

**LAPORAN PRAKTIKUM  
ALGORITMA DAN PEMROGRAMAN 2**

**MODUL II  
REVIEW STRUKTUR KONTROL**



**Disusun Oleh :**

**ALTHAF TEGAR SOFYAN / 2311102217**

**Kelas IF 11 06**

**Dosen Pengampu :**

**Abednego Dwi Septiadi, S.Kom., M.Kom.**

**PROGRAM STUDI S1 TEKNIK INFORMATIKA**

**FAKULTAS INFORMATIKA**

**TELKOM UNIVERSITY PURWOKERTO**

**2024**

## I. DASAR TEORI

Golang, atau Go, adalah bahasa pemrograman yang dikembangkan oleh Google. Dikenalkan pada tahun 2009, Go dirancang untuk kebutuhan sistem yang efisien dan memiliki kinerja tinggi, dengan fokus pada kesederhanaan dan produktivitas pengembang. Go merupakan bahasa kompilasi, yang berarti kode sumbernya akan diterjemahkan menjadi kode mesin sebelum dijalankan.

### Karakteristik Utama Golang

- **Statically Typed:** Go adalah bahasa yang memiliki tipe data statis, artinya jenis variabel ditentukan saat kompilasi.
- **Concurrency:** Go memiliki dukungan bawaan untuk pemrograman bersamaan (concurrent programming) melalui goroutines dan channel, yang memungkinkan eksekusi fungsi secara paralel.
- **Garbage Collection:** Go secara otomatis mengelola memori, menghapus objek yang tidak lagi digunakan untuk mencegah kebocoran memori.
- **Sederhana dan Mudah Dibaca:** Sintaks Go dirancang agar mudah dipahami, membuatnya cocok untuk pengembang pemula dan profesional.

### Tipe Data

Go mendukung berbagai tipe data, antara lain:

- **Integer:** int, int8, int16, int32, int64
- **Floating Point:** float32, float64
- **Boolean:** bool
- **String:** string
- **Array dan Slice:** Struktur data untuk menyimpan koleksi elemen.
- **Map:** Struktur data yang menyimpan pasangan kunci-nilai.

Fungsi dalam Go didefinisikan dengan kata kunci func. Fungsi dapat memiliki parameter dan mengembalikan nilai.

## II. GUIDED

1. Telusuri program berikut dengan cara mengkompilasi dan mengeksekusi program. Silahkan masukkan data yang sesuai sebanyak yang diminta program. Perhatikan keluaran yang diperoleh.

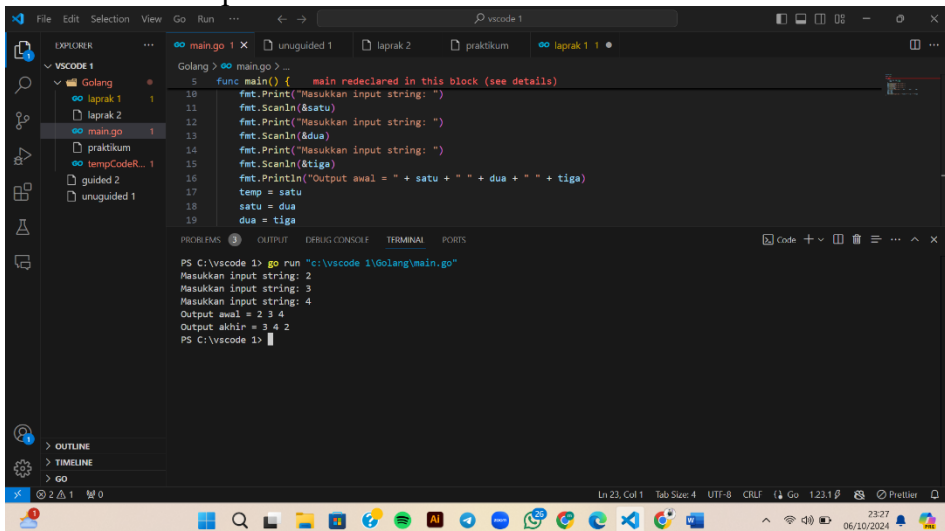
### Source Code

```
package main

import "fmt"

func main(){
    var(
        satu, dua, tiga string
        temp string
    )
    fmt.Print("Masukkan input string: ")
    fmt.Scanln(&satu)
    fmt.Print("Masukkan input string: ")
    fmt.Scanln(&dua)
    fmt.Print("Masukkan input string: ")
    fmt.Scanln(&tiga)
    fmt.Println("Output awal = " + satu + " " + dua + " " +
tiga)
    temp = satu
    satu = dua
    dua = tiga
    tiga = temp
    fmt.Println("Output akhir = " + satu + " " + dua + " " +
tiga)
}
```

### Screenshoot Output

The screenshot shows the Visual Studio Code interface. The Explorer sidebar on the left shows a project named 'Golang' with files 'laprak 1', 'laprak 2', 'praktikum', 'tempCodeR...', 'guided 2', and 'unguided 1'. The main editor displays a Go program with line numbers 5 to 19. The program prompts for three input strings and prints the initial concatenated output, then performs a swap using a temporary variable and prints the final concatenated output. The Output panel at the bottom shows the execution results: 'PS C:\vscode 1> go run "c:\vscode 1\Golang\main.go"', followed by the three input prompts and the two output lines: 'Output awal = 2 3 4' and 'Output akhir = 3 4 2'. The status bar at the bottom indicates 'Ln 23, Col 1', 'Tab Size 4', 'UTF-8', 'CRLF', and '123.1 B'.

Deskripsi Program: Program mendeklarasi tiga variable string, pengguna diminta untuk memasukkan tiga input string. Program akan mencetak output awal yang menggabungkan ketiga string. Kemudian menggunakan variable sementara melakukan penukaran nilai string. Program mencetak output akhir yang menunjukkan ketiga string setelah proses pertukaran.

2. Tahun kabisat adalah tahun yang habis dibagi 400 atau habis dibagi 4 tetapi tidak habis dibagi 100. Buatlah sebuah program yang menerima input sebuah bilangan dan memeriksa apakah bilangan tersebut merupakan tahun kabisat atau bukan

#### Source Code

```
package main

import "fmt"

func isKabisat(year int) bool {
    if year%400 == 0 {
        return true
    }
    if year%4 == 0 && year%100 != 0 {
        return true
    }

    return false
}

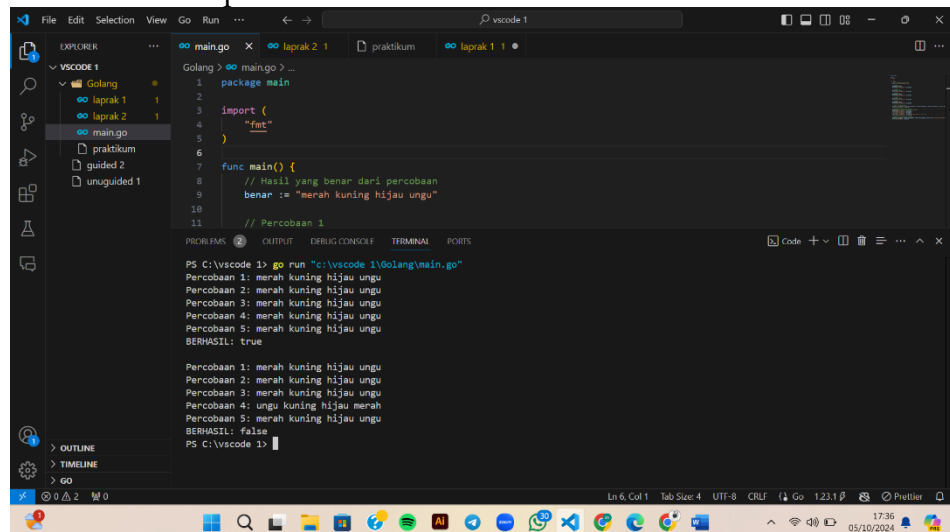
func main() {
    var tahun int

    fmt.Print("Tahun: ")
    fmt.Scanln(&tahun)

    kabisat := isKabisat(tahun)

    fmt.Printf("Kabisat: %t\n", kabisat)
}
```

#### Screenshoot Output



Deskripsi Program: Program ini menggunakan fungsi is kabisat yang dimana fungsi ini menerima parameter year dan mengembalikan nilai boolean. Program meminta pengguna untuk memasukkan tahun yang ingin

diperiksa. Program memanggil fungsi isKabisat dengan tahun yang di inputkan untuk menentukan apakah tahun tersebut kabisat atau bukan. Program mencetak hasil pemeriksaan.

3. Buat program bola yang menerima input jari jari suatu bola. Tampilkan volume dan luas kulit bola dan luas bola

#### Sourcecode

```
package main

import (
    "fmt"
    "math"
)

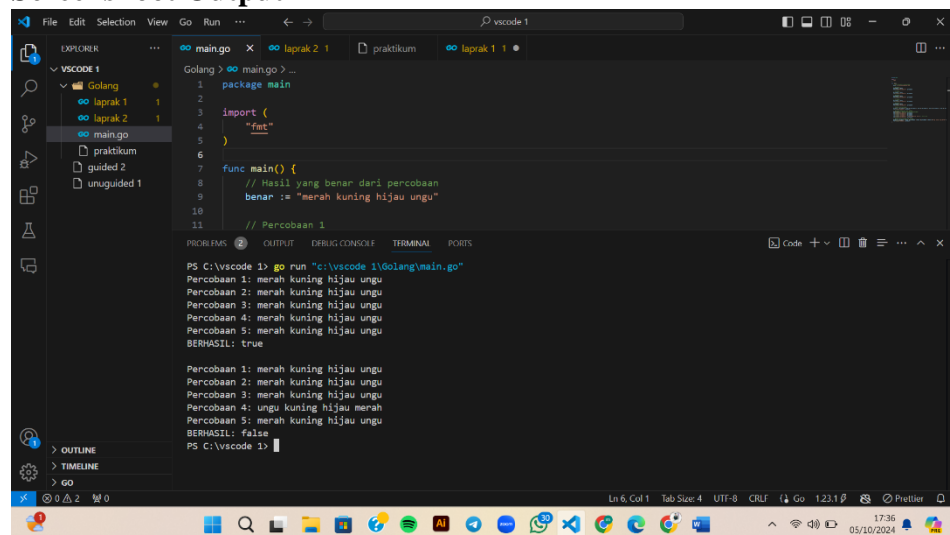
func main() {
    var jejariBola float64
    fmt.Print("Jari-Jari Bola: ")
    fmt.Scanln(&jejariBola)

    volumeBola := (4.0/3.0) * math.Pi * math.Pow(jejariBola,
3)

    luasBola := 4 * math.Pi * math.Pow(jejariBola, 2)

    fmt.Printf("Bola dengan jejari %.0f memiliki volume %.4f
dan luas kulit %.4f\n", jejariBola, volumeBola, luasBola)
}
```

#### Screenshoot Output



**Deskripsi Program:** Program tersebut untuk menghitung volume dan luas permukaan bola berdasarkan jari jari yang di inputkan. Program akan mencetak hasil volume dan luas permukaan bola dengan menampilkan jari jari, volume, dan luas dengan presisi empat angka desimal

### III. UNGUIDED

#### 1. Soal Latihan modul 2B. NO 1

##### Source Code

```
package main

import (
    "fmt"
)

func main() {
    // Hasil yang benar dari percobaan
    benar := "merah kuning hijau ungu"

    // Percobaan 1
    percobaan1 := benar
    fmt.Println("Percobaan 1:", percobaan1)

    // Percobaan 2
    percobaan2 := benar
    fmt.Println("Percobaan 2:", percobaan2)

    // Percobaan 3
    percobaan3 := benar
    fmt.Println("Percobaan 3:", percobaan3)

    // Percobaan 4
    percobaan4 := benar
    fmt.Println("Percobaan 4:", percobaan4)

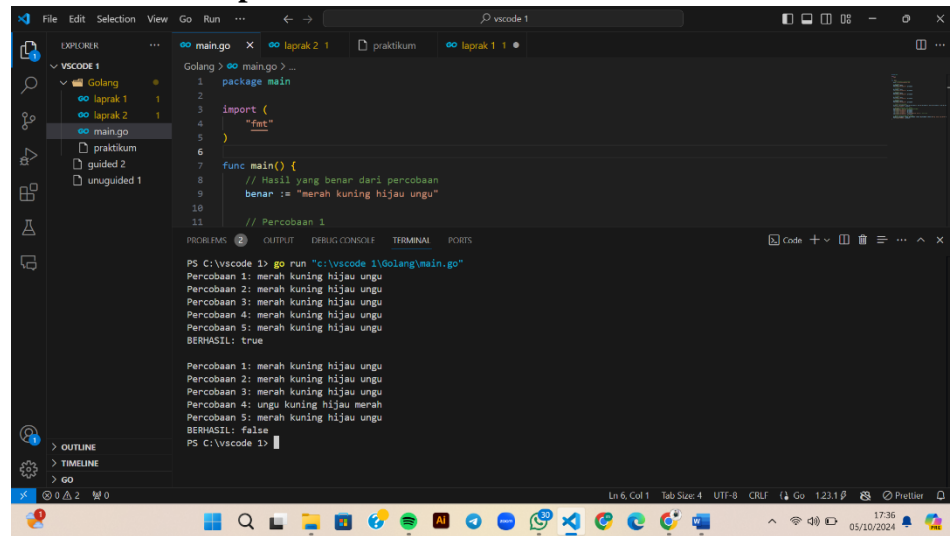
    // Percobaan 5
    percobaan5 := benar
    fmt.Println("Percobaan 5:", percobaan5)

    // Mengecek hasil percobaan
    berhasil := percobaan1 == benar && percobaan2 == benar &&
    percobaan3 == benar && percobaan4 == benar && percobaan5 ==
    benar
    fmt.Println("BERHASIL:", berhasil)

    // Percobaan dengan hasil yang berbeda
    fmt.Println() // Memberikan jarak antara percobaan
    fmt.Println("Percobaan 1:", percobaan1)
    fmt.Println("Percobaan 2:", percobaan2)
    fmt.Println("Percobaan 3:", percobaan3)
    fmt.Println("Percobaan 4:", "ungu kuning hijau merah") //
    Hasil salah
    fmt.Println("Percobaan 5:", percobaan5)

    // Mengecek hasil percobaan yang berbeda
    berhasil = percobaan1 == benar && percobaan2 == benar &&
    percobaan3 == benar && "ungu kuning hijau merah" == benar &&
    percobaan5 == benar
    fmt.Println("BERHASIL:", berhasil)
}
```

## Screenshoot Output



```
1 package main
2
3 import (
4     "fmt"
5 )
6
7 func main() {
8     // Hasil yang benar dari percobaan
9     benar := "merah kuning hijau ungu"
10
11     // Percobaan 1
12
13     PS C:\vscode 1> go run "C:\vscode 1\golang\main.go"
14     Percobaan 1: merah kuning hijau ungu
15     Percobaan 2: merah kuning hijau ungu
16     Percobaan 3: merah kuning hijau ungu
17     Percobaan 4: merah kuning hijau ungu
18     Percobaan 5: merah kuning hijau ungu
19     BERNHASIL: true
20
21     Percobaan 1: merah kuning hijau ungu
22     Percobaan 2: merah kuning hijau ungu
23     Percobaan 3: merah kuning hijau ungu
24     Percobaan 4: ungu kuning hijau merah
25     Percobaan 5: merah kuning hijau ungu
26     BERNHASIL: false
27     PS C:\vscode 1>
```

**Deskripsi Program:** Program diatas menggunakan variable string “merah, kuning, hijau, ungu”. Kemudian terdapat lima percobaan dan kemudian akan dicek oleh variable, apakah nilai tersebut benar atau salah. Program mencetak Kembali hasil dari semua percobaan dan mencetak hasilnya.

## 2. Soal Latihan modul 2B. NO 2

### Source Code

```
package main

import (
    "fmt"
    "strings"
)

func main() {
    var N_2311102217 int
    var bunga string
    pita := ""

    fmt.Print("Masukkan jumlah bunga: ")
    fmt.Scanln(&N_2311102217)

    // Memastikan input adalah bilangan positif
    if N_2311102217 <= 0 {
        fmt.Println("Jumlah bunga harus bilangan bulat positif.")
        return
    }

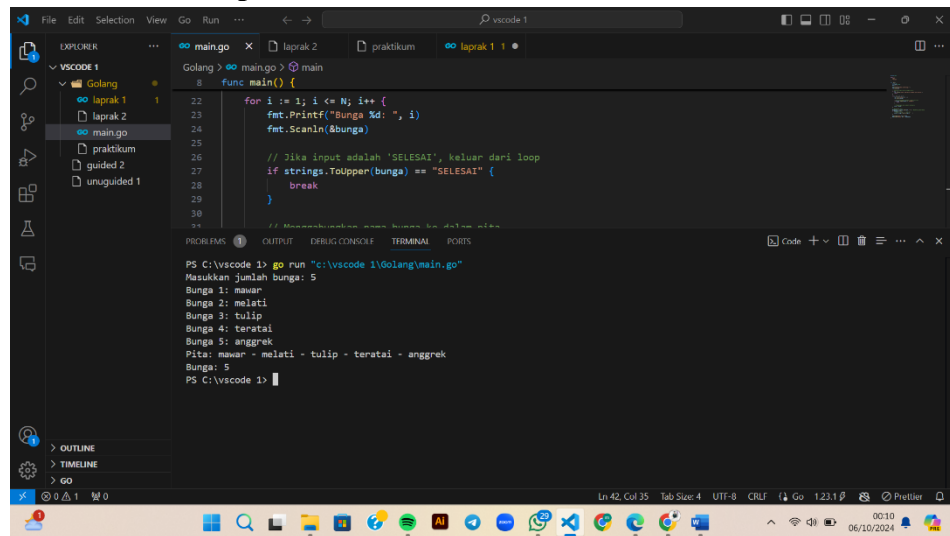
    for i := 1; i <= N_2311102217; i++ {
        fmt.Printf("Bunga %d: ", i)
        fmt.Scanln(&bunga)

        // Jika input adalah 'SELESAI', keluar dari loop
    }
}
```



```
        if strings.ToUpper(bunga) == "SELESAI" {  
            break  
        }  
  
        // Menggabungkan nama bunga ke dalam pita  
        pita += bunga + " - "  
    }  
  
    // Menampilkan hasil  
    bungaCount := strings.Count(pita, "-") // Menghitung jumlah  
bunga  
    if bungaCount > 0 {  
        // Hapus trailing " - " jika ada  
        pita = strings.TrimSuffix(pita, " - ")  
    }  
  
    fmt.Printf("Pita: %s\n", pita)  
    fmt.Printf("Bunga: %d\n", bungaCount)  
}
```

## Screenshoot Output



The screenshot shows the Visual Studio Code editor with a Go project. The Explorer sidebar on the left shows the file structure: `main.go`, `laprak 1`, `laprak 2`, `praktikum`, `guided 2`, and `unguided 1`. The main editor displays the code for `main.go`:

```
func main() {  
    for i := 1; i <= N; i++ {  
        fmt.Printf("Bunga %d: ", i)  
        fmt.Scanln(&bunga)  
  
        // Jika input adalah 'SELESAI', keluar dari loop  
        if strings.ToUpper(bunga) == "SELESAI" {  
            break  
        }  
    }  
  
    // Menampilkan nama bunga ke dalam array  
    pita = append(pita, bunga)  
}
```

The Output window at the bottom shows the execution results:

```
PS C:\vscode> go run "c:\vscode\1\golang\main.go"  
Masukkan jumlah bunga: 5  
Bunga 1: mawar  
Bunga 2: melati  
Bunga 3: tulip  
Bunga 4: teratai  
Bunga 5: anggrek  
Pita: mawar - melati - tulip - teratai - anggrek  
Bunga: 5  
PS C:\vscode>
```

**Deskripsi Program:** Program tersebut mendeklarasikan variable `N_2311102217` untuk jumlah bunga yang akan dimasukkan, variable menyimpan nama bunga, dan variable `pita` untuk menyimpan nama bunga yang digabungkan. Pengguna diminta untuk memasukkan jumlah bunga yang di inginkan. Kemudian dengan menggunakan loop program meminta pengguna untuk memasukkan nama bunga satu persatu, jika pengguna memasukkan “SELESAI” maka loop akan berhenti. Setelah selesai program akan menghitung dan menampilkan jumlah bunga yang telah dimasukkan.

### 3. Soal Latihan modul 2B. NO 3

#### Source Code

```
package main

import (
    "fmt"
)

func main() {
    var beratKantong1_2311102217,
    beratKantong2_2311102217 float64

    for {
        fmt.Print("Masukan berat belanjaan di kedua
kantong: ")
        _, err := fmt.Scan(&beratKantong1_2311102217,
&beratKantong2_2311102217)

        if err != nil {
            fmt.Println("Input tidak valid, silakan
masukkan angka.")
            continue
        }

        // Memeriksa jika salah satu kantong beratnya
negatif
        if beratKantong1_2311102217 < 0 ||
beratKantong2_2311102217 < 0 {
            fmt.Println("Salah satu kantong beratnya
negatif. Proses selesai.")
            break
        }

        // Menghitung total berat
        totalBerat := beratKantong1_2311102217 +
beratKantong2_2311102217

        // Memeriksa jika total berat melebihi 150 kg
        if totalBerat > 150 {
            fmt.Println("Total berat isi kantong melebihi
150 kg. Proses selesai.")
            break
        }

        // Menghitung selisih berat
        selisih := beratKantong1_2311102217 -
beratKantong2_2311102217

        // Memeriksa jika selisih lebih dari atau sama
dengan 9 kg
        if selisih >= 9 || selisih <= -9 {
            fmt.Println("Sepeda motor pak Andi akan
oleng: true")
        } else {
            fmt.Println("Sepeda motor pak Andi akan
oleng: false")
        }
    }
}
```

```

    }

    fmt.Println("Proses selesai.")
}

```

## Screenshoot Output

```

7 func main() {
8
9     // Memeriksa jika selisih lebih dari atau sama dengan 9 kg
10    if selisih >= 9 || selisih <= -9 {
11        fmt.Println("Sepeda motor pak Andi akan oleng: true")
12    } else {
13        fmt.Println("Sepeda motor pak Andi akan oleng: false")
14    }
15
16    fmt.Println("Proses selesai.")
17 }

```

```

PS C:\vscode 1> go run "C:\vscode 1\Golang\main.go"
Masukan berat belanjaan di kedua kantong: 9.2 23.2
Sepeda motor pak Andi akan oleng: true
Masukan berat belanjaan di kedua kantong: 5 10
Sepeda motor pak Andi akan oleng: false
Masukan berat belanjaan di kedua kantong: 55.6 72.8
Sepeda motor pak Andi akan oleng: true
Masukan berat belanjaan di kedua kantong: 59.5 98.7
Total berat isi kantong melebihi 150 kg. Proses selesai.
Proses selesai.
PS C:\vscode 1>

```

**Deskripsi Program:** Program tersebut mendeklarasikan dua variable yaitu beratKantong1 2311102217 dan beratKantong2 2311102217 untuk menyimpan berat kedua kantong. Program menggunakan loop untuk meminta pengguna memasukkan berat kedua kantong. Program juga memeriksa apakah salah satu berat kantong negative, jika iya akan dihentikan prosesnya dan jika total berat lebih dari 150, proses juga dihentikan. Program menghitung selisih kedua berat, jika berat lebih atau sama dengan 9 kg, program mencetak pesan “motor pak andi akan oleng” jika tidak program mencetak “sepeda motor pak andi stabil”.

#### 4. Soal Latihan modul 2B. NO 4

##### Source Code

```
package main

import (
    "fmt"
    "math"
)

func main() {
    var K_2311102217 int

    // Meminta input dari pengguna
    fmt.Print("Masukkan nilai K: ")
    fmt.Scan(&K_2311102217)

    // Menghitung nilai f(K) sesuai persamaan
    fK := float64(4*K_2311102217+2) /
float64((K_2311102217+1)*(K_2311102217+3))

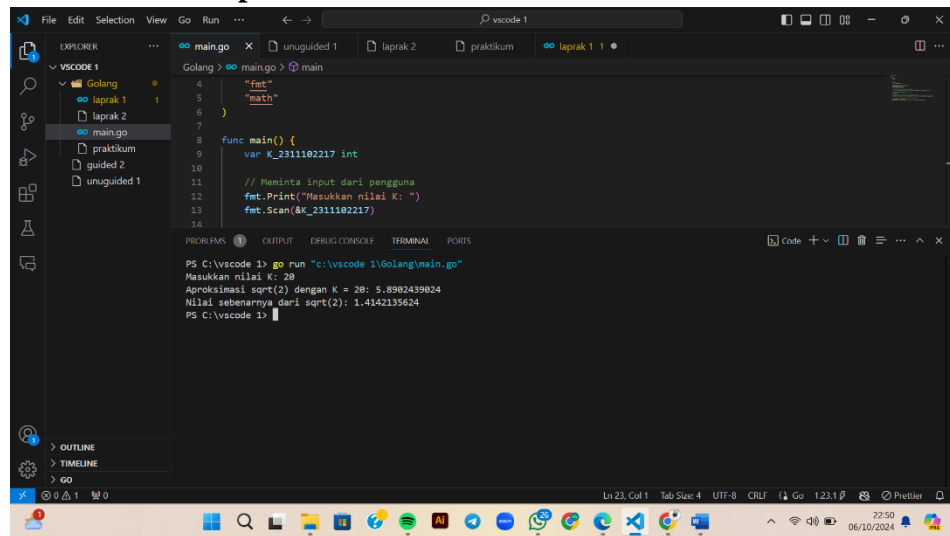
    // Menghitung nilai aproksimasi sqrt(2)
    approxSqrt2 := 1 / fK

    // Menampilkan hasil dengan presisi 10 angka desimal
    fmt.Printf("Aproksimasi sqrt(2) dengan K = %d: %.10f\n",
K_2311102217, approxSqrt2)

    // Menampilkan nilai sebenarnya untuk perbandingan
    actualSqrt2 := math.Sqrt(2)

    fmt.Printf("Nilai sebenarnya dari sqrt(2): %.10f\n",
actualSqrt2)
}
```

## Screenshoot Output



```
4  "fmt"
5  "math"
6
7
8  func main() {
9      var K_2311102217 int
10
11     // Meminta input dari pengguna
12     fmt.Print("Masukkan nilai K: ")
13     fmt.Scan(&K_2311102217)
14
15     // Hitung nilai f(k) menggunakan rumus
16     // yang diberikan
17     fK := 4*K_2311102217 / (K_2311102217+1)*(K_2311102217+3)
18
19     // Hitung aproksimasi akar kuadrat 2
20     approxSqrt2 := 1/fK
21
22     // Cetak nilai aproksimasinya
23     fmt.Println("Aproksimasi sqrt(2) dengan K =", K_2311102217, ":", approxSqrt2)
24 }
```

PS C:\vscode 1> go run "c:\vscode 1\Golang\main.go"

Masukkan nilai K: 20

Aproksimasi sqrt(2) dengan K = 20: 1.4142135624

Nilai sebenarnya dari sqrt(2): 1.4142135624

PS C:\vscode 1>

**Deskripsi Program:** Program mendeklarasikan variable K\_2311102217 untuk menyimpan input dari pengguna. Pengguna diminta untuk memasukkan nilai K. Program menghitung nilai f(k) menggunakan rumus yang diberikan

$$f(K) = 4K + 2 / (K + 1)(K + 3)$$

Kemudian menghitung aproksimasi akar kuadrat 2

$$f(K): \text{approxSqrt2} = 1/f(k)$$

Program akan mencetak nilai aproksimasinya

## 5. Soal Latihan modul 2A. NO 4

### Source Code

```
package main

import (
    "fmt"
)

func main() {
    var Celsius_2311102217 float64

    // Meminta input suhu dalam derajat Celsius
    fmt.Print("Masukkan suhu dalam derajat Celsius: ")
    _, err := fmt.Scan(&Celsius_2311102217)

    if err != nil {
        fmt.Println("Input tidak valid. Silakan masukkan angka.")
        return
    }

    // Menghitung konversi
    fahrenheit := (Celsius_2311102217 * 9 / 5) + 32
    reamur := Celsius_2311102217 * 4 / 5
    kelvin := Celsius_2311102217 + 273.15

    // Menampilkan hasil konversi
    fmt.Printf("Temperatur Celsius_2311102217: %.2f\n", Celsius_2311102217)

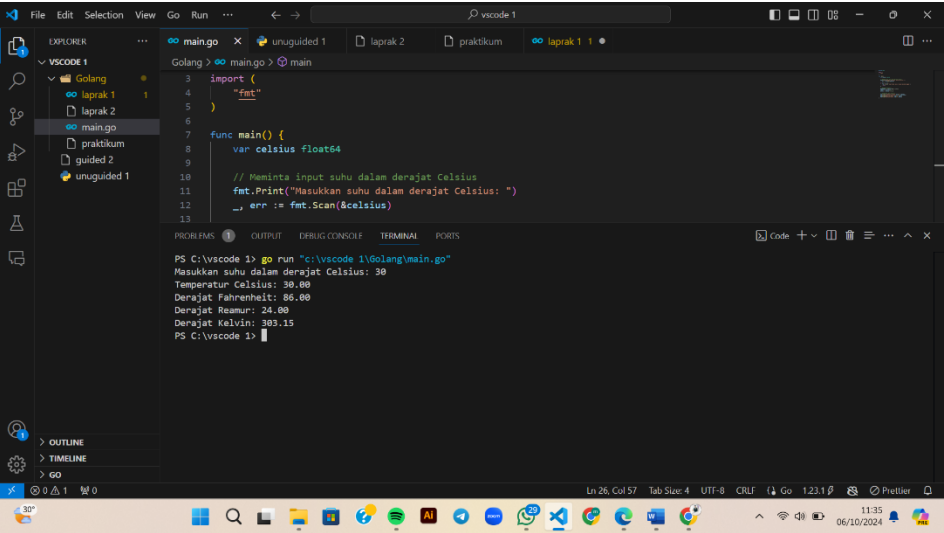
    fmt.Printf("Derajat Fahrenheit: %.2f\n", fahrenheit)
```

```
fmt.Printf("Derajat Reamur: %.2f\n", reamur)

fmt.Printf("Derajat Kelvin: %.2f\n", kelvin)

}
```

**Screenshoot Output**



**Deskripsi Program:** Program mendeklarasikan variable Celcius\_2311102217 untuk menyimpan input suhu dalam derajat. Pengguna diminta untuk memasukkan suhu dalam derajat celcius. Kemudian program menghitung suhu tiga sklaa yaitu Fahrenheit, reamur, dan kelvin. Program akan mencetak suhu dalam derajat celcius dan konversi ke derajat Fahrenheit, reamur, dan kelvin.



## 6. Soal Latihan modul 2A. NO 5

### Source Code

```
package main

import (
    "fmt"
)

func main() {
    var angka_2311102217 [5]int
    var karakter [3]rune

    // Membaca 5 buah data integer
    fmt.Println("Masukkan 5 angka (dari 32 s.d. 127):")
    for i := 0; i < 5; i++ {
        fmt.Scan(&angka_2311102217[i])
    }

    // Membaca 3 buah karakter
    fmt.Println("Masukkan 3 karakter berdampingan (tanpa spasi):")
    var input string
    fmt.Scan(&input)

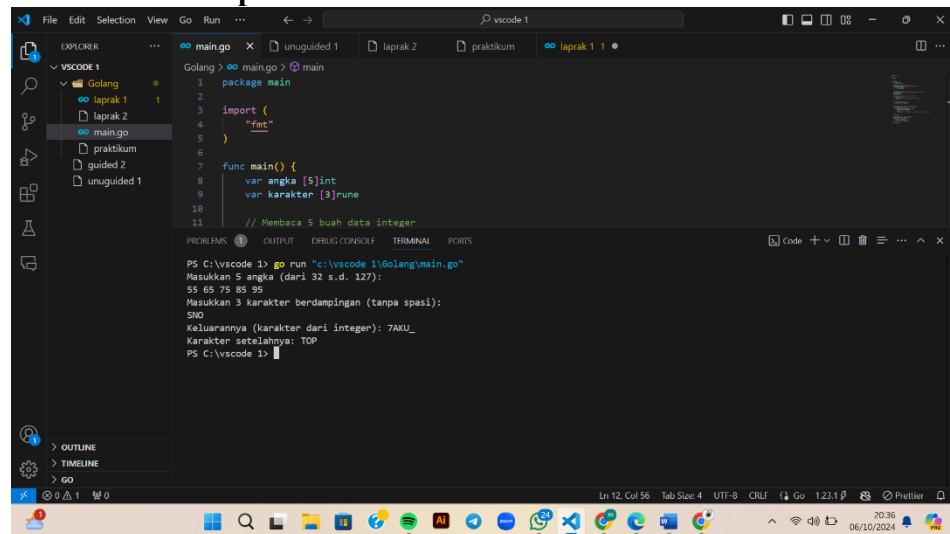
    // Memastikan input memiliki tepat 3 karakter
    if len(input) < 3 {
        fmt.Println("Input harus terdiri dari 3 karakter.")
        return
    }

    for i := 0; i < 3; i++ {
        karakter[i] = rune(input[i])
    }

    // Menampilkan hasil untuk 5 buah data integer
    fmt.Print("Keluarannya (karakter dari integer): ")
    for i := 0; i < 5; i++ {
        if angka_2311102217[i] >= 32 && angka_2311102217[i] <=
127 {
            fmt.Printf("%c", angka_2311102217[i])
        } else {
            fmt.Print("?") // Menampilkan tanda tanya untuk
nilai di luar range
        }
    }
    fmt.Println()

    // Menampilkan hasil untuk 3 karakter
    fmt.Print("Karakter setelahnya: ")
    for i := 0; i < 3; i++ {
        nextChar := karakter[i] + 1
        fmt.Printf("%c", nextChar)
    }
    fmt.Println()
}
```

## Screenshoot Output



```
1 package main
2
3 import (
4     "fmt"
5 )
6
7 func main() {
8     var angka [5]int
9     var karakter [3]rune
10
11     // Membaca 5 buah data integer
12
13     PS C:\vscode 1> go run "c:\vscode 1\Golang\main.go"
14     Masukkan 5 angka (dari 32 s.d. 127):
15     55 65 75 85 95
16     Masukkan 3 karakter berdampingan (tanpa spasi):
17     7AKU_
18     Keluarannya (karakter dari integer): 7AKU_
19     Karakter setelahnya: TOP
20     PS C:\vscode 1>
```

**Deskripsi Program:** Program mendeklarasikan dua array yaitu `angka_2311102217` untuk menyimpan 5 integer dan karakter untuk menyimpan karakter. Pengguna diminta untuk memasukkan 5 angka antara 32-127 dan diminta untuk memasukkan 3 karakter secara berdampingan tanpa spasi. Program akan mencetak setiap angka dan karakter yang dimasukkan.