

**LAPORAN PRAKTIKUM ALGORITMA
DAN PEMROGRAMAN 1**

**MODUL 4
I/O TIPE DATA DAN VARIABEL**



Disusun oleh:

JOSHUA NATHANAEL

109082530033

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, jam, menit int

    fmt.Scan(&detik)

    jam = detik / 3600

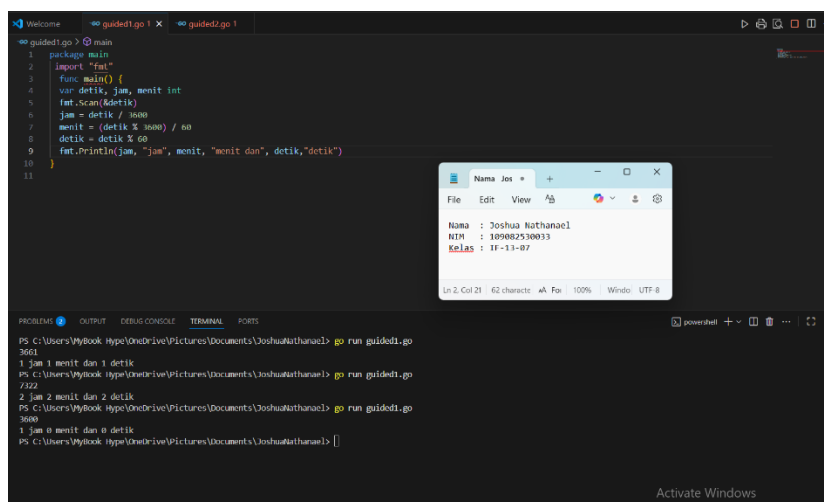
    menit = (detik % 3600) / 60

    detik = detik % 60

    fmt.Println(jam, "jam", menit, "menit dan", detik,"detik")

}
```

Screenshoot program



Deskripsi program:

Program ini dibuat untuk mengubah waktu yang awalnya dalam bentuk detik menjadi jam, menit, dan detik supaya lebih mudah dipahami. Dengan cukup memasukkan jumlah detik, lalu program otomatis menghitung berapa jam, menit, dan sisa detiknya. Setelah itu, akan menampilkan hasil dengan format sederhana seperti "x jam y menit dan z detik".

2. Guided 2

Source Code

```
package main

import "fmt"

func main() {

    var bilangan, d1, d2, d3 int

    fmt.Scan(&bilangan)

    d1 = bilangan / 100

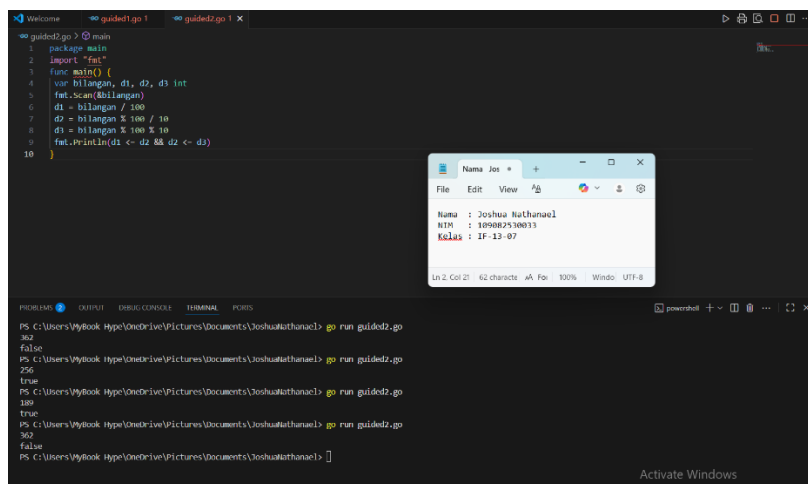
    d2 = bilangan % 100 / 10

    d3 = bilangan % 100 % 10

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

}
```

Screenshoot program



Deskripsi program:

Program dibuat untuk mengecek apakah angka tiga digit yang kita masukan tersusun secara naik atau turun. Jadi, saat kita mengetikkan sebuah angka, program akan memisahkan angka tersebut menjadi ratusan, puluhan, dan satuan. Program akan membandingkan apakah digit pertama lebih kecil atau sama dengan digit ke dua dan seterusnya.

3. Guided 3

Source Code

```
package main

import "fmt"

func main() {

    var beratBadan, tinggiBadan, bmi float64
```

```

    fmt.Scan(&beratBadan, &tinggiBadan)

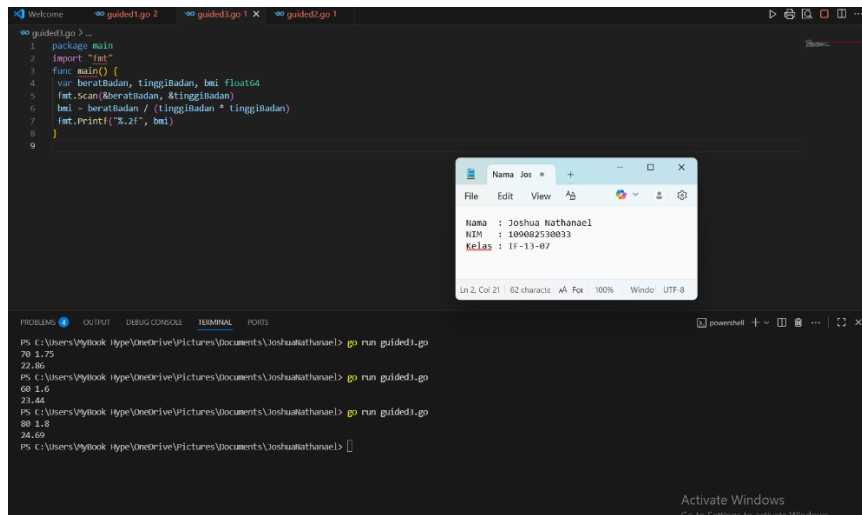
    bmi = beratBadan / (tinggiBadan * tinggiBadan)

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

}

```

Screenshoot program



Deskripsi program:

Program tersebut berfungsi untuk menghitung nilai body massa index dengan cara memasukan berat badan dan tinggi badan dan biasanya digunakan untuk mengetahui apakah seseorang tergolong kurus, ideal, atau kelebihan berat badan.

TUGAS

1. Tugas 1

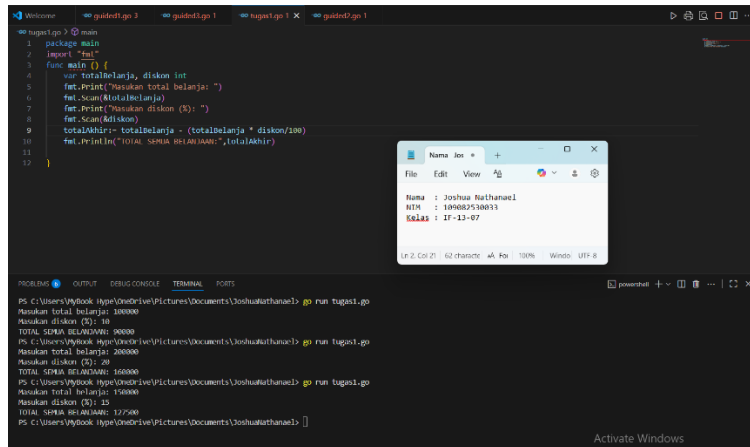
Source code

```

package main
import "fmt"
func main () {
    var totalBelanja, diskon int
    fmt.Print("Masukan total belanja: ")
    fmt.Scan(&totalBelanja)
    fmt.Print("Masukan diskon (%): ")
    fmt.Scan(&diskon)
    totalAkhir:= totalBelanja - (totalBelanja * diskon/100)
    fmt.Println("TOTAL SEMUA BELANJAAN:", totalAkhir)
}

```

Screenshoot program



Deskripsi program:

Dibuat untuk menghitung total akhir belanja setelah mendapatkan diskon, dengan cara menghitung potongan harga dengan rumus $\text{total belanja} \times \text{diskon} / 100$ lalu mengurangkannya dengan total belanja awal dan akan mengeluarkan

2. Tugas 2

Source code

```
package main

import "fmt"

func main() {

    var BMI, TINGGIBADAN, BERATBADAN float64

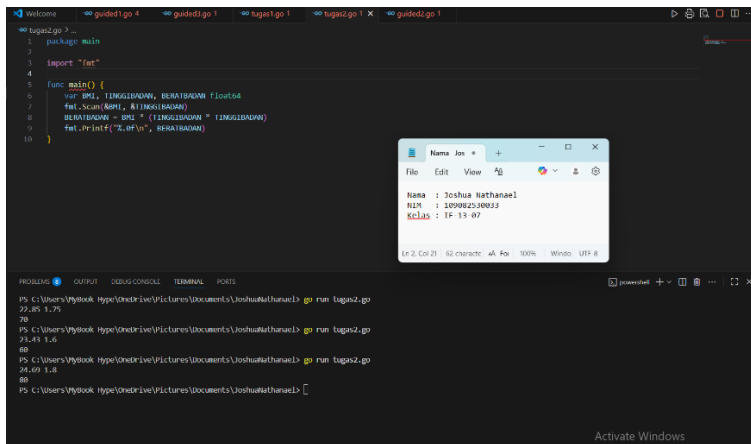
    fmt.Scan(&BMI, &TINGGIBADAN)

    BERATBADAN = BMI * (TINGGIBADAN * TINGGIBADAN)

    fmt.Printf("%.0f\n", BERATBADAN)

}
```

Screenshoot program



Deskripsi program:

Dibuat untuk menghitung berat badan berdasarkan nilai BMI dan tinggi badan program akan menghitung dengan rumus BMI program ini bias membantu memperkirakan berat badan ideal seseorang

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)

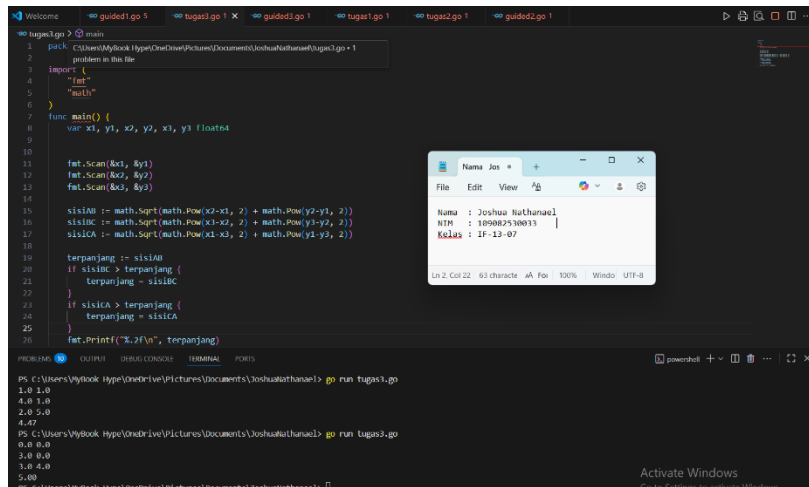
    sisiAB := math.Sqrt(math.Pow(x2-x1, 2) + math.Pow(y2-y1, 2))
    sisiBC := math.Sqrt(math.Pow(x3-x2, 2) + math.Pow(y3-y2, 2))
    sisiCA := math.Sqrt(math.Pow(x1-x3, 2) + math.Pow(y1-y3, 2))

    terpanjang := sisiAB
    if sisiBC > terpanjang {
        terpanjang = sisiBC
    }
    if sisiCA > terpanjang {
        terpanjang = sisiCA
    }
    fmt.Printf("%.2f\n", terpanjang)
}

```

```
}
```

Screenshoot program



The screenshot shows a Go program in VS Code and its execution output in the terminal. The program calculates the longest side of a triangle given three points (x1, y1), (x2, y2), and (x3, y3). It uses the distance formula to calculate the lengths of the three sides and then compares them to find the longest one.

```
1 package main
2 import (
3     "fmt"
4     "math"
5 )
6 func main() {
7     var x1, y1, x2, y2, x3, y3 float64
8
9     fmt.Scan(&x1, &y1)
10    fmt.Scan(&x2, &y2)
11    fmt.Scan(&x3, &y3)
12
13    sisiAB := math.Sqrt(math.Pow(x2-x1, 2) + math.Pow(y2-y1, 2))
14    sisiBC := math.Sqrt(math.Pow(x3-x2, 2) + math.Pow(y3-y2, 2))
15    sisiCA := math.Sqrt(math.Pow(x1-x3, 2) + math.Pow(y1-y3, 2))
16
17    terpanjang := sisiAB
18    if sisiBC > terpanjang {
19        terpanjang = sisiBC
20    }
21    if sisiCA > terpanjang {
22        terpanjang = sisiCA
23    }
24    fmt.Printf("%.2f\n", terpanjang)
25 }
```

The terminal output shows the program being run twice. The first run takes input 1.0 1.0, 4.0 1.0, 2.0 5.0 and outputs 4.47. The second run takes input 0.0 0.0, 3.0 0.0, 3.0 4.0 and outputs 5.00.

```
PS C:\Users\Vytbook\Hyperdrive\Pictures\Documents\JoshuaHathanuel> go run tugas3.go
1.0 1.0
4.0 1.0
2.0 5.0
4.47
PS C:\Users\Vytbook\Hyperdrive\Pictures\Documents\JoshuaHathanuel> go run tugas3.go
0.0 0.0
3.0 0.0
3.0 4.0
5.00
```

Deskripsi program:

Program dibuat untuk mencari sisi terpanjang dari sebuah segitiga dengan memasukan koordinat tiga titik (x1,y1);(x2,y2);(x3,y3), lalu akan di hitung Panjang masing masing sisi segitiga menggunakan rumus jarak antara dua titik. lalu akan menampilkan Panjang sisi terpanjang tersebut dengan dua angka di belakang koma.