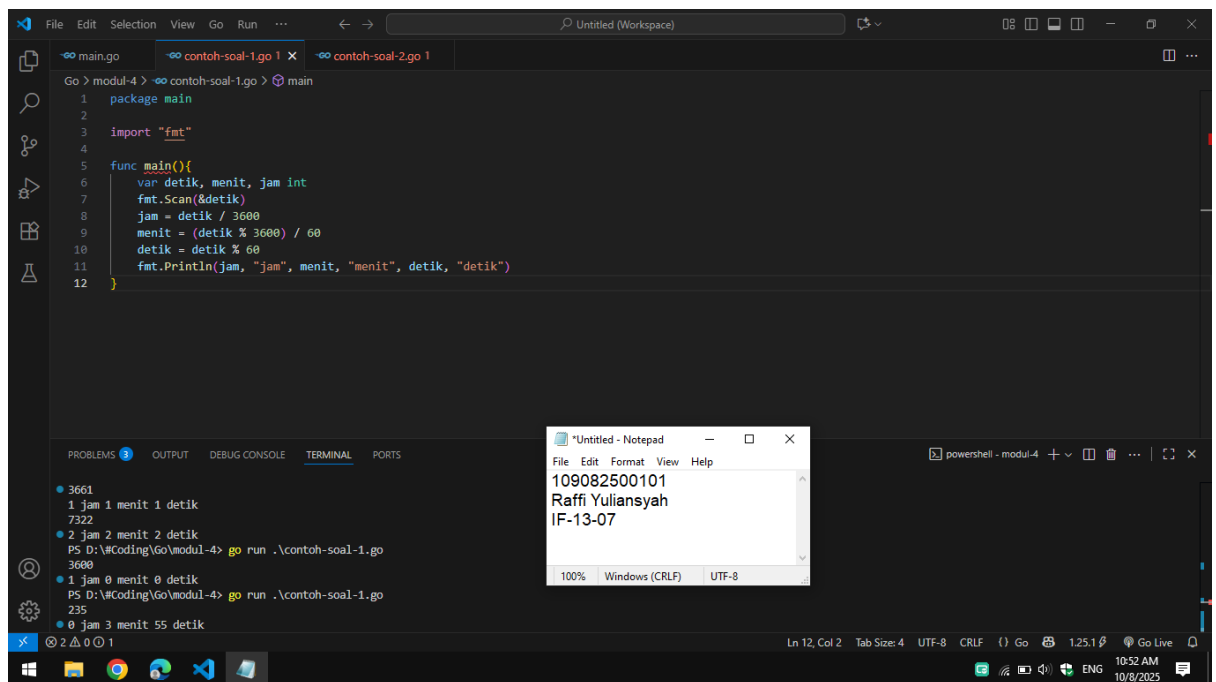
    jam = detik / 3600

    menit = (detik % 3600) / 60

    detik = detik % 60

    fmt.Println(jam, "jam", menit, "menit", detik,
"detik")
}
```

Screenshoot program



Deskripsi program

Program di atas berfungsi untuk mengubah waktu dengan satuan detik menjadi format jam, menit, dan detik.

Program awanya akan mendeklarasikan tiga variabel bilangan bulat (int), yaitu detik, menit, dan jam. Kemudian program akan menerima input berupa nilai dari jumlah detik dan akan disimpan ke dalam variabel detik. Setelah itu, program akan menghitung jumlah jam dengan membagi total detik dengan 3600. Kedua, menghitung menit dengan modulo detik dengan 3600 agar mendapat sisa detik, lalu membaginya dengan 60 untuk mendapatkan menit. Dan ketiga, menghitung sisa detik terakhir menggunakan modulo 3600 kemudian modulo 60 sehingga didapatkan sisa detik. Hasil perhitungan tersebut disimpan kedalam variabel masing masing. Terakhir, program menampilkan hasil perhitungan jam, menit, dan detik tersebut.

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

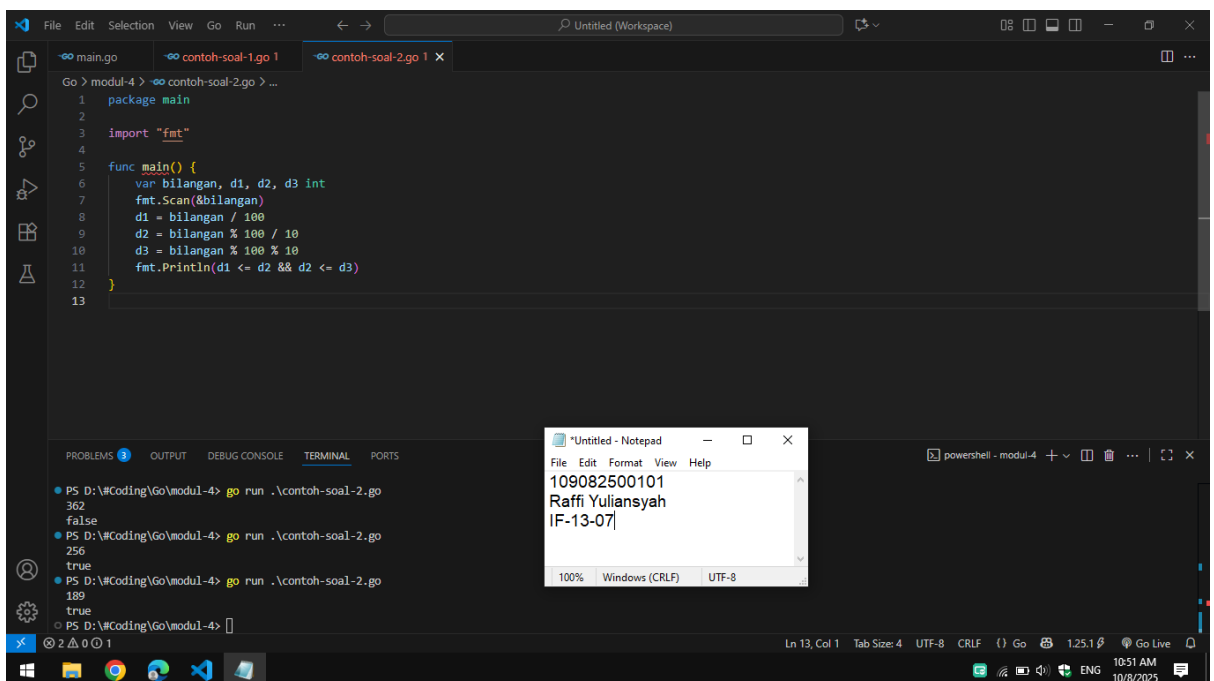
    d3 = bilangan % 100 % 10

    fmt.Println(d1 <= d2 && d2 <= d3)

}

```

Screenshoot program



Deskripsi program

Program ini berfungsi untuk mengecek apakah digit-digit dari sebuah bilangan tiga angka/digit memiliki urutan menaik. Program dimulai dengan mendeklarasikan empat variabel bilangan bulat (int): variabel "bilangan" untuk menampung input angka secara utuh(3 digit), kemudian variabel d1, d2, dan d3 untuk menyimpan masing-masing digitnya sesuai dengan urutan, d1 untuk digit pertama dan seterusnya. Program kemudian akan menerima input(fmt.Scan()) yang berupa bilangan bulat 3 digit, setelahnya bilangan tersebut disimpan ke dalam variabel "bilangan".

Selanjutnya, program akan melakukan pemisahan digit dari variabel "bilangan":

- Digit pertama (variabel d1) diambil dengan membagi bilangan dengan 100.
- Digit kedua (variabel d2) didapat dengan mengambil sisa bagi bilangan dengan 100, lalu hasilnya dibagi 10.
- Digit ketiga (variabel d3) didapat dengan mengambil sisa bagi bilangan dengan 100 dan 10.

Setelah ketiga digit terpisah, program akan melakukan proses pengecekan logika: apakah d1 lebih kecil dari d2 DAN d2 lebih kecil dari d3 ($d1 \leq d2 \ \&\& \ d2 \leq d3$). Proses tersebut akan menghasilkan nilai benar atau salah (true jika kondisi terpenuhi atau false jika tidak) yang ditampilkan langsung lewat `fmt.Println`.

3. Guided 3

Source Code

```
package main

import "fmt"

func main() {

    var beratBadan, tinggiBadan, bmi float64

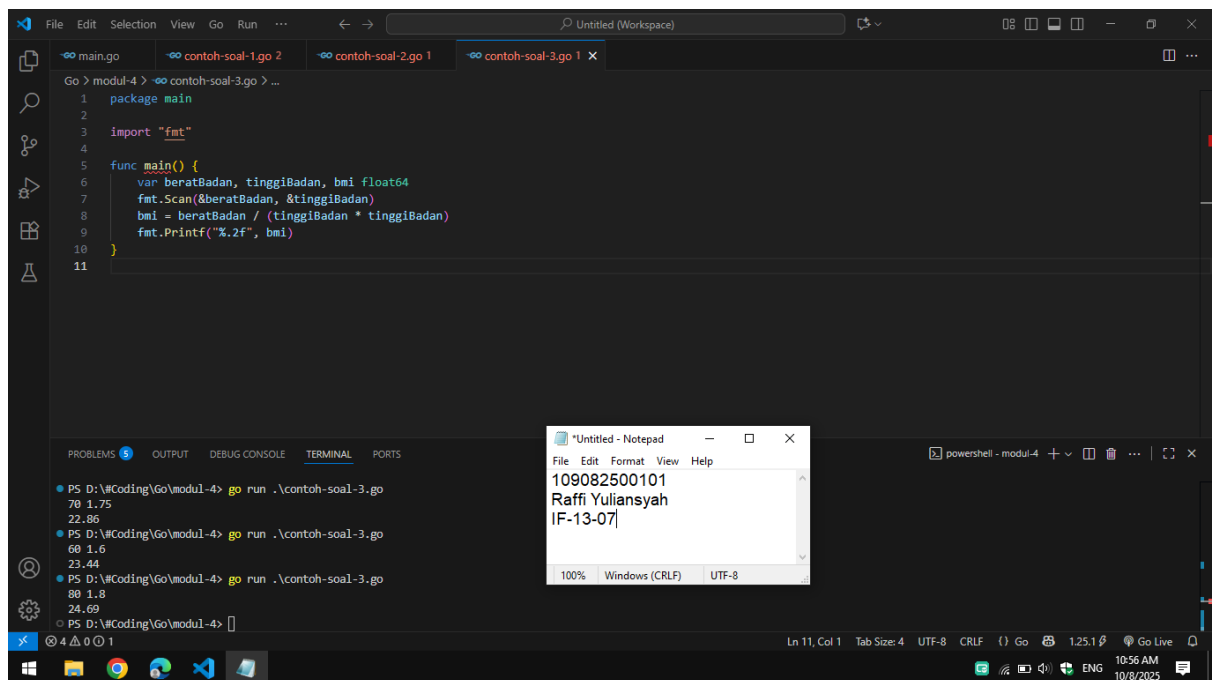
    fmt.Scan(&beratBadan, &tinggiBadan)

    bmi = beratBadan / (tinggiBadan * tinggiBadan)

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

}
```

Screenshoot program



Deskripsi program

Program untuk menghitung Indeks Massa Tubuh (BMI) berdasarkan input berat dan tinggi badan. Program diawali dengan mendeklarasikan tiga variabel bertipe data bilangan desimal (float64): beratBadan, tinggiBadan, dan bmi. Program kemudian mendapat input dua angka, yaitu berat badan (dalam kg) dan tinggi badan (dalam meter), yang kemudian akan disimpan ke dalam variabel beratBadan dan tinggiBadan. Setelahnya, program melakukan perhitungan BMI dengan membagi beratBadan dengan kuadrat dari tinggiBadan (tinggi badan dikali tinggi badan). Hasil dari perhitungan kemudian disimpan ke dalam variabel bmi. Terakhir, program menampilkan output nilai BMI dari variabel bmi dengan format yang dibulatkan menjadi dua angka di belakang koma (%.2f).

TUGAS

1. Tugas 1

Source code

```
package main

import "fmt"

func main () {

    var totalBelanja, diskon float64

    fmt.Print("Input, Total Belanja : ")

    fmt.Scanln(&totalBelanja)

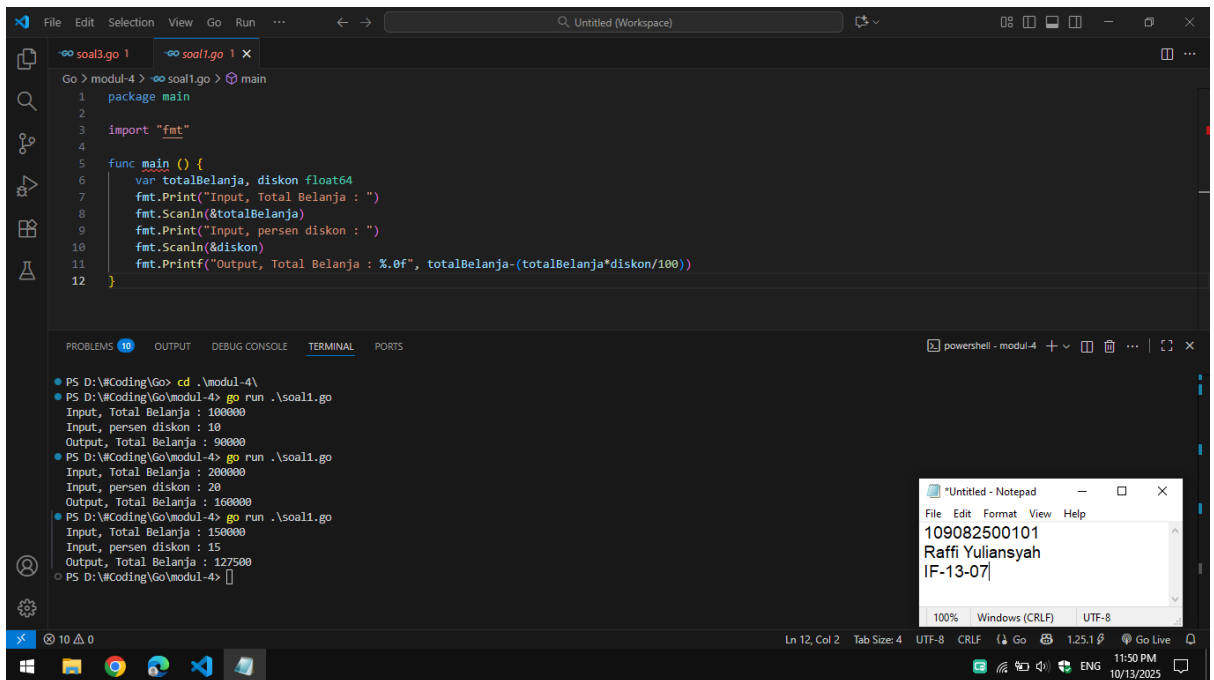
    fmt.Print("Input, persen diskon : ")

    fmt.Scanln(&diskon)

    fmt.Printf("Output, Total Belanja : %.0f", totalBelanja-
(totalBelanja*diskon/100))

}
```

Screenshoot program



Deskripsi program

Program ini akan menghitung harga/total belanja yang harus dibayarkan setelah mendapatkan diskon (dalam persen). Program dimulai dengan mendeklarasikan tiga variabel bilangan desimal (float64), yaitu totalBelanja, diskon, dan totalBelanjaAkhir. Program meminta memasukkan total belanja awal dan menyimpannya di variabel totalBelanja. Setelah itu, program kembali meminta memasukkan persentase diskon (dalam persen) yang akan disimpan di variabel diskon.

Program akan menghitung jumlah potongan harga dengan mengalikan totalBelanja dengan diskon lalu dibagi 100. totalBelanja kemudian dikurangkan dengan potongan harga yang tadi sudah dihitung. Sehingga total akhir yang harus dibayarkan didapatkan dan disimpan ke variabel totalBelanjaAkhir. Terakhir, program menampilkan total belanja akhir tersebut dengan format yang dibulatkan menjadi bilangan bulat (tanpa angka desimal).

2. Tugas 2

Source code

```
package main

import "fmt"

func main() {

    var bmi, tinggiBadan float64
```

```

    fmt.Print("Input, BMI dan Tinggi Badan : ")

    fmt.Scan(&bmi)

    fmt.Scanln(&tinggiBadan)

    fmt.Printf("Output, Berat Badan : %.0f",
    bmi*(tinggiBadan*tinggiBadan))

}

```

Screenshoot program

The screenshot shows a Go IDE with the following source code in `soal2.go`:

```

1 package main
2
3 import "fmt"
4
5 func main () {
6     var bmi, tinggiBadan float64
7     fmt.Print("Input, BMI dan Tinggi Badan : ")
8     fmt.Scan(&bmi)
9     fmt.Scanln(&tinggiBadan)
10    fmt.Printf("Output, Berat Badan : %.0f", bmi*(tinggiBadan*tinggiBadan) )
11 }

```

The terminal output shows three test runs:

```

PS D:\Coding\Go\modul-4> go run soal2.go
Input, BMI dan Tinggi Badan : 22.85 1.75
Output, Berat Badan : 70.0
PS D:\Coding\Go\modul-4> go run soal2.go
Input, BMI dan Tinggi Badan : 23.43 1.6
Output, Berat Badan : 60.0
PS D:\Coding\Go\modul-4> go run soal2.go
Input, BMI dan Tinggi Badan : 24.69 1.8
Output, Berat Badan : 80.0
PS D:\Coding\Go\modul-4>

```

An inset window shows a Notepad file with the text:

```

109082500101
Raffi Yuliansyah
IF-13-07

```

Deskripsi program

Program berfungsi untuk menghitung berat badan berdasarkan skor Indeks Massa Tubuh (IMT atau BMI) dan tinggi badan yang dimasukkan.

Programnya dimulai dengan mendklarasikan tiga variabel bilangan desimal (float64), yaitu `bmi`, `tinggiBadan`, dan `beratBadan`. Program kemudian akan meminta memasukkan nilai BMI serta tinggi badan dalam satuan meter. Kedua nilai input ini akan disimpan ke dalam variabel `bmi` dan `tinggiBadan`.

Program akan menghitung berat badan mengalikan nilai `bmi` yang diinput dengan kuadrat dari `tinggiBadan` (sesuai rumus BMI yang dibalik: $\text{beratBadan} = \text{bmi} * \text{tinggiBadan} * \text{tinggiBadan}$). Hasil dari perhitungan tersebut kemudian disimpan ke dalam variabel `beratBadan`. Terakhir, program akan menampilkan hasil perhitungan berat badan dengan hasil yang dibulatkan menjadi nol angka di belakang koma (bilangan bulat terdekat).

3. Tugas 3

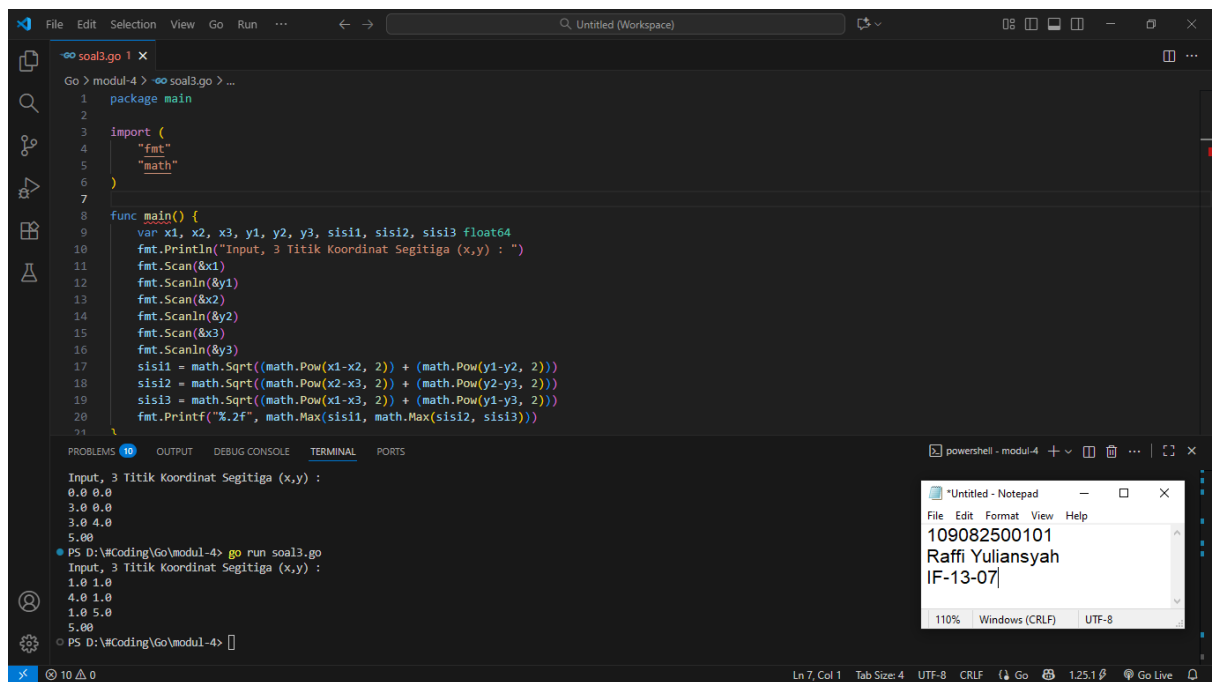
Source code

```
package main

import (
    "fmt"
    "math"
)

func main() {
    var x1, x2, x3, y1, y2, y3, sisi1, sisi2, sisi3
    float64
    fmt.Println("Input, 3 Titik Koordinat Segitiga (x,y) : ")
    fmt.Scan(&x1)
    fmt.Scanln(&y1)
    fmt.Scan(&x2)
    fmt.Scanln(&y2)
    fmt.Scan(&x3)
    fmt.Scanln(&y3)
    sisi1 = math.Sqrt((math.Pow(x1-x2, 2)) +
    (math.Pow(y1-y2, 2)))
    sisi2 = math.Sqrt((math.Pow(x2-x3, 2)) +
    (math.Pow(y2-y3, 2)))
    sisi3 = math.Sqrt((math.Pow(x1-x3, 2)) +
    (math.Pow(y1-y3, 2)))
    fmt.Printf("%.2f", math.Max(sisi1, math.Max(sisi2,
    sisi3)))
}
```

Screenshoot program



```
1 package main
2
3 import (
4     "fmt"
5     "math"
6 )
7
8 func main() {
9     var x1, x2, x3, y1, y2, y3, sisi1, sisi2, sisi3 float64
10    fmt.Println("Input, 3 Titik Koordinat Segitiga (x,y) : ")
11    fmt.Scan(&x1)
12    fmt.Scanln(&y1)
13    fmt.Scan(&x2)
14    fmt.Scanln(&y2)
15    fmt.Scan(&x3)
16    fmt.Scanln(&y3)
17    sisi1 = math.Sqrt((math.Pow(x1-x2, 2)) + (math.Pow(y1-y2, 2)))
18    sisi2 = math.Sqrt((math.Pow(x2-x3, 2)) + (math.Pow(y2-y3, 2)))
19    sisi3 = math.Sqrt((math.Pow(x1-x3, 2)) + (math.Pow(y1-y3, 2)))
20    fmt.Printf("%.2f", math.Max(sisi1, math.Max(sisi2, sisi3)))
21 }
```

Terminal Output:

```
Input, 3 Titik Koordinat Segitiga (x,y) :
0.0 0.0
3.0 0.0
3.0 4.0
5.00
PS D:\Coding\Go\modul-4> go run soal3.go
Input, 3 Titik Koordinat Segitiga (x,y) :
1.0 1.0
4.0 1.0
1.0 5.0
5.00
```

Deskripsi program

Program ini berfungsi untuk menghitung dan menentukan sisi terpanjang dari sebuah segitiga berdasarkan koordinat tiga titiknya (x, y). Program ini menggunakan library “math” untuk menghitung hasil kuadrat dan akar kuadrat.

Program dimulai dengan mendeklarasikan variabel bertipe data bilangan desimal (float64) untuk menyimpan koordinat keenam titik yakni variabel x1, y1, x2, y2, x3, y3 dan panjang dari ketiga sisi yaitu, sisi1, sisi2, sisi3. Program kemudian meminta memasukkan koordinat dari ketiga titik segitiga sesuai dengan urutan pada baris pertama x1, y1, pada baris kedua x2, y2, pada baris ketiga x3, y3. Kemudian masing-masing input koordniat akan disimpan ke dalam variabel yang sesuai.

Setelah itu, pertama program menghitung panjang setiap sisi menggunakan rumus jarak antara dua titik (Teorema Pythagoras, $sisi = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$), pada program dituliskan dengan, `math.Sqrt((math.Pow(x1-x2, 2)) + (math.Pow(y1-y2, 2)))`. Proses ini menggunakan fungsi `math.Pow` untuk pemangkatan dan `math.Sqrt` untuk akar kuadrat. Setelah panjang sisi1, sisi2, dan sisi3 dihitung, program mencari nilai maksimum di antara ketiganya menggunakan fungsi `math.Max` (fungsi yang akan mengambil nilai terbesar dari dua value yang dibandingkan. Dengan membandingkan sisi2 dan sisi3 kemudian hasilnya akan dibandingkan dengan sisi1 sehingga didapatkan satu sisi terpanjang. Terakhir, program menampilkan panjang sisi terpanjang dengan format yang dibulatkan menjadi dua angka di belakang koma.