

**LAPORAN PRAKTIKUM**  
**Algoritma Pemrograman**

**MODUL 4**  
**I/O, Tipe Data & Variabel**



**Disusun oleh:**  
**DADI MAULANA MUHAMMAD**  
**109082500080**  
**S1IF-13-04**

**PROGRAM STUDI S1 INFORMATIKA**  
**FAKULTAS INFORMATIKA**  
**TELKOM UNIVERSITY PURWOKERTO**  
**2025**

## LATIHAN KELAS – GUIDED

### 1. Guided 1

#### Source Code

```
//SALIN KODE KESINI

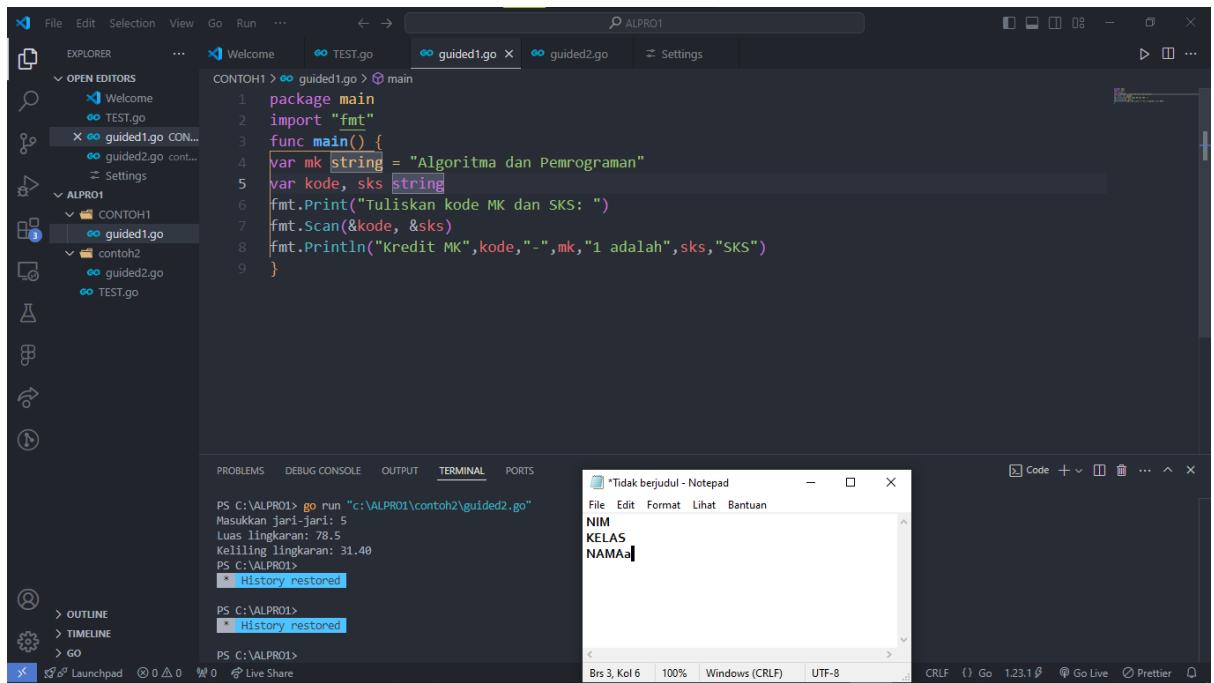
ATURAN:

PENULISAN SEESUAI MODUL 1

GUNAKAN FONT Courier New ukuran 11pt dengan spasi baris
dan paragraf 1,5
```

#### Screenshot program

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



The screenshot shows a code editor interface with a dark theme. On the left is the Explorer sidebar showing a project structure with files like 'Welcome', 'TEST.go', 'guided1.go', 'guided2.go', and 'contoh2'. The main editor area displays the following Go code:

```
package main
import "fmt"
func main() {
    var mk string = "Algoritma dan Pemrograman"
    var kode, sks string
    fmt.Println("Tuliskan kode MK dan SKS: ")
    fmt.Scan(&kode, &sks)
    fmt.Println("Kredit MK", kode, "-", mk, "1 adalah", sks, "SKS")
}
```

Below the editor is a terminal window showing command-line output:

```
PS C:\ALPRO1> go run "c:\ALPRO1\contoh2\guided2.go"
Masukkan jari-jari: 5
Luas lingkaran: 78.5
Keiling lingkaran: 31.40
PS C:\ALPRO1>
* History restored

PS C:\ALPRO1>
* History restored
PS C:\ALPRO1>
```

To the right of the terminal is a small Notepad window titled 'Tidak berjudul - Notepad' containing the text:

```
NIM
KELAS
NAMAa|
```

#### Deskripsi program

Jelaskan kode yang ada di source code, semakin detil semakin baik nilainya

## 2. Guided 2

### 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")
}
```

### Screenshot program

The screenshot shows a code editor window with the file named `guided-01.go`. The code implements a function to convert seconds into hours, minutes, and seconds. The editor interface includes tabs for PROBLEMS, OUTPUT, DEBUG CONSOLE, TERMINAL, and PORTS. The TERMINAL tab shows the command `go run "f:\Me\Others\Kuliah\Algoritma-dan-Pemrograman\week-04"` being run twice, resulting in the output "1 jam 1 menit dan 1 detik" and "2 jam 2 menit dan 2 detik". The DEBUG CONSOLE tab displays the user's profile information: Nama : Dadi Maulana Muhammad, NIM : 109082500080, and Kelas : S1IF-13-04.

```
guided-01.go
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")
```

PROBLEMS 4 OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS F:\Me\Others\Kuliah\Algoritma-dan-Pemrograman\week-04> go run "f:\Me\Others\Kuliah\Algoritma-dan-Pemrograman\week-04"
3661
1 jam 1 menit dan 1 detik
PS F:\Me\Others\Kuliah\Algoritma-dan-Pemrograman\week-04> go run "f:\Me\Others\Kuliah\Algoritma-dan-Pemrograman\week-04"
7322
2 jam 2 menit dan 2 detik
PS F:\Me\Others\Kuliah\Algoritma-dan-Pemrograman\week-04> go run "f:\Me\Others\Kuliah\Algoritma-dan-Pemrograman\week-04"
3600
1 jam 0 menit dan 0 detik
PS F:\Me\Others\Kuliah\Algoritma-dan-Pemrograman\week-04>

Nama : Dadi Maulana Muhammad  
NIM : 109082500080  
Kelas : S1IF-13-04

Ln 12, Col 1 69 character Plain text 130% Wind UTF-8

### **Deskripsi program**

Program ini meminta pengguna untuk menginput satu buah bilangan bulat yang merepresentasikan jumlah detik. Nilai input tersebut disimpan dalam variabel detik. Setelah itu, program menghitung konversi waktu dari detik tersebut menjadi jam, menit, dan detik dengan langkah sebagai berikut:

Nilai jam diperoleh dari hasil pembagian detik / 3600.

Nilai menit diperoleh dari sisa pembagian (detik % 3600) / 60.

Nilai detik kemudian diperbarui menjadi detik % 60 untuk mendapatkan sisa detik setelah dikonversi ke jam dan menit.

Setelah perhitungan selesai, program menampilkan hasil konversi waktu tersebut ke layar dalam format:

x jam y menit dan z detik,

di mana x, y, dan z adalah hasil perhitungan jam, menit, dan detik yang sesuai.

### **3. Guided 3**

#### **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

```
guided-02.go
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)
}
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS F:\Me\Others\Kuliah\Algoritma-dan-Pemrograman\week-04> go run "f:\Me\Others\Kuliah\Algoritma-dan-Pemrograman\week-04\guided-02.go"
362
false
PS F:\Me\Others\Kuliah\Algoritma-dan-Pemrograman\week-04> go run "f:\Me\Others\Kuliah\Algoritma-dan-Pemrograman\week-04\guided-02.go"
256
true
PS F:\Me\Others\Kuliah\Algoritma-dan-Pemrograman\week-04> go run "f:\Me\Others\Kuliah\Algoritma-dan-Pemrograman\week-04\guided-02.go"
189
true
PS F:\Me\Others\Kuliah\Algoritma-dan-Pemrograman\week-04>
```

Nama : Dadi Maulana Muhammad  
NIM : 109082500080  
Kelas : S1IF-13-04

Ln 3, Col 19 69 character Plain text 130% Wind UTF-8  
Ln 12, Col 1 Tab Size: 2 UTF-8 CRLF { } Go ⚡ 1.25.1 Go Live Q

## Deskripsi program

Program ini meminta pengguna untuk menginput satu bilangan bulat tiga digit, kemudian memeriksa apakah setiap digit dalam bilangan tersebut tersusun secara **menaik (ascending)** dari kiri ke kanan.

Nilai input disimpan dalam variabel bilangan. Program kemudian memisahkan masing-masing digit bilangan tersebut menjadi tiga variabel:

d1 menyimpan digit ratusan dengan rumus bilangan / 100,

d2 menyimpan digit puluhan dengan rumus (bilangan % 100) / 10,

d3 menyimpan digit satuan dengan rumus bilangan % 10.

Selanjutnya, program mengevaluasi kondisi logika `d1 <= d2 && d2 <= d3` untuk memeriksa apakah digit-digit bilangan tersusun secara menaik.

Hasil evaluasi berupa nilai **true** jika urutan digit menaik, atau **false** jika tidak.

Terakhir, hasil logika tersebut ditampilkan ke layar menggunakan `fmt.Println()`.

#### 4. Guided 4

##### 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)
}
```

##### Screenshot program

The screenshot shows a code editor interface with a dark theme. On the left, there is a file tree with a single file named 'guided-03.go'. The code itself is a simple Go program that calculates the Body Mass Index (BMI) based on weight and height input from the user. The code uses the standard library's `fmt` package for input and output. The editor has tabs for PROBLEMS, OUTPUT, DEBUG CONSOLE, TERMINAL, and PORTS. The TERMINAL tab is active, showing the command `go run "f:\Me\Others\Kuliah\Algoritma-dan-Pemrograman\week-04"` followed by two lines of output: `70 1.75` and `22.86`. The DEBUG CONSOLE tab shows the user's input and the calculated BMI value. The OUTPUT tab shows the final output of the program, which includes the user's name, NIM, and class information.

```
PS F:\Me\Others\Kuliah\Algoritma-dan-Pemrograman\week-04> go run "f:\Me\Others\Kuliah\Algoritma-dan-Pemrograman\week-04"
70 1.75
22.86
PS F:\Me\Others\Kuliah\Algoritma-dan-Pemrograman\week-04> go run "f:\Me\Others\Kuliah\Algoritma-dan-Pemrograman\week-04"
60 1.6
23.44
PS F:\Me\Others\Kuliah\Algoritma-dan-Pemrograman\week-04> go run "f:\Me\Others\Kuliah\Algoritma-dan-Pemrograman\week-04"
80 1.8
24.69
PS F:\Me\Others\Kuliah\Algoritma-dan-Pemrograman\week-04>
```

Name : Dadi Maulana Muhammad
NIM : 109082500080
Kelas : S1IF-13-04

Ln 3, Col 19 69 character Plain text 130% Wind UTF-8  
Ln 10, Col 1 Tab Size: 2 UTF-8 CRLF {} Go ⌂ 1.25.1 ⌂ Go Live ⌂

### **Deskripsi program**

Program ini meminta pengguna untuk menginput **dua buah nilai bertipe float64**, yaitu **berat badan (dalam kilogram)** dan **tinggi badan (dalam meter)**. Nilai berat badan yang dimasukkan kemudian disimpan dalam variabel beratBadan, sedangkan tinggi badan disimpan dalam variabel tinggiBadan.

Setelah itu, program menghitung **Body Mass Index (BMI)** dengan menggunakan rumus:

$$\text{BMI} = \text{berat badan} / (\text{tinggi badan})^2$$

Hasil perhitungan disimpan dalam variabel bmi.

Terakhir, nilai BMI tersebut ditampilkan ke layar menggunakan format dua angka di belakang koma sebagai output

.

## TUGAS

### 1. Tugas 1

#### Source code

```
package main

import "fmt"

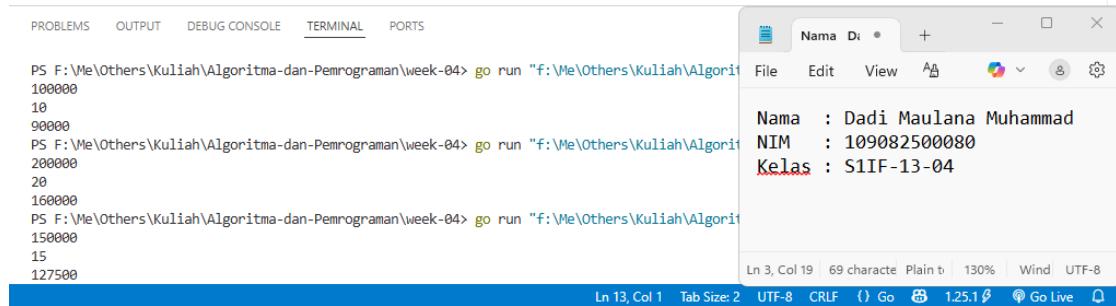
func main() {
    var totalBelanja, diskon int
    fmt.Scan(&totalBelanja)
    fmt.Scan(&diskon)

    totalAkhir := totalBelanja - (totalBelanja * diskon / 100)
    fmt.Println(totalAkhir)
}
```

#### Screenshot program



A screenshot of a code editor window titled "assignment-01.go". The code is identical to the one provided above, showing a Go program that calculates a discounted total based on user input for totalBelanja and diskon.



A screenshot of a terminal window showing the execution of the "assignment-01.go" program. The terminal output shows the user entering values for totalBelanja and diskon, and the program outputting the calculated totalAkhir. The terminal interface includes tabs for PROBLEMS, OUTPUT, DEBUG CONSOLE, TERMINAL, and PORTS, and a status bar at the bottom.

### Deskripsi program

Program ini meminta pengguna untuk menginput **dua buah bilangan integer**, yaitu **total belanja** dan **persentase diskon**.

Nilai total belanja yang dimasukkan disimpan dalam variabel totalBelanja, sedangkan nilai diskon disimpan dalam variabel diskon.

Selanjutnya, program menghitung **total akhir setelah diskon** menggunakan rumus:  
$$\text{totalAkhir} = \text{totalBelanja} - (\text{totalBelanja} \times \text{diskon}/100)$$

Hasil perhitungan tersebut kemudian disimpan dalam variabel totalAkhir.

Terakhir, program menampilkan nilai total akhir belanja ke layar sebagai output.

## 2. Tugas 2

### Source code

```
package main

import (
    "fmt"
    "math"
)

func main() {
    var bmi, tinggi float64
    fmt.Scan(&bmi, &tinggi)

    berat := bmi * tinggi * tinggi
    fmt.Println(int(math.Round(berat)))
}
```

## Screenshot program

```
assignment-02.go x
assignment-02.go > main
1 package main
2
3 import (
4     "fmt"
5     "math"
6 )
7
8 func main() {
9     var bmi, tinggi float64
10    fmt.Scan(&bmi, &tinggi)
11
12    berat := bmi * tinggi * tinggi
13    fmt.Println(int(math.Round(berat)))
14 }
15
```



PS F:\Me\Others\Kuliah\Algoritma-dan-Pemrograman\week-04> go run "f:\Me\Others\Kuliah\Algoritma-dan-Pemrograman\week-04\main.go"  
22.85 1.75  
70  
PS F:\Me\Others\Kuliah\Algoritma-dan-Pemrograman\week-04> go run "f:\Me\Others\Kuliah\Algoritma-dan-Pemrograman\week-04\main.go"  
23.43 1.6  
60  
PS F:\Me\Others\Kuliah\Algoritma-dan-Pemrograman\week-04> go run "f:\Me\Others\Kuliah\Algoritma-dan-Pemrograman\week-04\main.go"  
24.69 1.8  
80  
PS F:\Me\Others\Kuliah\Algoritma-dan-Pemrograman\week-04> █

File Edit View A:     

Nama : Dadi Maulana Muhammad  
NIM : 109082500080  
Kelas : S1IF-13-04

Ln 3, Col 19 69 character Plain text 130% Wind UTF-8

## Deskripsi program

Program ini meminta pengguna untuk menginput dua buah bilangan riil, yaitu BMI (Body Mass Index) dan tinggi badan dalam meter.

Nilai BMI disimpan ke dalam variabel bmi, sedangkan tinggi badan disimpan ke dalam variabel tinggi.

Program kemudian menghitung **berat badan** berdasarkan rumus:

**berat=bmixtinggixtinggi**

Nilai hasil perhitungan berat badan tersebut disimpan dalam variabel berat.

Selanjutnya, hasil perhitungan dibulatkan ke bilangan bulat terdekat menggunakan fungsi `math.Round()`, kemudian dikonversi menjadi tipe data integer agar dapat ditampilkan tanpa desimal.

Terakhir, hasil berupa berat badan (dalam satuan kilogram) ditampilkan ke layar sebagai output.

### 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)

    d1 := math.Sqrt(math.Pow(x2-x1, 2) + math.Pow(y2-y1, 2))
    d2 := math.Sqrt(math.Pow(x3-x2, 2) + math.Pow(y3-y2, 2))
    d3 := math.Sqrt(math.Pow(x3-x1, 2) + math.Pow(y3-y1, 2))

    longest := math.Max(d1, math.Max(d2, d3))

    fmt.Printf("%.2f\n", longest)
}
```

#### Screenshot program

The screenshot shows a code editor interface with a Go file named `assignment-03.go`. The code implements the formula for calculating the perimeter of a triangle given its vertices' coordinates. The editor has syntax highlighting and a code completion dropdown. Below the editor is a terminal window displaying the command `go run "f:\Me\Others\Kuliah\Algoritma-dan-Pemrograman\week-04"` and the resulting output, which shows the perimeter of the triangle for various input sets.

```
PS F:\Me\Others\Kuliah\Algoritma-dan-Pemrograman\week-04> go run "f:\Me\Others\Kuliah\Algoritma-dan-Pemrograman\week-04"
1.0 1.0
4.0 1.0
1.0 5.0
5.00
PS F:\Me\Others\Kuliah\Algoritma-dan-Pemrograman\week-04> go run "f:\Me\Others\Kuliah\Algoritma-dan-Pemrograman\week-04"
0.0 0.0
3.0 0.0
3.0 4.0
5.00
PS F:\Me\Others\Kuliah\Algoritma-dan-Pemrograman\week-04>
```

### **Deskripsi program**

Program ini meminta pengguna untuk menginput **tiga pasang koordinat titik (x, y)** yang merepresentasikan posisi tiga titik di bidang kartesius.

Masing-masing pasangan nilai disimpan dalam variabel x1, y1, x2, y2, x3, dan y3.

Setelah menerima input, program menghitung **jarak antar setiap pasangan titik** dengan menggunakan **rumus jarak Euclidean**:

$$d = \sqrt{((x_2 - x_1)^2 + (y_2 - y_1)^2)}$$

Perhitungan dilakukan untuk tiga kombinasi titik:

d1 = jarak antara titik 1 dan titik 2

d2 = jarak antara titik 2 dan titik 3

d3 = jarak antara titik 3 dan titik 1

Ketiga hasil jarak tersebut kemudian dibandingkan menggunakan fungsi math.Max() untuk menemukan **jarak terpanjang** di antara ketiganya.

Nilai jarak terpanjang tersebut disimpan dalam variabel longest, kemudian ditampilkan ke layar dengan format dua angka di belakang koma.

.

## TUGAS PENDAHULUAN

Pemberi soal : Abyan Fito Arrasyid

### 1. Tugas Pendahuluan 1

#### Deskipsi soal

Buatlah program yang digunakan untuk menghitung jumlah tahun, bulan, dan hari dari total hari yang dimasukkan oleh pengguna!

**Masukan** berupa satu bilangan bulat positif yang menyatakan total hari.

**Keluaran** menampilkan jumlah tahun, bulan, dan hari yang diperlukan untuk mencapai total hari tersebut (anggap 1 tahun = 365 hari dan 1 bulan = 30 hari).

**Contoh masukan dan keluaran:**

No	Masukan	Keluaran
1	370	1 Tahun, 0 Bulan, 5 Hari
2	800	2 Tahun, 2 Bulan, 10 Hari
3	65	0 Tahun, 2 Bulan, 5 Hari

**Source code**

```
package main

import "fmt"

func main() {
    var total int
    fmt.Scan(&total)

    const (
        tahun = 365
        bulan = 30
    )

    t := total / tahun
    b := (total % tahun) / bulan
    h := (total % tahun) % bulan

    fmt.Println(t, "Tahun,", b, "Bulan,", h, "Hari")
}
```

## Screenshot program

The screenshot shows a code editor with a Go file named `tugas-pendahuluan-01.go`. The code defines a package `main`, imports `fmt`, and calculates the number of years, months, and days from a total number of days. The terminal window below shows the execution of the program and its output.

```
PS F:\Me\Others\Kuliah\Algoritma-dan-Pemrograman\week-04> go run "f:\Me\Others\Kuliah\Algoritma-dan-Pemrograman\week-04\tugas-pendahuluan-01.go"
370
1 Tahun, 0 Bulan, 5 Hari
PS F:\Me\Others\Kuliah\Algoritma-dan-Pemrograman\week-04> go run "f:\Me\Others\Kuliah\Algoritma-dan-Pemrograman\week-04\tugas-pendahuluan-01.go"
800
2 Tahun, 2 Bulan, 10 Hari
PS F:\Me\Others\Kuliah\Algoritma-dan-Pemrograman\week-04> go run "f:\Me\Others\Kuliah\Algoritma-dan-Pemrograman\week-04\tugas-pendahuluan-01.go"
65
0 Tahun, 2 Bulan, 5 Hari
```

Output window content:

Nama : Dadi Maulana Muhammad
NIM : 109082500080
Kelas : S1IF-13-04

Terminal status bar: Ln 20, Col 1 Tab Size: 2 Ln 3, Col 19 | 69 character Plain t 130% Wind UTF-8

## Deskripsi program

Program ini meminta pengguna untuk menginput satu bilangan bulat yang merepresentasikan jumlah total hari. Setelah pengguna memasukkan nilai tersebut, program menyimpannya ke dalam variabel `total`.

Selanjutnya, program mengonversi nilai total hari tersebut menjadi bentuk tahun, bulan, dan hari, dengan ketentuan:

1 tahun = 365 hari

1 bulan = 30 hari

Proses konversi dilakukan dengan:

Menghitung jumlah tahun (`t`) dengan membagi total hari dengan 365.

Menghitung jumlah bulan (`b`) dari sisa hari setelah dikonversi ke tahun, kemudian dibagi 30.

Menghitung jumlah hari (`h`) dari sisa pembagian bulan.

Setelah proses konversi selesai, hasil berupa jumlah tahun, bulan, dan hari ditampilkan ke layar sebagai output.

## 2. Tugas Pendahuluan 2

### Deskirpsi soal

Buatlah program yang digunakan untuk mengubah total detik menjadi format waktu lengkap, yaitu hari, jam, menit, dan detik.

**Masukan** berupa satu bilangan bulat positif yang menyatakan total detik.

**Keluaran** menampilkan waktu dalam bentuk hari, jam, menit, dan detik (anggap 1 hari = 24 jam, 1 jam = 60 menit, dan 1 menit = 60 detik).

### Contoh masukan dan keluaran:

No	Masukan	Keluaran
1	90061	1 hari, 1 jam, 1 menit, 1 detik
2	172800	2 hari, 0 jam, 0 menit, 0 detik
3	29110	0 hari, 8 jam, 5 menit, 10 detik

### Source code

```
package main

import (
    "fmt"
)

func main() {
    var totalDetik int
    fmt.Scan(&totalDetik)

    hari := totalDetik / 86400
    jam := (totalDetik % 86400) / 3600
    menit := (totalDetik % 3600) / 60
    detik := totalDetik % 60

    fmt.Printf("%d hari, %d jam, %d menit, %d detik\n", hari,
    jam, menit, detik)
}
```

## Screenshoot program

The screenshot shows a code editor window with a Go file named `tugas-pendahuluan-02.go`. The code defines a package `main` that reads a total number of seconds from the user, converts it into days, hours, minutes, and seconds, and prints the result. Below the code editor is a terminal window showing the execution of the program with input values of 90061, 172800, and 29110, and their corresponding outputs in days, hours, minutes, and seconds.

```
-eo tugas-pendahuluan-02.go X
-eo tugas-pendahuluan-02.go > main
1 package main
2
3 import (
4     "fmt"
5 )
6
7 func main() {
8     var totalDetik int
9     fmt.Scan(&totalDetik)
10
11    hari := totalDetik / 86400
12    jam := (totalDetik % 86400) / 3600
13    menit := (totalDetik % 3600) / 60
14    detik := totalDetik % 60
15
16    fmt.Printf("%d hari, %d jam, %d menit, %d detik\n", hari, jam, menit, detik)
17 }
18
```

PROBLEMS 8 OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
.go"
90061
1 hari, 1 jam, 1 menit, 1 detik
PS F:\Me\Others\Kuliah\Algoritma-dan-Pemrograman\week-04> go run "f:\Me\Others\Kuliah\Algori
.go"
172800
2 hari, 0 jam, 0 menit, 0 detik
PS F:\Me\Others\Kuliah\Algoritma-dan-Pemrograman\week-04> go run "f:\Me\Others\Kuliah\Algori
.go"
29110
0 hari, 8 jam, 5 menit, 10 detik
PS F:\Me\Others\Kuliah\Algoritma-dan-Pemrograman\week-04>
```

Nama : Dadi Maulana Muhammad  
NIM : 109082500080  
Kelas : S1IF-13-04

Ln 10, Col 1 Tab Size 2 Ln 3, Col 19 69 character Plain text 130% Wind UTF-8

## Deskripsi program

Program ini meminta pengguna untuk menginput satu bilangan bulat yang merepresentasikan jumlah total detik. Setelah pengguna memasukkan nilai tersebut, program menyimpannya ke dalam variabel `totalDetik`.

Nilai `totalDetik` kemudian dikonversi menjadi satuan waktu yang lebih mudah dipahami, yaitu **hari**, **jam**, **menit**, dan **detik**. Proses konversi dilakukan dengan menggunakan operasi pembagian dan sisa bagi (/ dan %), dengan ketentuan:

1 hari = 86.400 detik

1 jam = 3.600 detik

1 menit = 60 detik

Langkah-langkah konversi:

Menghitung jumlah **hari** dengan membagi total detik dengan 86.400.

Menghitung jumlah **jam** dari sisa pembagian hari dengan membagi hasilnya dengan 3.600.

Menghitung jumlah **menit** dari sisa pembagian jam dengan membagi hasilnya dengan 60.

Menghitung sisa pembagian terakhir sebagai jumlah **detik**.

Setelah proses konversi selesai, hasil berupa jumlah hari, jam, menit, dan detik ditampilkan ke layar sebagai output.

.