

LAPORAN PRAKTIKUM
Algoritma Pemrograman

MODUL 3
Operasi Aritmetika & Struktur Dasar Program



Disusun oleh:

Nama:Nafisah Salsabila

NIM:109082500063

S1IF-13-04

PROGRAM STUDI S1 INFORMATIKA
FAKULTAS INFORMATIKA
TELKOM UNIVERSITY PURWOKERTO

2025

LATIHAN KELAS – GUIDED

1. Guided 1

Source Code

```
package main

import "fmt"

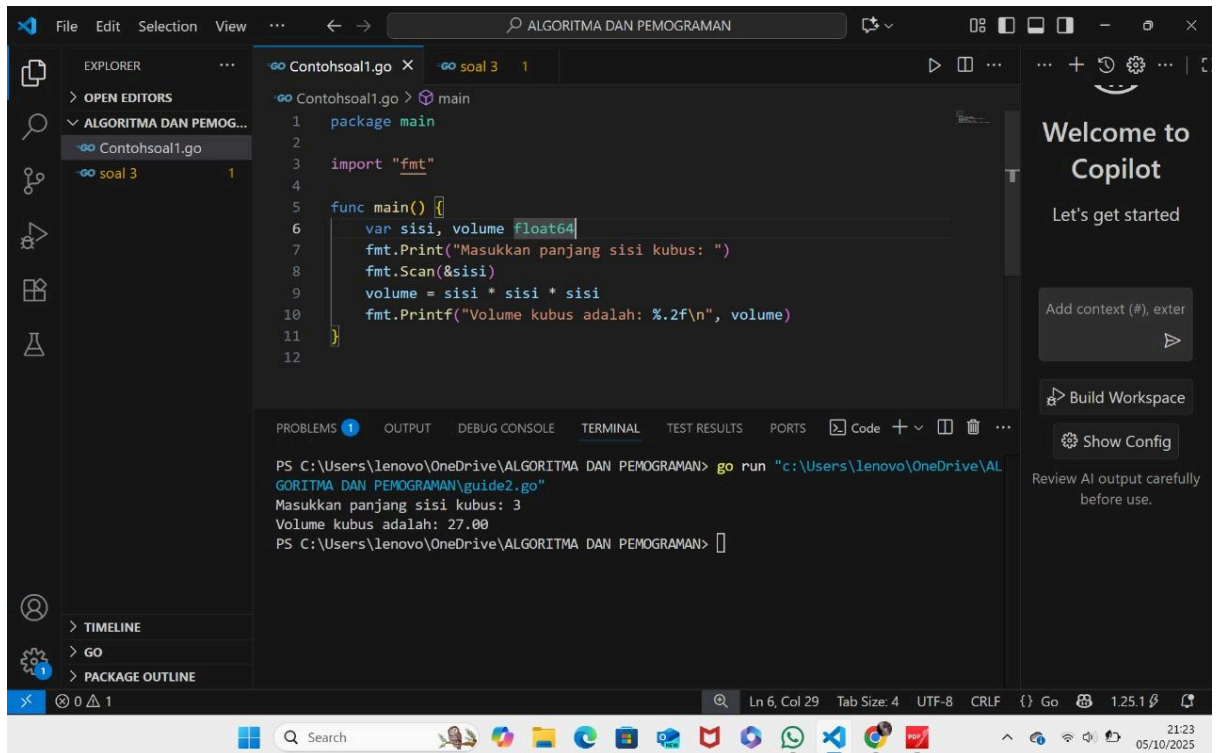
func main() {
    var sisi, volume float64
    fmt.Print("Masukkan panjang sisi kubus: ")
    fmt.Scan(&sisi)
    volume = sisi * sisi * sisi
    fmt.Printf("Volume kubus adalah: %.2f\n",
volume)
}
```

Output:

Masukkan panjang sisi kubus: 3

Volume kubus adalah: 27.00

Screenshoot program



Deskripsi program

Program ini dibuat untuk menghitung volume suatu kubus berdasarkan panjang sisi yang dimasukkan oleh pengguna.

Ketika dijalankan, program meminta pengguna untuk memasukkan nilai sisi kubus melalui keyboard. Nilai sisi tersebut kemudian digunakan untuk menghitung volume dengan rumus $\text{volume} = \text{sisi} \times \text{sisi} \times \text{sisi}$.

Setelah proses perhitungan selesai, program akan menampilkan hasil volume kubus ke layar dalam format dua angka di belakang koma. Program ini menggunakan tipe data float64 agar bisa menerima angka desimal.

2. Guided 2

Source Code

```
package main

import "fmt"

func main() {
    var alas, tinggi, luas float64
    fmt.Print("Masukkan alas segitiga: ")
    fmt.Scan(&alas)
    fmt.Print("Masukkan tinggi segitiga: ")
}
```

```
fmt.Scan(&tinggi)

luas = 0.5 * alas * tinggi

fmt.Printf("Luas segitiga adalah: %.2f\n", luas)

}
```

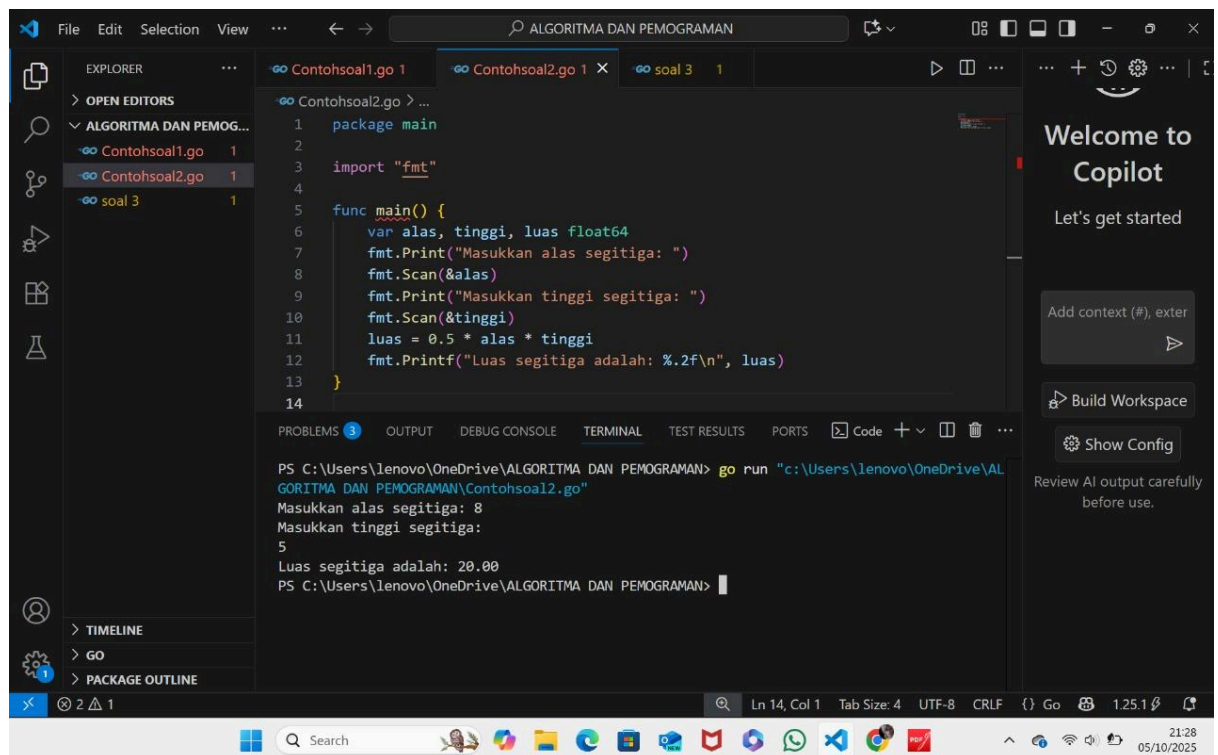
Output:

Masukkan alas segitiga: 8

Masukkan tinggi segitiga:5

Luas segitiga adalah: 20.00

Screenshoot program



Deskripsi program

Program ini berfungsi untuk menghitung luas sebuah segitiga berdasarkan nilai alas dan tinggi yang dimasukkan oleh pengguna.

Saat program dijalankan, pengguna diminta untuk menginput dua nilai, yaitu panjang alas dan tinggi segitiga. Program kemudian menghitung luas dengan menggunakan rumus $luas = 0.5 \times alas \times tinggi$.

Hasil perhitungan luas ditampilkan di layar dalam format dua angka di belakang koma. Program ini sederhana namun efektif untuk memahami penggunaan operator aritmetika dan input-output pada bahasa Go.

3. Guided 3

Source Code

```
package main

import "fmt"

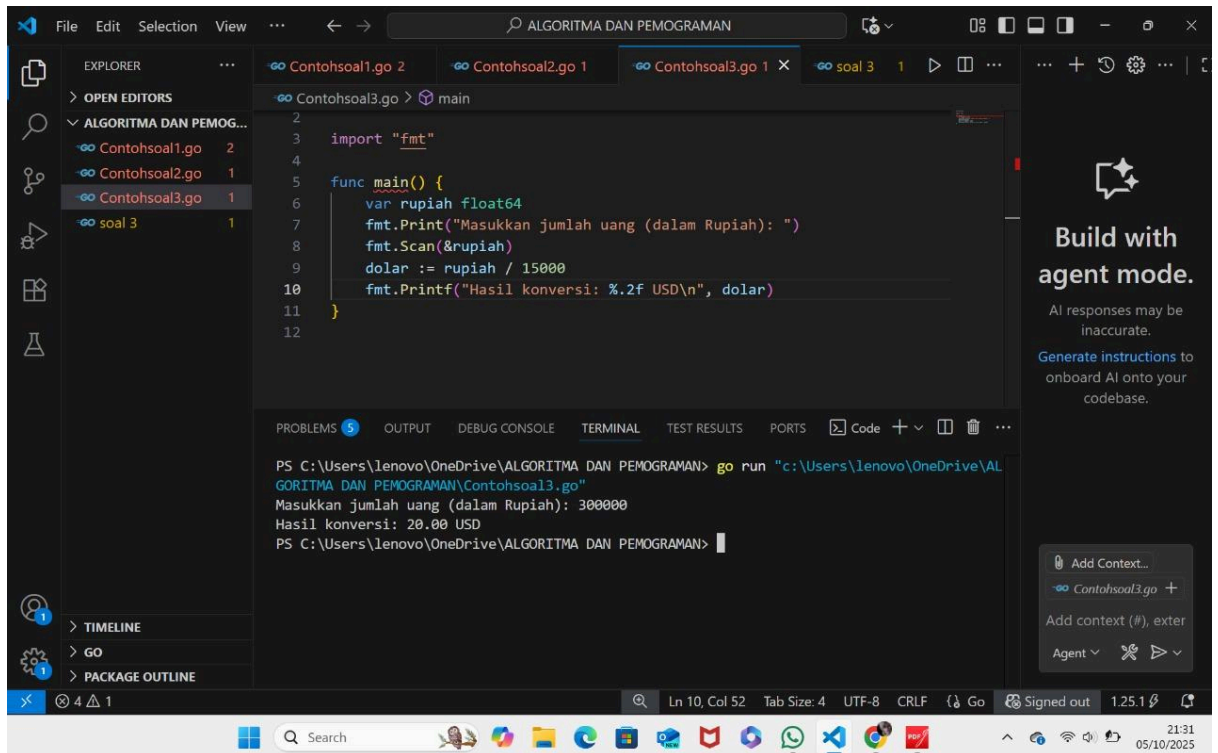
func main() {
    var rupiah float64
    fmt.Print("Masukkan jumlah uang (dalam Rupiah):")
    fmt.Scan(&rupiah)
    dolar := rupiah / 15000
    fmt.Printf("Hasil konversi: %.2f USD\n", dolar)
}
```

Output:

Masukkan jumlah uang (dalam Rupiah): 300000

Hasil konversi: 20.00 USD

Screenshoot program



Deskripsi program

Program ini digunakan untuk mengonversi nilai uang dari Rupiah (IDR) ke Dolar Amerika Serikat (USD) berdasarkan kurs yang telah ditentukan.

Saat dijalankan, program meminta pengguna untuk memasukkan jumlah uang dalam Rupiah. Nilai tersebut akan dikonversi ke Dolar dengan rumus $USD = \text{Rupiah} / 15000$, di mana 1 USD dianggap setara dengan 15.000 Rupiah.

Setelah melakukan perhitungan, program menampilkan hasil konversi ke layar dengan dua angka di belakang koma. Program ini membantu memahami operasi pembagian dan format tampilan hasil pada bahasa Go.

TUGAS

1. Tugas 1

Source code

```
package main

import "fmt"

func main() {
    var x, fx float64

    fmt.Print("Masukkan nilai x: ")
}
```

```

    fmt.Scan(&x)

    fx = 2/(x+5) + 5

    fmt.Println("Nilai f(x):", fx)
}

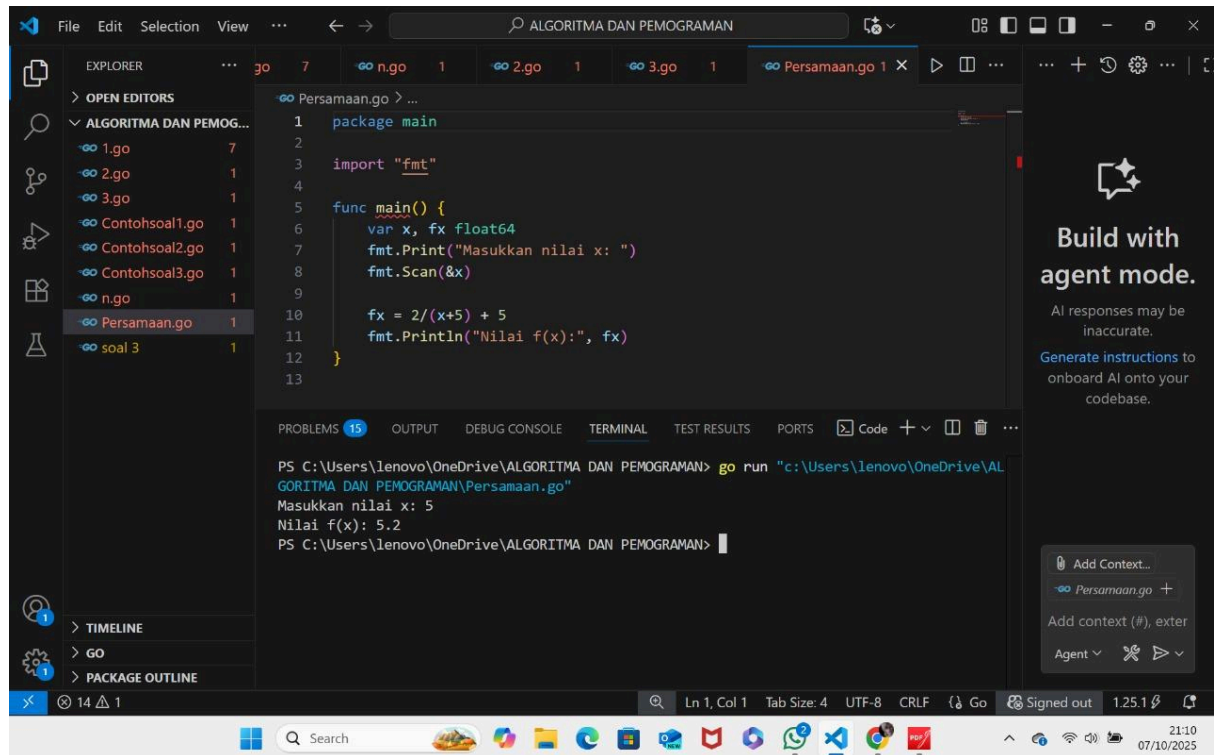
```

Output:

Masukkan nilai x: 5

Nilai f(x): 5.2

Screenshoot program



Deskripsi program

Program ini digunakan untuk menghitung nilai fungsi matematika $f(x) = 2 / (x + 5) + 5$ dengan input berupa nilai x dari pengguna.

Program memanggil package fmt untuk menangani proses input dan output. Variabel x dan fx dideklarasikan dengan tipe data float64, karena fungsi ini melibatkan operasi pembagian yang menghasilkan bilangan desimal.

Program meminta pengguna untuk memasukkan nilai x melalui perintah `fmt.Scan(&x)`. Setelah itu, nilai $f(x)$ dihitung menggunakan rumus $2 / (x + 5) + 5$ dan disimpan ke dalam variabel `fx`.

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

2. Tugas 2

Source code

```
package main

import (
    "fmt"
    "math"
)

func main() {
    var r, volume, luas float64

    fmt.Print("Masukkan jari-jari bola: ")
    fmt.Scan(&r)

    volume = (4.0 / 3.0) * math.Pi * math.Pow(r, 3)
    luas = 4 * math.Pi * math.Pow(r, 2)

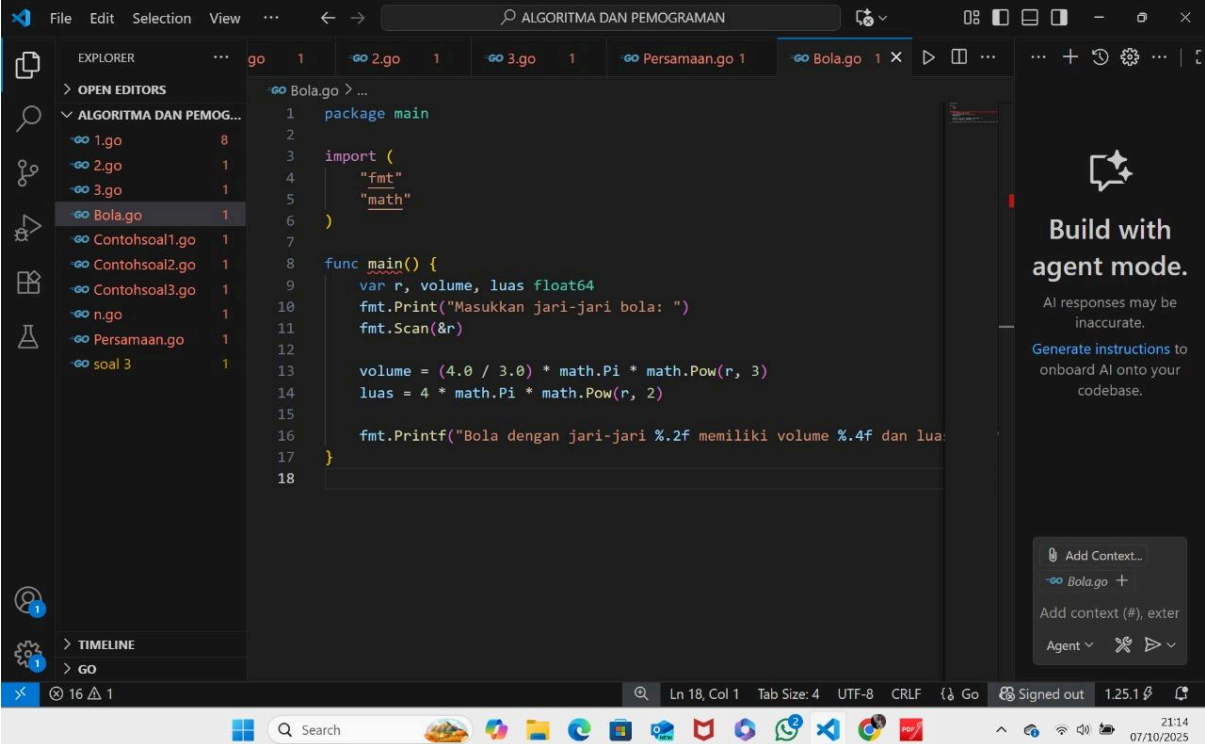
    fmt.Printf("Bola dengan jari-jari %.2f memiliki volume %.4f dan luas %.4f\n", r, volume, luas)
}
```

Output:

Masukkan jari-jari bola: 5

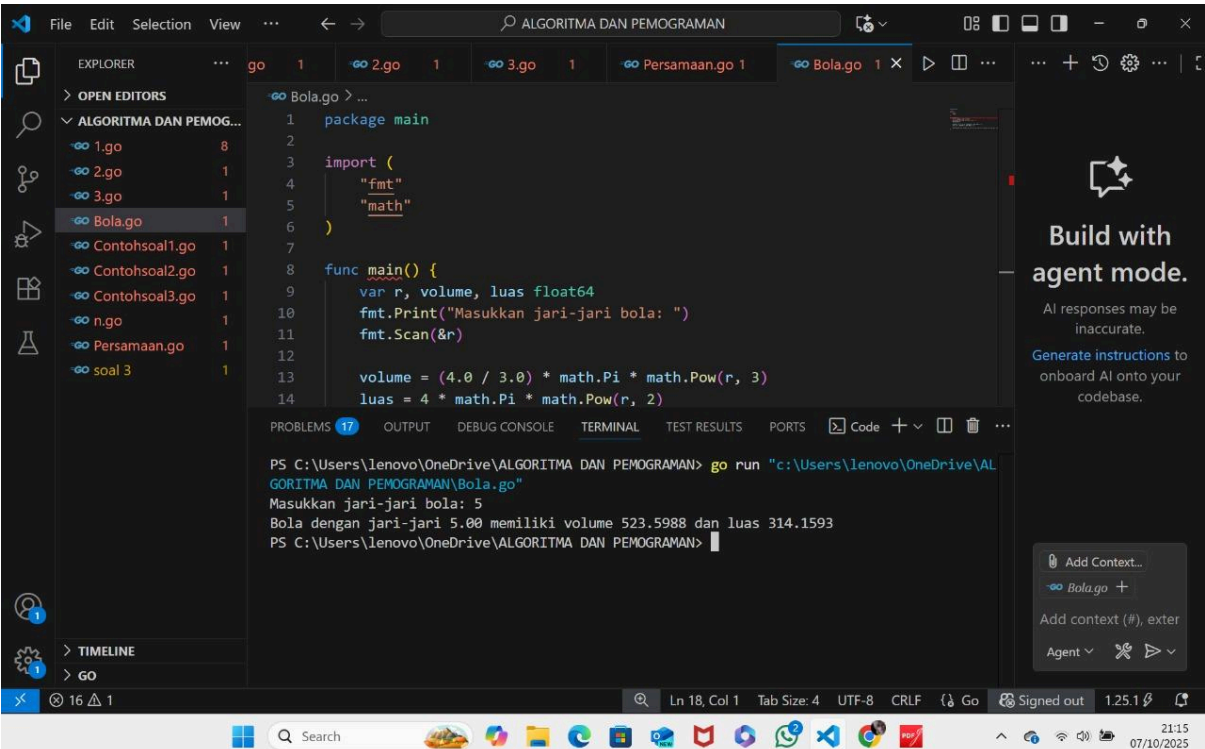
Bola dengan jari-jari 5 memiliki volume 523.5988 dan luas 314.1593

Screenshoot program



The screenshot shows the Visual Studio Code editor with a Go file named `Bola.go` open. The code calculates the volume and surface area of a sphere based on a user input radius. The Explorer sidebar on the left shows a project named "ALGORITMA DAN PEMOG..." with several files including `1.go`, `2.go`, `3.go`, `Bola.go`, `Contohsoal1.go`, `Contohsoal2.go`, `Contohsoal3.go`, `n.go`, `Persamaan.go`, and `soal 3`. The right sidebar contains a "Build with agent mode." panel with a warning that AI responses may be inaccurate and a button to "Generate instructions to onboard AI onto your codebase."

```
1 package main
2
3 import (
4     "fmt"
5     "math"
6 )
7
8 func main() {
9     var r, volume, luas float64
10    fmt.Print("Masukkan jari-jari bola: ")
11    fmt.Scan(&r)
12
13    volume = (4.0 / 3.0) * math.Pi * math.Pow(r, 3)
14    luas = 4 * math.Pi * math.Pow(r, 2)
15
16    fmt.Printf("Bola dengan jari-jari %.2f memiliki volume %.4f dan lua.
17
18
```



This screenshot shows the same VS Code editor with the `Bola.go` file, but now the "TERMINAL" tab is active at the bottom. It displays the command `go run "c:\Users\lenovo\OneDrive\ALGORITMA DAN PEMOGRAMAN\Bola.go"` and its output. The program prompts for the radius, which is entered as 5. The output shows the calculated volume and surface area.

```
PS C:\Users\lenovo\OneDrive\ALGORITMA DAN PEMOGRAMAN> go run "c:\Users\lenovo\OneDrive\ALGORITMA DAN PEMOGRAMAN\Bola.go"
Masukkan jari-jari bola: 5
Bola dengan jari-jari 5.00 memiliki volume 523.5988 dan luas 314.1593
PS C:\Users\lenovo\OneDrive\ALGORITMA DAN PEMOGRAMAN>
```

Deskripsi program

Program ini digunakan untuk menghitung volume dan luas permukaan bola berdasarkan nilai jari-jari yang dimasukkan oleh pengguna.

Program ditulis menggunakan bahasa pemrograman Go dan memanfaatkan package `fmt` untuk proses input serta output, dan `math` untuk menggunakan konstanta π (pi) dan fungsi perpangkatan.

Setelah pengguna memasukkan jari-jari bola, program akan menghitung volume dan luas permukaannya menggunakan rumus matematika bola, yaitu $V = \frac{4}{3} \times \pi \times r^3$ untuk volume dan $L = 4 \times \pi \times r^2$ untuk luas permukaan.

Hasil perhitungan kemudian ditampilkan ke layar dalam format angka desimal. Program ini membantu pengguna memperoleh hasil perhitungan bola dengan cepat dan akurat.

3. Tugas 3

Source code

```
package main
import "fmt"

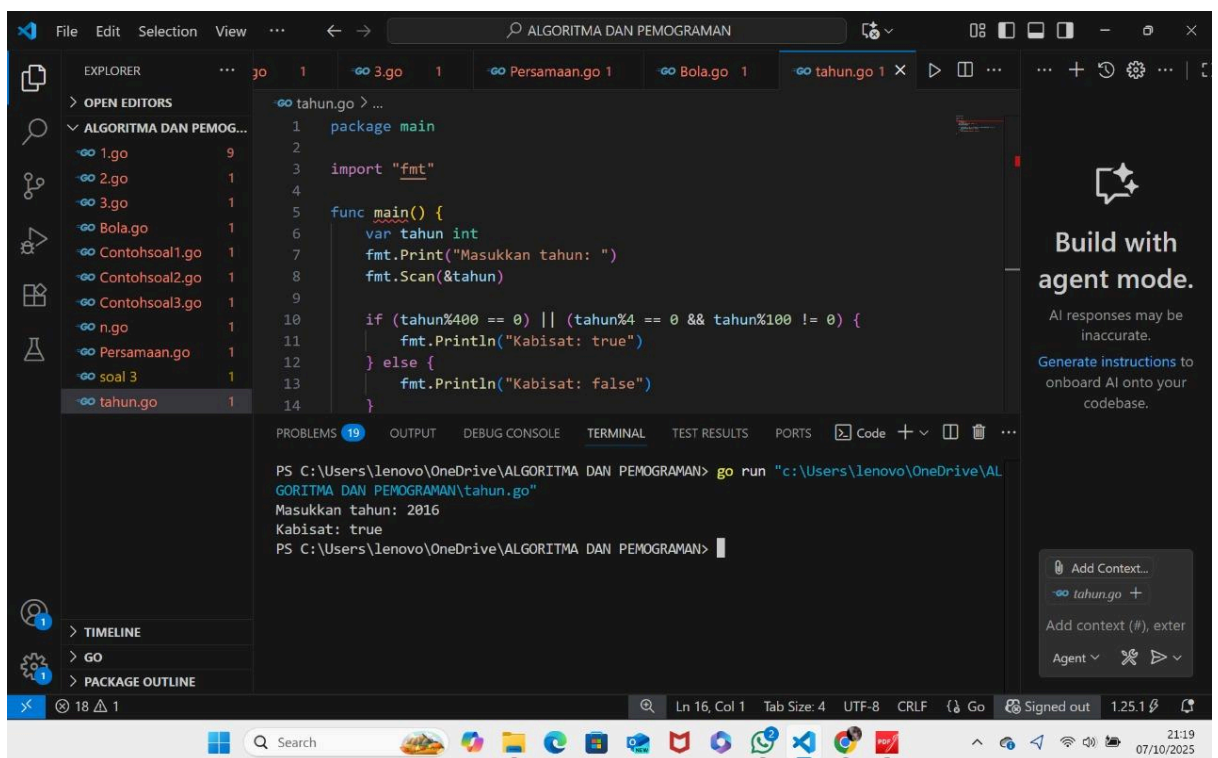
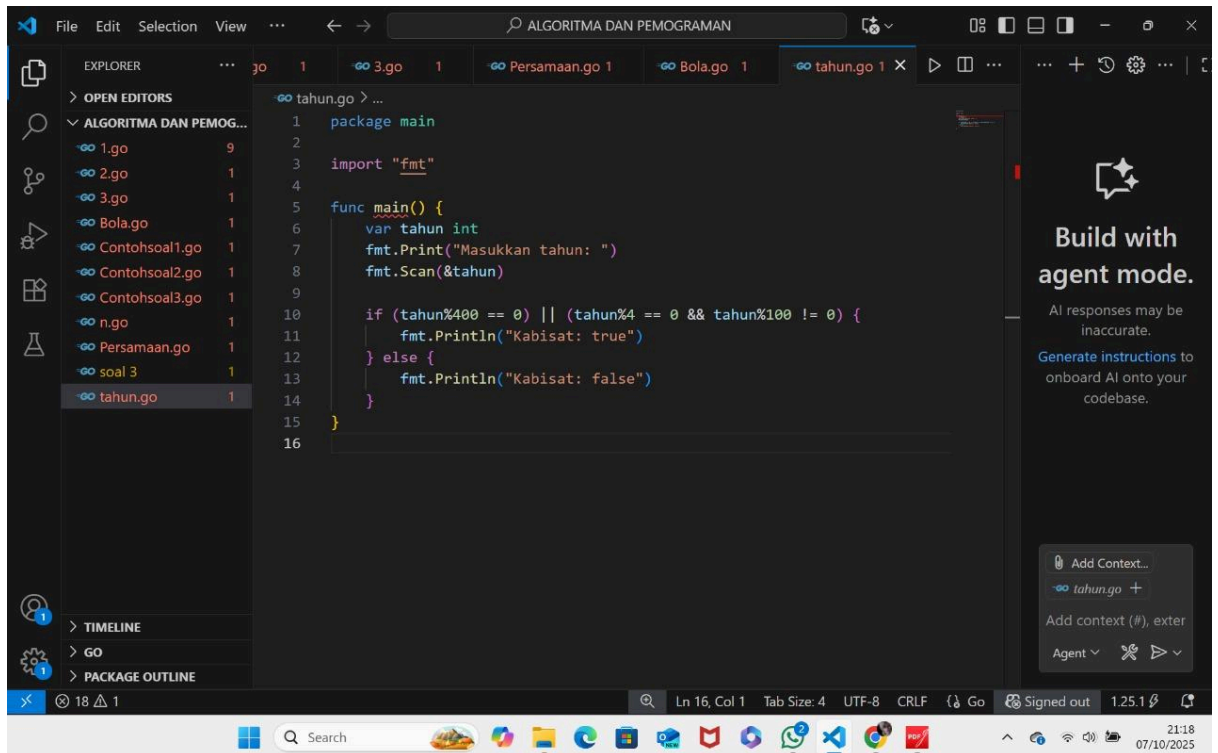
func main() {
    var tahun int
    fmt.Print("Masukkan tahun: ")
    fmt.Scan(&tahun)

    if (tahun%400 == 0) || (tahun%4 == 0 && tahun%100 != 0) {
        fmt.Println("Kabisat: true")
    } else {
        fmt.Println("Kabisat: false")
    }
}
```

Output:

```
Masukkan tahun: 2016
Kabisat: true
```

Screenshoot program



Deskripsi program

Program ini digunakan untuk menentukan apakah suatu tahun merupakan tahun kabisat atau bukan. Program dibuat menggunakan bahasa pemrograman Go dengan memanfaatkan struktur percabangan if-else.

Pengguna diminta untuk memasukkan sebuah nilai tahun. Program kemudian memeriksa kondisi berdasarkan aturan tahun kabisat, yaitu: tahun yang habis dibagi 400, atau tahun yang habis dibagi 4 tetapi tidak habis dibagi 100.

Jika salah satu kondisi tersebut terpenuhi, program menampilkan hasil "Kabisat: true", sedangkan jika tidak terpenuhi, program menampilkan "Kabisat: false". Program ini membantu pengguna mengecek status tahun kabisat secara otomatis dan cepat.

4. Tugas 4

Source code

```
package main
import "fmt"

func main() {
    var celsius, fahrenheit, reamur, kelvin float64
    fmt.Println("Masukkan suhu Celsius: ")
    fmt.Scan(&celsius)

    fahrenheit = (celsius * 9 / 5) + 32
    reamur = celsius * 4 / 5
    kelvin = celsius + 273.15

    fmt.Println("Derajat Reamur:", reamur)
    fmt.Println("Derajat Fahrenheit:", fahrenheit)
    fmt.Println("Derajat Kelvin:", kelvin)
}
```

Output:

```
Masukkan suhu Celsius: 50
Derajat Reamur: 40
Derajat Fahrenheit: 122
Derajat Kelvin: 323.15
```

Screenshot program

The screenshot shows the Visual Studio Code editor with the file 'suhu.go' open. The code is as follows:

```
1 package main
2
3 import "fmt"
4
5 func main() {
6     var celsius, fahrenheit, reamur, kelvin float64
7     fmt.Print("Masukkan suhu Celsius: ")
8     fmt.Scan(&celsius)
9
10    fahrenheit = (celsius * 9 / 5) + 32
11    reamur = celsius * 4 / 5
12    kelvin = celsius + 273.15
13
14    fmt.Println("Derajat Reamur:", reamur)
15    fmt.Println("Derajat Fahrenheit:", fahrenheit)
16    fmt.Println("Derajat Kelvin:", kelvin)
17 }
18
```

The Explorer sidebar on the left shows a project named 'ALGORITMA DAN PEMOGRAMAN' with several files, including 'suhu.go' which is currently selected. The status bar at the bottom indicates 'Ln 8, Col 23', 'Tab Size: 4', 'UTF-8', 'CRLF', and 'Signed out'.

This screenshot shows the same VS Code editor with the 'suhu.go' file, but now the 'TERMINAL' tab is active, displaying the program's output after execution:

```
PS C:\Users\lenovo\OneDrive\ALGORITMA DAN PEMOGRAMAN> go run "c:\Users\lenovo\OneDrive\ALGORITMA DAN PEMOGRAMAN\suhu.go"
Masukkan suhu Celsius: 50
Derajat Reamur: 40
Derajat Fahrenheit: 122
Derajat Kelvin: 323.15
PS C:\Users\lenovo\OneDrive\ALGORITMA DAN PEMOGRAMAN>
```

The Explorer sidebar remains on the left, and the status bar at the bottom shows 'Ln 8, Col 23', 'Tab Size: 4', 'UTF-8', 'CRLF', and 'Signed out'.

Deskripsi program

Program ini digunakan untuk mengonversi suhu dari satuan Celsius ke tiga satuan suhu lainnya, yaitu Reamur, Fahrenheit, dan Kelvin. Program dibuat menggunakan bahasa pemrograman Go dan memanfaatkan operasi aritmetika sederhana untuk melakukan perhitungan.

Pengguna diminta untuk memasukkan nilai suhu dalam Celsius. Setelah itu, program secara otomatis menghitung nilai suhu dalam tiga satuan lain dengan menggunakan rumus konversi, yaitu: Reamur = $(4/5) \times \text{Celsius}$, Fahrenheit = $(9/5 \times \text{Celsius}) + 32$, dan Kelvin = $\text{Celsius} + 273.15$.

Hasil konversi ditampilkan di layar dengan format yang mudah dibaca. Program ini membantu pengguna melakukan perbandingan suhu antar satuan dengan cepat dan akurat.

Tugas Pendahuluan

Soal:

1. Jelaskan perbedaan `fmt.Print()`, `fmt.Println()`, dan `fmt.Printf()` di Go!
2. Jelaskan penggunaan tipe data `int`, `float64`, `bool`, dan `string` di Go!
3. Bagaimana Cara mendeklarasikan variabel dengan kata kunci `var` dan dengan acara singkat? Berikan contoh! Hint `":="`
4. Apa perbedaan antara operator `==` dan `=` dalam bahasa Go?
5. Buatlah program Go sederhana untuk meminta input nama kalian, lalu menampilkan nama.

Jawaban:

1.
 - `fmt.Print()` adalah untuk mencetak teks ke layar tanpa menambah baris baru.

→ Contoh:

Go:

```
fmt.Print("Halo ")  
fmt.Print("Billa")
```

Output:

Halo Billa

- `fmt.Println()` → mencetak teks lalu otomatis pindah baris setelahnya.

→ Contoh:

Go:

```
fmt.Println("Halo")  
fmt.Println("Billa")
```

Output:

Halo
Billa

- `fmt.Println()` adalah mencetak teks lalu otomatis pindah baris setelahnya.

→ Contoh:

Go:

```
fmt.Printf("Nilai saya: %.2f", 90.555)
```

Output:

Nilai saya: 90.56

2. `int` → untuk bilangan bulat, misalnya 10, -7, 0.

float64 → untuk bilangan desimal, misalnya 3.14, 5.0.

bool → untuk nilai benar/salah, yaitu true atau false.

string → untuk teks, misalnya "Halo Dunia".

Contoh:

```
var umur int = 18
var tinggi float64 = 174.5
var sehat bool = true
var nama string = "Nana"
```

3. Dengan var:

```
var nama string = "Billa"
var umur int = 18
```

Dengan cara singkat (:=):

```
nama := "Billa"
umur := 18
```

Ini otomatis menentukan tipe data berdasarkan nilainya (string, int, dll).

4. = untuk memberikan nilai (assignment) ke variabel.

Contoh: x = 5

== untuk membandingkan dua nilai.

Contoh: x == 5 (hasilnya true kalau x memang 5)

5. Program sederhana untuk memasukkan nama:

```
package main
import "fmt"

func main() {
    var nama string
    fmt.Print("Masukkan nama kamu: ")
    fmt.Scan(&nama)
    fmt.Println("Halo,", nama, "selamat datang di Golang!")
}
```

Output:

Masukkan nama kamu: Billa

Halo, Billa selamat datang di Golang