

**LAPORAN PRAKTIKUM ALGORITMA  
DAN PEMROGRAMAN 1**

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



**Disusun oleh:**

**Dharma Chandra Viriya**

**109082500052**

**S1IF-13-02**

**Asisten Praktikum**

Adithana dharma putra

Alfin Ilham Berlianto

**PROGRAM STUDI S1 INFORMATIKA  
FAKULTAS INFORMATIKA  
TELKOM UNIVERSITY PURWOKERTO**

**2025**

## LATIHAN KELAS – GUIDED

### 1. Guided 1 - Konversi Detik ke Jam, Menit, Detik

#### Source Code

```
package main

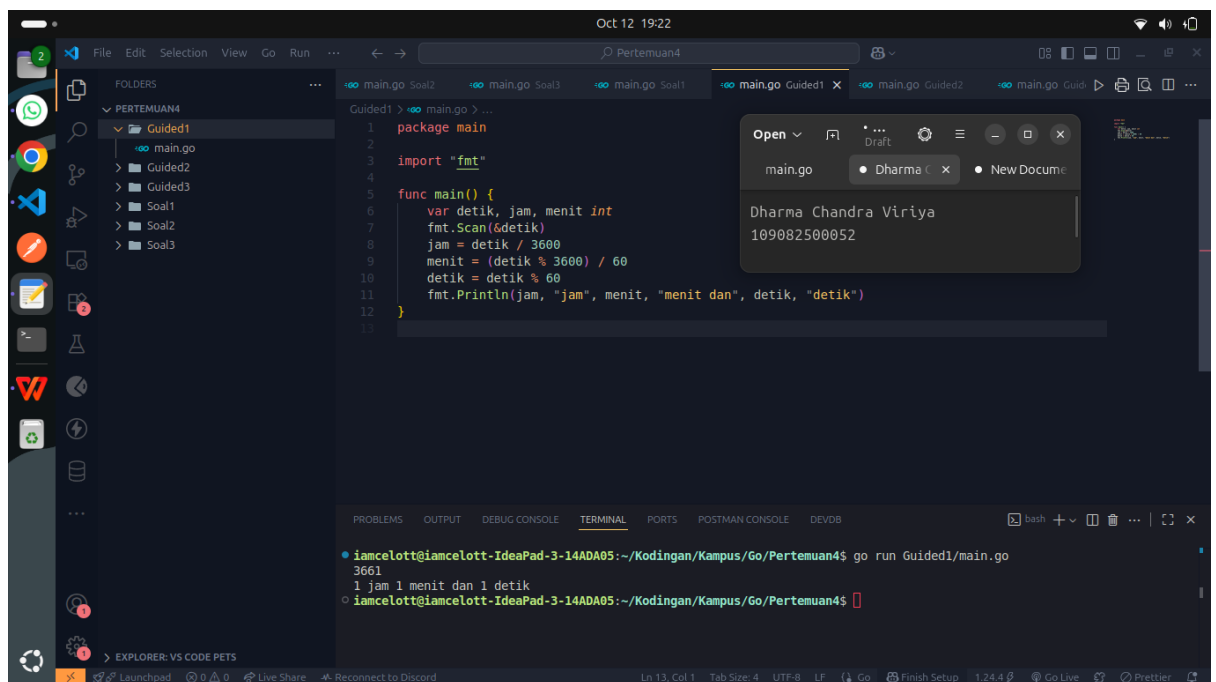
import "fmt"

func main() {
    var detik, jam, menit 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



### Deskripsi program

Kode program di atas merupakan kode program menghitung detik menjadi jam, menit, dan detik, dimana pada awal kode akan mendeklarasi kan package dengan inisialisasi main, lalu mengimpor library "fmt" dari Go yang digunakan untuk menangani input dan output. Kemudian, pada function main, mendeklarasikan variable detik jam, menit dengan tipe data int, lalu membuat kode program untuk menerima input dari user menggunakan fmt.Scan yang akan dimasukkan ke dalam variable detik, lalu pada bawah kode nya akan mengisikan variable jam dengan rumus detik / 3600, lalu mengisikan variable menit dengan rumus (detik % 3600) / 60, lalu mengisikan variable detik dengan rumus detik % 60, kemudian pada bawah kode nya akan mencetak hasil operasi ke dalam console menggunakan fmt.Println.

## 2. Guided 2 - Menentukan Digit Angka Semakin Membesar

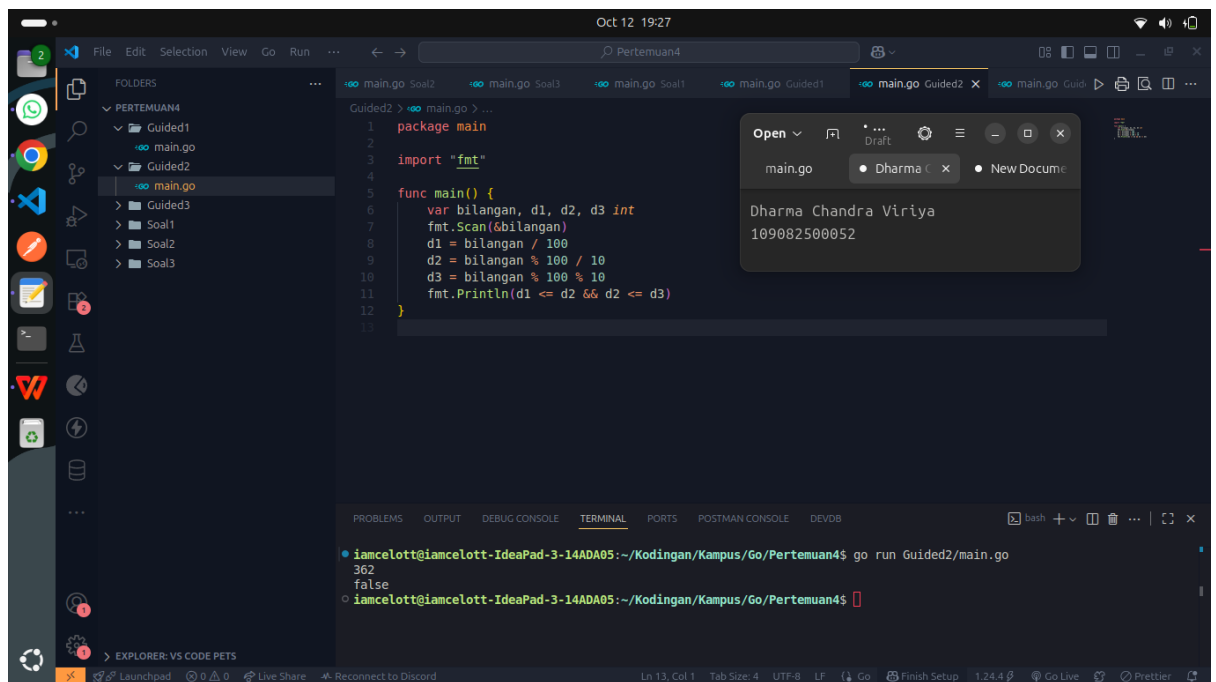
### Source Code

```
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



```
1 package main
2
3 import "fmt"
4
5 func main() {
6     var bilangan, d1, d2, d3 int
7     fmt.Scan(&bilangan)
8     d1 = bilangan / 100
9     d2 = bilangan % 100 / 10
10    d3 = bilangan % 100 % 10
11    fmt.Println(d1 <= d2 && d2 <= d3)
12 }
```

```
iamcelott@iamcelott-IdeaPad-3-14ADA05:~/Kodingan/Kampus/Go/Pertemuan4$ go run Guided2/main.go
362
false
iamcelott@iamcelott-IdeaPad-3-14ADA05:~/Kodingan/Kampus/Go/Pertemuan4$
```

### Deskripsi program

Kode program di atas berfungsi untuk pengecekan apakah setiap digit bilangan angka semakin membesar atau tidak. Di bagian awal, program mendeklarasikan package utama dengan nama main, lalu mengimpor library "fmt" dari Go yang digunakan untuk menangani input dan output. Kemudian, pada function main, mendeklarasikan variable bilangan, d1, d2, dan d3 dengan tipe data int, lalu membuat kode program untuk menerima input dari user menggunakan fmt.Scan yang akan dimasukkan ke dalam variable bilangan, lalu pada bawah kode nya akan mengisikan variable d1 dengan rumus  $\text{bilangan} / 100$ , lalu mengisikan variable d2 dengan rumus  $\text{bilangan} \% 100 / 10$ , lalu mengisikan variable d3 dengan rumus  $\text{bilangan} \% 100 \% 10$ , kemudian pada akhir kode akan melakukan pengecekan di dalam fmt.Println apakah setiap variable digit(d1, d2, d3) semakin membesar atau tidak dengan hasil akhir true/false

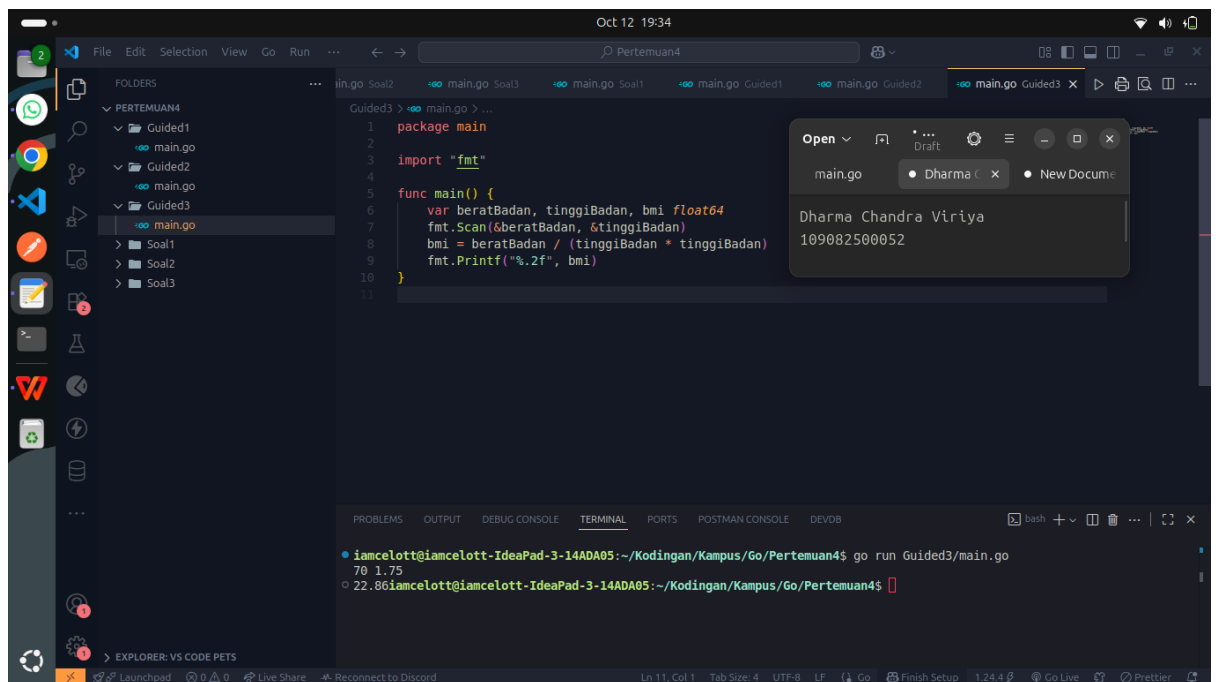
### 3. Guided 3 - Hitung BMI Source Code

```
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

Kode program di atas digunakan untuk menghitung BMI. Pada bagian awal, program mendeklarasikan package utama dengan inisialisasi main, lalu mengimpor library "fmt" dari Go yang berfungsi untuk menangani input dan output. Kemudian, pada function main, mendeklarasikan variable beratBadan, tinggiBadan, dan bmi dengan tipe data float64, lalu membuat kode program untuk menerima input dari user menggunakan fmt.Scan yang akan dimasukkan ke dalam variable beratBadan dan tinggiBadan, lalu pada bawah kode nya mendeklarasikan variable bmi dengan inisialisasi rumus Perhitungan BMI yaitu beratBadan / (tinggiBadan \* tinggiBadan), lalu hasil dari Perhitungan BMI tersebut akan dicetak ke dalam console menggunakan fmt.Println.

## TUGAS

### 1. Tugas 1 - Hitung Diskon

#### Source code

```
package main

import "fmt"

func HitungDiskon(total_harga_awal, diskon int) int {
    harga_diskon := (total_harga_awal * diskon) / 100
    return total_harga_awal - harga_diskon
}

func main() {
    var total_harga_awal, diskon int

    fmt.Print("Masukkan total harga barang: ")
    fmt.Scan(&total_harga_awal)
    fmt.Print("Masukkan diskon: ")
    fmt.Scan(&diskon)

    result := HitungDiskon(total_harga_awal, diskon)
    fmt.Println("Total harga akhir: ", result)
}
```

#### Screenshoot program

```
1 package main
2
3 import "fmt"
4
5 func HitungDiskon(total_harga_awal, diskon int) int {
6     harga_diskon := (total_harga_awal * diskon) / 100
7     return total_harga_awal - harga_diskon
8 }
9
10 func main() {
11     var total_harga_awal, diskon int
12
13     fmt.Print("Masukkan total harga barang: ")
14     fmt.Scan(&total_harga_awal)
15
16     fmt.Print("Masukkan diskon: ")
17     fmt.Scan(&diskon)
18
19     result := HitungDiskon(total_harga_awal, diskon)
20
21     fmt.Println("Total harga akhir: ", result)
22 }
23
```

```
iamcelott@iamcelott-IdeaPad-3-14ADA05:~/Kodingan/Kampus/Go/Pertemuan4$ go run Soal1/main.go
Masukkan total harga barang: 100000
Masukkan diskon: 10
Total harga akhir: 90000
iamcelott@iamcelott-IdeaPad-3-14ADA05:~/Kodingan/Kampus/Go/Pertemuan4$
```

## Deskripsi program

Kode program di atas berfungsi untuk menghitung harga akhir setelah mendapatkan diskon. Pada bagian awal, program mendeklarasikan package utama dengan inisialisasi main, lalu mengimpor library "fmt" dari Go yang berfungsi untuk menangani input dan output. Selanjutnya mendeklarasi kan function dengan nama HitungDiskon yang menerima parameter variable total\_harga\_awal, dan diskon dengan tipe data int, lalu isi dari function tersebut adalah, mendeklarasikan variable harga\_diskon dengan isi yaitu  $(total\_harga\_awal * diskon) / 100$ , kemudian akan mengembalikan (return) hasil dari total\_harga\_awal - harga\_diskon. Kemudian, pada function main, mendeklarasikan variable total\_harga\_awal, diskon dengan tipe data int, lalu membuat kode program untuk menerima input dari user menggunakan fmt.Scan yang akan dimasukkan ke dalam variable total\_harga\_awal & diskon, lalu pada bawah kode nya mendeklarasikan variable result dengan inisialisasi pemanggilan function HitungDiskon dan mengisikan parameter total\_harga\_awal, dan diskon dengan variable total\_harga\_awal, dan diskon, lalu hasil dari pemanggilan function MenyatakanX yang disimpan ke dalam variable result tersebut akan dicetak ke dalam console menggunakan fmt.Println.

## 2. Tugas 2 - Hitung Berat Badan

### Source code

```
package main
```

```
import (  
    "fmt"  
    "math"  
)  
  
func HitungBeratBadan(bmi, tb float64) float64 {  
    return bmi * math.Pow(tb, 2)  
}  
  
func main() {  
    var bmi, tb float64  
  
    fmt.Print("Masukkan BMI: ")  
    fmt.Scan(&bmi)  
    fmt.Print("Masukkan Tinggi Badan: ")  
    fmt.Scan(&tb)  
  
    result := HitungBeratBadan(bmi, tb)  
  
    fmt.Println(math.Ceil(result))  
}
```

**Screenshoot program**



```
5 | "math"
6 | }
7 |
8 | func HitungBeratBadan(bmi, tb float64) float64 {
9 |     return bmi * math.Pow(tb, 2)
10 | }
11 |
12 | func main() {
13 |     var bmi, tb float64
14 |
15 |     fmt.Print("Masukkan BMI: ")
16 |     fmt.Scan(&bmi)
17 |
18 |     fmt.Print("Masukkan Tinggi Badan: ")
19 |     fmt.Scan(&tb)
20 |
21 |     result := HitungBeratBadan(bmi, tb)
22 |
23 |     fmt.Println(math.Ceil(result))
24 | }
25 |
```

```
iamcelott@iamcelott-IdeaPad-3-14ADA05:~/Kodingan/Kampus/Go/Pertemuan4$ go run Soal2/main.go
Masukkan BMI: 22.85 1.75
Masukkan Tinggi Badan: 70
109082500052
```

### Deskripsi program

Kode program di atas berfungsi untuk menghitung Berat Badan pengguna yang diketahui nilai BMI nya. Pada bagian awal, program mendeklarasikan package utama dengan inisialisasi main, Pada bagian awal, program mendeklarasikan package utama dengan nama main, kemudian mengimpor dua library bawaan Go, yaitu "fmt" untuk menangani input dan output, serta "math" untuk menggunakan fungsi-fungsi matematika. Selanjutnya mendeklarasi kan function dengan nama HitungBeratBadan yang menerima parameter variable bmi & tb dengan tipe data float64, lalu isi dari function tersebut akan mengembalikan (return) hasil dari operasi matematika yaitu  $bmi * math.Pow(tb, 2)$ , math.Pow adalah fungsi math yang fungsi nya untuk menghitung pangkat, seperti case diatas adalah tb di pangkat kan 2. Kemudian, pada function main, mendeklarasikan variable bmi & tb dengan tipe data float64, kemudian membuat kode program untuk menerima input dari user menggunakan fmt.Scan yang akan dimasukkan ke dalam variable bmi & tb, lalu pada bawah kode nya mendeklarasikan variable result dengan inisialisasi pemanggilan function HitungBeratBadan dan mengisikan parameter bmi & tb dengan variable bmi & tb. Lalu, hasil dari pemanggilan function HitungBeratBadan yang telah disimpan ke dalam variable result tersebut akan dicetak ke dalam console menggunakan fmt.Println dengan isi math.Ceil result agar variable result akan dibulatkan ke atas.

### 3. Tugas 3 - Hitung Panjang Sisi Segitiga

#### Source code

```
package main
```

```
import (  
    "fmt"  
    "math"  
    "strconv"  
    "strings"  
)  
  
func HitungSisiP(ax, ay, bx, by, cx, cy float64)  
float64 {  
    ab := math.Sqrt(math.Pow(bx-ax, 2) + math.Pow(by-  
ay, 2))  
    bc := math.Sqrt(math.Pow(cx-bx, 2) + math.Pow(cy-  
by, 2))  
    ca := math.Sqrt(math.Pow(cx-ax, 2) + math.Pow(cy-  
ay, 2))  
  
    return math.Max(ab, math.Max(bc, ca))  
}  
  
func main() {  
    var ax, ay, bx, by, cx, cy float64  
  
    fmt.Print("Masukan nilai Ax : ")  
    fmt.Scanln(&ax)  
    fmt.Print("Masukan nilai Ay : ")  
    fmt.Scanln(&ay)  
    fmt.Print("Masukan nilai Bx : ")  
    fmt.Scanln(&bx)  
    fmt.Print("Masukan nilai by : ")  
    fmt.Scanln(&by)  
    fmt.Print("Masukan nilai Cx : ")  
    fmt.Scanln(&cx)
```

```

        fmt.Print("Masukan nilai Cy : ")

        fmt.Scanln(&cy)

        result := HitungSisiP(ax, ay, bx, by, cx, cy)

        splitResult := strings.Split(fmt.Sprintf("%.2f",
result), ".")

        index1, _ := strconv.ParseFloat(splitResult[0],
64)

        index2, _ := strconv.ParseFloat(splitResult[1],
64)

        if index2 != 0 {

            fmt.Printf("%.2f\n", result)

        } else {

            fmt.Println(index1)

        }

    }

```

## Screenshoot program

The screenshot shows a VS Code editor with a Go program in a file named `main.go`. The program defines a function `HitungSisiP` that calculates the hypotenuse of a right-angled triangle given two sides. It then uses this function to calculate the maximum side of a triangle given three sides. The terminal shows the program being run with input values 1.0, 1.0, 3.0, 1.0, 1.0, 5.0, resulting in an output of 4.47.

```

package main

import (
    "fmt"
    "math"
    "strconv"
    "strings"
)

func HitungSisiP(ax, ay, bx, by, cx, cy float64) float64 {
    ab := math.Sqrt(math.Pow(bx-ax, 2) + math.Pow(by-ay, 2))
    bc := math.Sqrt(math.Pow(cx-bx, 2) + math.Pow(cy-by, 2))
    ca := math.Sqrt(math.Pow(cx-ax, 2) + math.Pow(cy-ay, 2))

    return math.Max(ab, math.Max(bc, ca))
}

func main() {
    var ax, ay, bx, by, cx, cy float64

    fmt.Print("Masukan nilai Ax : ")
    fmt.Scanln(&ax)

    fmt.Print("Masukan nilai Ay : ")
    fmt.Scanln(&ay)

    fmt.Print("Masukan nilai Bx : ")
    fmt.Scanln(&bx)

    fmt.Print("Masukan nilai By : ")
    fmt.Scanln(&by)

    fmt.Print("Masukan nilai Cx : ")
    fmt.Scanln(&cx)

    fmt.Print("Masukan nilai Cy : ")
    fmt.Scanln(&cy)

    result := HitungSisiP(ax, ay, bx, by, cx, cy)

    splitResult := strings.Split(fmt.Sprintf("%.2f",
result), ".")

    index1, _ := strconv.ParseFloat(splitResult[0],
64)

    index2, _ := strconv.ParseFloat(splitResult[1],
64)

    if index2 != 0 {

        fmt.Printf("%.2f\n", result)

    } else {

        fmt.Println(index1)

    }
}

```

The terminal output shows the program being run with input values 1.0, 1.0, 3.0, 1.0, 1.0, 5.0, resulting in an output of 4.47.

```

iamcelott@iamcelott-IdeaPad-3-14ADA05:~/Kodingan/Kampus/Go/Pertemuan4$ go run Soal3/main.go
Masukan nilai Ax : 1.0
Masukan nilai Ay : 1.0
Masukan nilai Bx : 3.0
Masukan nilai By : 1.0
Masukan nilai Cx : 1.0
Masukan nilai Cy : 5.0
4.47

```

### Deskripsi program

Kode program di atas berfungsi untuk menghitung panjang sisi segitiga. Pada bagian awal, program mendeklarasikan package utama dengan inisialisasi main, lalu mengimpor library "fmt" dari Go yang berfungsi untuk menangani input dan output, serta "math" untuk menggunakan fungsi-fungsi matematika, serta strconv untuk menggunakan fungsi-fungsi convert string, dan juga strings untuk menggunakan fungsi-fungsi yang ada pada tipe data string. Selanjutnya mendeklarasikan function dengan nama HitungSisiP yang menerima parameter variable ax, ay, bx, by, cx, cy dengan tipe data float64, lalu isi dari function tersebut adalah mendeklarasikan variable ab dengan inisialisasi  $\text{math.Sqrt}(\text{math.Pow}(\text{bx}-\text{ax}, 2) + \text{math.Pow}(\text{by}-\text{ay}, 2))$ , lalu mendeklarasikan lagi variable bc dengan inisialisasi  $\text{math.Sqrt}(\text{math.Pow}(\text{cx}-\text{bx}, 2) + \text{math.Pow}(\text{cy}-\text{by}, 2))$ , lalu mendeklarasikan lagi variable ca dengan inisialisasi  $\text{math.Sqrt}(\text{math.Pow}(\text{cx}-\text{ax}, 2) + \text{math.Pow}(\text{cy}-\text{ay}, 2))$ , kemudian akan mengembalikan(return)  $\text{math.Max}(\text{ab}, \text{math.Max}(\text{bc}, \text{ca}))$ , fungsi math.Sqrt adalah fungsi matematika untuk menghitung akar dari argumen yang diisikan, dan math.Max adalah kode untuk membandingkan 2 argumen mana yang lebih besar untuk di return. Kemudian, pada function main, mendeklarasikan variable ax, ay, bx, by, cx, cy dengan tipe data float64, lalu membuat kode program untuk menerima input dari user menggunakan fmt.Scan yang akan dimasukkan ke dalam variable ax, ay, bx, by, cx, cy, lalu pada bawah kode nya mendeklarasikan variable result dengan inisialisasi pemanggilan function HitungSisiP, lalu dibawah nya mendeklarasikan splitResult dengan inisialisasi value dari variable result yang di konversi menjadi string kemudian di split(pecah) dengan trigger ".", kemudian hasil dari pemecahan string tersebut akan menghasilkan 2 buah index yaitu index1 merupakan angka bagian sisi kiri sebelum titik dan index2 yang merupakan bagian sisi kanan setelah titik, kemudian index1 dan index2 tersebut di parsing lagi menjadi float64. Lalu, melakukan pengecekan jika index2 tidak sama dengan 0 berarti variable result merupakan bilangan pecahan maka cetak ke dalam console menggunakan fmt.Printf dengan verb %2.f\n. Namun, jika  $\text{index2} == 0$ , itu berarti value dari variable result tersebut merupakan sebuah bilangan bulat dengan angka 0 setelah titik, maka akan menjalankan fmt.Println index1 yang isi nya hanya angka bulat

Contoh:

result = 4.47

fmt.Print = 4.47

result = 5.0

fmt.Print = 5