

**LAPORAN PRAKTIKUM  
ALGORITMA DAN PEMROGRAMAN 2**

**MODUL II**

**REVIEW STRUKTUR KONTROL**



Oleh:

FAJAR FARIZQI AZMI

2311102192

IF-11-02

**S1 TEKNIK INFORMATIKA**

**INSTITUT TEKNOLOGI TELKOM PURWOKERTO**

**2024**

## **I. DASAR TEORI**

### Dasar Teori

#### **Struktur Kontrol Golang**

Struktur kontrol merupakan komponen penting dalam bahasa pemrograman karena berfungsi untuk mengatur alur eksekusi program. Dalam bahasa Go (Golang), struktur kontrol yang tersedia mencakup pengulangan (loops), seleksi (conditional statements), dan berbagai bentuk pengendalian alur lainnya, seperti goroutine untuk konkuren.

#### **Pengulangan (Loops)**

Dalam Golang, pengulangan diimplementasikan melalui perintah `for`. Menariknya, Go tidak memiliki perintah `while` atau `do-while` seperti pada beberapa bahasa lainnya. Sebaliknya, semua bentuk pengulangan diimplementasikan menggunakan variasi dari pernyataan `for`. Bentuk dasar pengulangan menggunakan sintaks `for init; condition; post { }` yang serupa dengan C dan Java. Golang juga mendukung pengulangan tanpa kondisi yang akan terus berjalan sampai ada perintah `break` untuk menghentikannya.

#### **Seleksi (Conditional Statements)**

Instruksi kondisional `if` digunakan dalam Golang untuk menjalankan kode berdasarkan kondisi tertentu. Golang juga menyediakan `else if` dan `else` untuk menangani berbagai kondisi. Uniknya, Golang memperkenalkan penggunaan statement pendek sebelum kondisi `if`, yang memungkinkan deklarasi variabel yang hanya relevan di dalam blok kondisional. Selain itu, Golang memiliki `switch` statement yang digunakan untuk memilih salah satu dari beberapa blok kode untuk dijalankan berdasarkan nilai dari ekspresi.

#### **Goroutine dan Pengendalian Konkuren**

Salah satu fitur kontrol alur yang membedakan Golang adalah dukungannya terhadap concurrency melalui goroutine. Goroutine memungkinkan program menjalankan fungsi secara asinkron, tanpa perlu menunggu fungsi lain selesai. Dikombinasikan dengan kanal (channels), goroutine

memberikan mekanisme sinkronisasi yang efisien. Penggunaan kanal untuk berkomunikasi antara goroutine menghilangkan banyak masalah yang biasanya terkait dengan penguncian data (locking) pada program yang menjalankan paralelisme

Dengan fitur-fitur ini, Golang memberikan fleksibilitas dan efisiensi dalam pengelolaan alur program, menjadikannya pilihan populer untuk pengembangan aplikasi yang memerlukan kinerja tinggi dan skalabilitas, seperti layanan web, alat otomatisasi, dan sistem terdistribusi

## II. GUIDED

### GUIDED 1

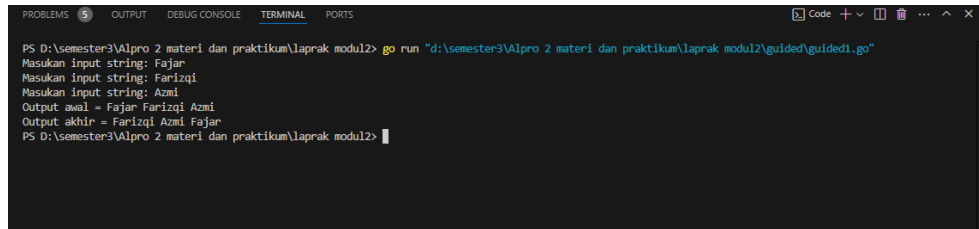
#### Source code

```
package main

import "fmt"

func main() {
    var (
        satu, dua, tiga string
        temp      string
    )
    fmt.Print("Masukan input string: ")
    fmt.Scanln(&satu)
    fmt.Print("Masukan input string: ")
    fmt.Scanln(&dua)
    fmt.Print("Masukan 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)
}
```

## Screenshot output



```
PS D:\semester3\Alpro 2 materi dan praktikum\laprak modul2> go run "d:\semester3\Alpro 2 materi dan praktikum\laprak modul2\guided\guided1.go"
Masukan input string: Fajar
Masukan input string: Farizqi
Masukan input string: Azmi
Output awal = Fajar Farizqi Azmi
Output akhir = Farizqi Azmi Fajar
PS D:\semester3\Alpro 2 materi dan praktikum\laprak modul2>
```

## Deskripsi program

Program ini melakukan pertukaran nilai tiga string yang dimasukkan oleh pengguna, menampilkan urutan sebelum dan sesudah penukaran.

## GUIDED II

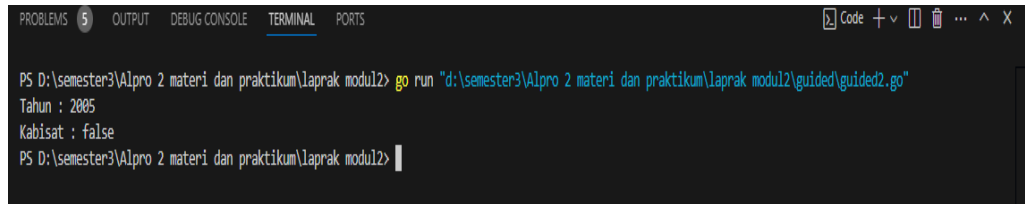
### Source code

```
package main

import "fmt"

func main() {
    var tahun int
    fmt.Print("Tahun : ")
    fmt.Scan(&tahun)
    fmt.Printf("Kabisat : %t \n", (tahun%4 == 0 && (tahun%100 != 0 ||
tahun%400 == 0)))
}
```

## Screenshot output



```
PROBLEMS 5 OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS D:\semester3\Alpro 2 materi dan praktikum\laprak modul2> go run "d:\semester3\Alpro 2 materi dan praktikum\laprak modul2\guided\guided2.go"
Tahun : 2005
Kabisat : false
PS D:\semester3\Alpro 2 materi dan praktikum\laprak modul2> |
```

## Deskripsi program

Program ini memeriksa apakah sebuah tahun adalah tahun kabisat dengan menggunakan aturan pembagian dan menampilkan hasil dalam bentuk boolean (true atau false).

## GUIDED III

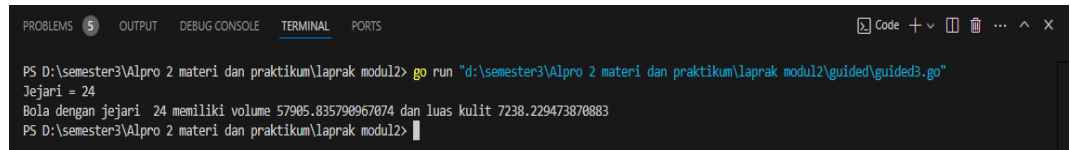
### Source code

```
package main

import (
    "fmt"
    "math"
)

func main() {
    var r float64
    fmt.Print("Jejari = ")
    fmt.Scan(&r)
    volume := 4.0 / 3.0 * math.Pi * math.Pow(r, 3)
    luas := 4 * math.Pi * math.Pow(r, 2)
    fmt.Println("Bola dengan jejari ", r, "memiliki volume", volume,
        "dan luas kulit", luas)
}
```

## Screenshot output



```
PROBLEMS 5 OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS D:\semester3\Alpro 2 materi dan praktikum\laprak modul2> go run "d:\semester3\Alpro 2 materi dan praktikum\laprak modul2\guided\guided3.go"
Jejari = 24
Bola dengan jejari 24 memiliki volume 57905.835790967074 dan luas kulit 7238.229473870883
PS D:\semester3\Alpro 2 materi dan praktikum\laprak modul2>
```

## Deskripsi program

Program ini menerima input berupa jejari bola dari pengguna, menghitung volume dan luas permukaan bola menggunakan rumus matematika, dan menampilkan hasil perhitungannya.

### III. UNGUIDED

#### ➤ UNGUIDED 2B 1

Source code :

```
//fajar farizqi azmi
//2311102192

package main

import (
    "bufio"
    "fmt"
    "os"
    "strings"
)

func main() {

    correctOrder := []string{"merah", "kuning", "hijau", "ungu"}

    reader := bufio.NewReader(os.Stdin)
    var hadError bool

    for i := 1; i <= 5; i++ {
        fmt.Printf("Percobaan %d: ", i)

        input, _ := reader.ReadString('\n')
        input = strings.TrimSpace(input)
```



```
colors := strings.Split(input, " ")

for j := 0; j < 4; j++ {
    if colors[j] != correctOrder[j] {
        hadError = true
        break
    }
}

if !hadError {
    fmt.Println("BERHASIL: true")
} else {
    fmt.Println("BERHASIL: false")
}
}
```

**Screenshot output :**

```
PS D:\semester3\Alpro 2 materi dan praktikum\laprak modul2> go run "d:\semester3\Alpro 2 materi dan praktikum\laprak modul2\unguided2b\unguided2
b1.go"
Percobaan 1: merah kuning hijau ungu
Percobaan 2: merah kuning hijau ungu
Percobaan 3: merah kuning hijau ungu
Percobaan 4: merah kuning hijau ungu
Percobaan 5: merah kuning hijau ungu
BERHASIL: true
PS D:\semester3\Alpro 2 materi dan praktikum\laprak modul2>
```

```
PS D:\semester3\Alpro 2 materi dan praktikum\laprak modul2> go run "d:\semester3\Alpro 2 materi dan praktikum\laprak modul2\unguided2b\unguided2
b1.go"
Percobaan 1: merah kuning hijau ungu
Percobaan 2: merah kuning hijau ungu
Percobaan 3: merah kuning hijau ungu
Percobaan 4: merah kuning hijau ungu
Percobaan 5: merah kuning hijau pink
BERHASIL: false
PS D:\semester3\Alpro 2 materi dan praktikum\laprak modul2> █
```

## Deskripsi program

Program ini adalah permainan sederhana untuk memverifikasi apakah pengguna dapat memasukkan urutan warna yang benar sebanyak 5 kali. Jika semua percobaan benar, program akan menampilkan "BERHASIL: true", dan jika ada kesalahan, program akan menampilkan "BERHASIL: false".

### ➤ UNGUIDED 2b 2

#### Source code :

```
// fajar farizqi azmi
// 2311102192

package main

import (
    "bufio"
    "fmt"
    "os"
    "strings"
)

func main() {

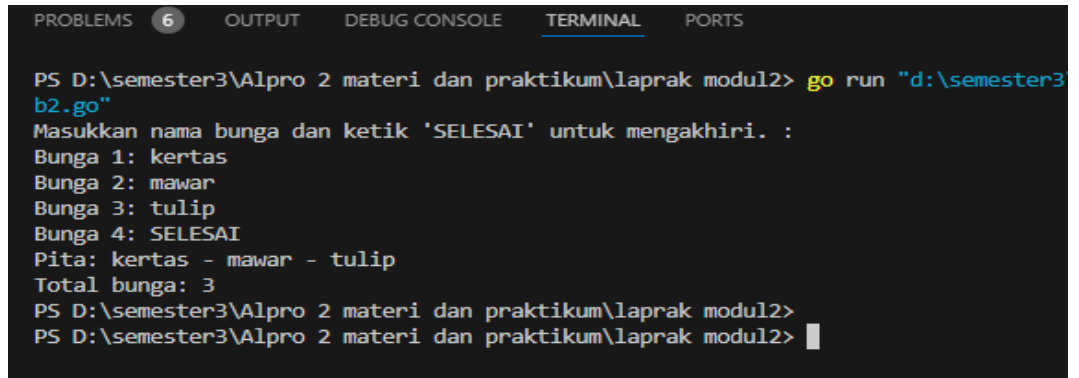
    var pita string
    var TotalBunga int

    scanner := bufio.NewScanner(os.Stdin)

    fmt.Print("Masukkan nama bunga dan ketik 'SELESAI' untuk mengakhiri. :\n")
```

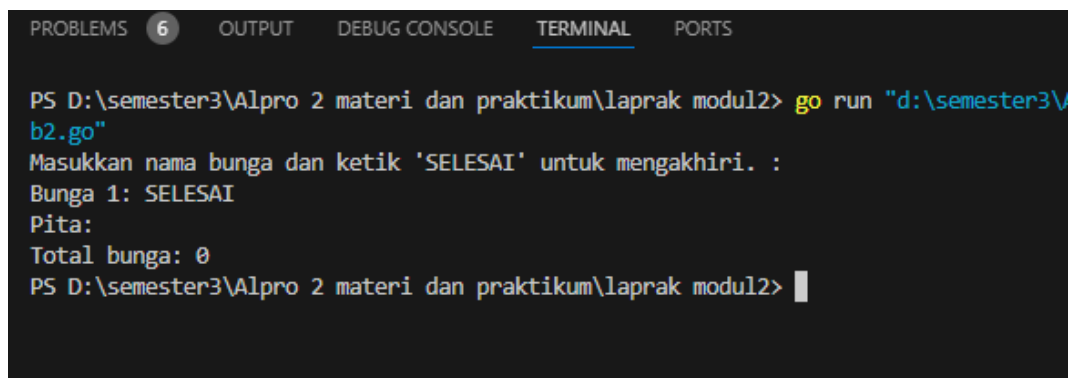
```
for {  
    TotalBunga++  
    fmt.Printf("Bunga %d: ", TotalBunga)  
  
    scanner.Scan()  
    input := scanner.Text()  
  
    if strings.ToUpper(input) == "SELESAI" {  
        TotalBunga--  
        break  
    }  
  
    if pita == "" {  
        pita = input  
    } else {  
        pita += " - " + input  
    }  
}  
  
fmt.Println("Pita:", pita)  
fmt.Println("Total bunga:", TotalBunga )  
}
```

### Screenshot output :



```
PROBLEMS 6 OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS D:\semester3\Alpro 2 materi dan praktikum\laprak modul2> go run "d:\semester3\
b2.go"
Masukkan nama bunga dan ketik 'SELESAI' untuk mengakhiri. :
Bunga 1: kertas
Bunga 2: mawar
Bunga 3: tulip
Bunga 4: SELESAI
Pita: kertas - mawar - tulip
Total bunga: 3
PS D:\semester3\Alpro 2 materi dan praktikum\laprak modul2>
PS D:\semester3\Alpro 2 materi dan praktikum\laprak modul2>
```



```
PROBLEMS 6 OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS D:\semester3\Alpro 2 materi dan praktikum\laprak modul2> go run "d:\semester3\
b2.go"
Masukkan nama bunga dan ketik 'SELESAI' untuk mengakhiri. :
Bunga 1: SELESAI
Pita:
Total bunga: 0
PS D:\semester3\Alpro 2 materi dan praktikum\laprak modul2>
PS D:\semester3\Alpro 2 materi dan praktikum\laprak modul2>
```

### Deskripsi program :

- Program ini mengumpulkan nama bunga dari pengguna dalam format string yang dipisahkan dengan tanda -
- Program menghitung jumlah bunga yang dimasukkan dan menampilkan daftar bunga serta total bunga yang telah dimasukkan ketika pengguna mengetik "SELESAI".

➤ **Unguided 2b 3**

**Source code :**

```
// fajar farizqi azmi
// 2311102192

package main

import "fmt"

func main() {
    for {
        var berat1, berat2 float64

        fmt.Print("Masukkan berat belanjaan di kedua
kantong: ")
        _, err := fmt.Scan(&berat1, &berat2)
        if err != nil {
            fmt.Println("Input tidak valid.")
            return
        }

        if berat1+berat2 > 150 || berat1 < 0 || berat2 < 0 {
            fmt.Println("Proses selesai.")
            break
        }

        selisih := berat1 - berat2
        if selisih < 0 {
            selisih = -selisih
        }
    }
}
```

```

        akanOlang := selisih >= 9

        fmt.Printf("Sepeda motor pak Andi akan oleng:
        %t\n", akanOlang)
    }
}

```

### Screenshot output :

```

PROBLEMS 8 OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS D:\semester3\Alpro 2 materi dan praktikum\laprak modul2> go run "d:\semester3\b3.go"
Masukkan berat belanjaan di kedua kantong: 5 10
Sepeda motor pak Andi akan oleng: false
Masukkan berat belanjaan di kedua kantong: 55.6 70.2
Sepeda motor pak Andi akan oleng: true
Masukkan berat belanjaan di kedua kantong: 72.3 66.9
Sepeda motor pak Andi akan oleng: false
Masukkan berat belanjaan di kedua kantong: 59.5 98.7
Proses selesai.
PS D:\semester3\Alpro 2 materi dan praktikum\laprak modul2>

```

### Deskripsi Program :

- Program ini memberikan cara yang sederhana untuk menentukan keseimbangan berat di dua kantong belanjaan. Jika total berat melebihi 150 kg atau salah satu kantong memiliki berat negatif, program akan berhenti. Jika selisih berat antara kedua kantong lebih dari atau sama dengan 9 kg, sepeda motor dianggap akan oleng, dan hasilnya ditampilkan kepada pengguna.

### ➤ Unguided 2b 4

**Source code :**

```
// fajar farizqi azmi
// 2311102192

package main

import (
    "fmt"
    "math"
)

func main() {
    var k float64

    fmt.Print("Masukkan nilai K: ")
    fmt.Scan(&k)

    fk := math.Pow(4*k+2, 2) / ((4*k + 1) * (4*k + 3))

    fmt.Printf("Nilai f(k) = %.10f\n",fk)
}
```

**Sesudah di modifikasi :**

```
// fajar farizqi azmi
// 2311102192

package main

import (
```

```

        "fmt"
        "math"
    )

func main() {
    var k int

    for {
        fmt.Print("Masukkan nilai K: ")
        _, err := fmt.Scan(&k)
        if err != nil {
            fmt.Println("Input tidak valid.")
            return
        }

        if k < 0 {
            fmt.Println("Nilai K harus positif.")
            continue
        }

        akarDua := 0.0
        for i := 0; i <= k; i++ {
            pembilang := math.Pow(4*float64(i)+2, 2)
            penyebut := (4*float64(i) + 1) * (4*float64(i)
+ 3)

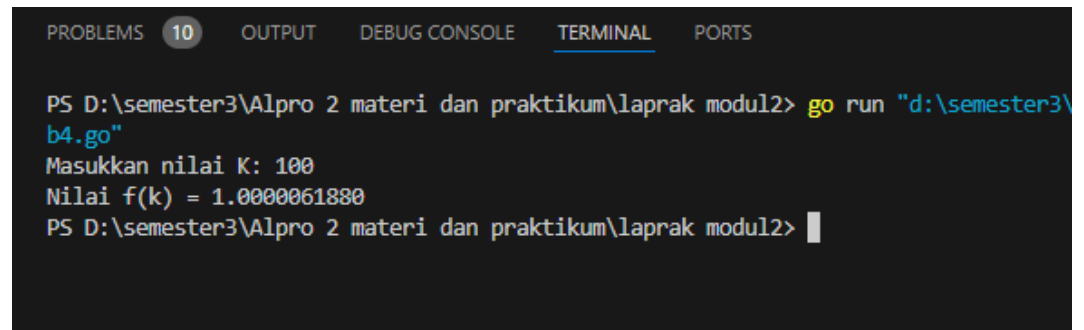
            akarDua += pembilang / penyebut
        }

        fmt.Printf("Nilai akar 2 = %.10f\n", akarDua)
    }
}

```



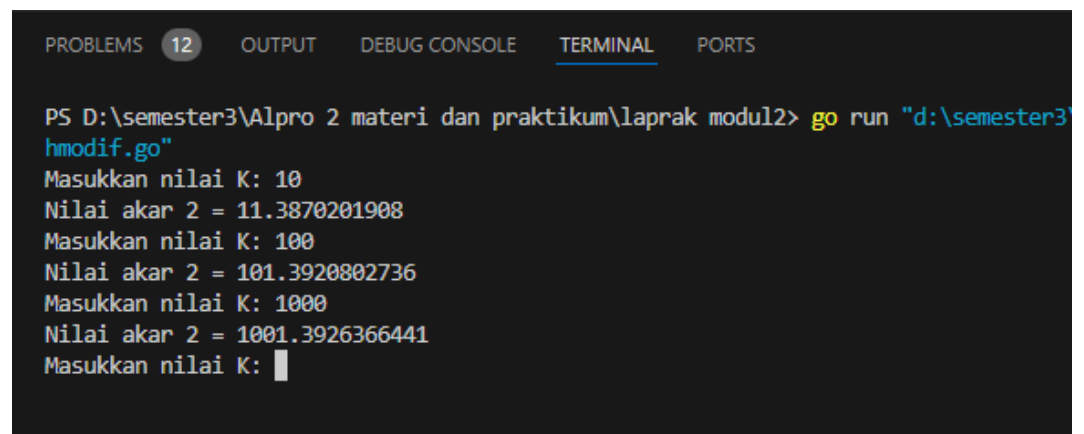
### Screenshot output :



```
PROBLEMS 10 OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS D:\semester3\Alpro 2 materi dan praktikum\laprak modul2> go run "d:\semester3\
b4.go"
Masukkan nilai K: 100
Nilai f(k) = 1.0000061880
PS D:\semester3\Alpro 2 materi dan praktikum\laprak modul2> |
```

### Sesudah di modif :



```
PROBLEMS 12 OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS D:\semester3\Alpro 2 materi dan praktikum\laprak modul2> go run "d:\semester3\
hmodif.go"
Masukkan nilai K: 10
Nilai akar 2 = 11.3870201908
Masukkan nilai K: 100
Nilai akar 2 = 101.3920802736
Masukkan nilai K: 1000
Nilai akar 2 = 1001.3926366441
Masukkan nilai K: |
```

### Deskripsi program :

Program Go ini meminta pengguna untuk memasukkan nilai K, kemudian menghitung nilai  $f(k)$  menggunakan rumus matematika yang ditentukan. Hasil perhitungan ditampilkan dengan presisi 10 desimal. Program ini berfungsi untuk melakukan evaluasi fungsi matematika berdasarkan input yang diberikan.

setelah dimodifikasi Program Go ini meminta pengguna untuk memasukkan nilai K dan menghitung nilai perkiraan akar dua berdasarkan rumus tertentu. Program melakukan validasi untuk memastikan K adalah angka positif. Selama loop, program menghitung

nilai berdasarkan rumus yang melibatkan penjumlahan pecahan hingga K dan menampilkan hasilnya dengan presisi 10 desimal. Program ini terus meminta input hingga pengguna memilih untuk menghentikannya.

➤ **Uguided 2c 1**

**Sourch code :**

```
// Fajar farizqi azmi
// 2311102192

package main

import "fmt"

func hitungBiaya(beratGram int) (int, int, int) {

    beratKg := beratGram / 1000
    sisaGram := beratGram % 1000

    biayaKg := beratKg * 10000

    var biayaSisa int
    if sisaGram >= 500 {
        biayaSisa = sisaGram * 5
    } else if sisaGram > 0 && beratKg <= 10 {
        biayaSisa = sisaGram * 15
    } else {
        biayaSisa = 0
    }
}
```

```

        return biayaKg, biayaSisa, beratKg
    }

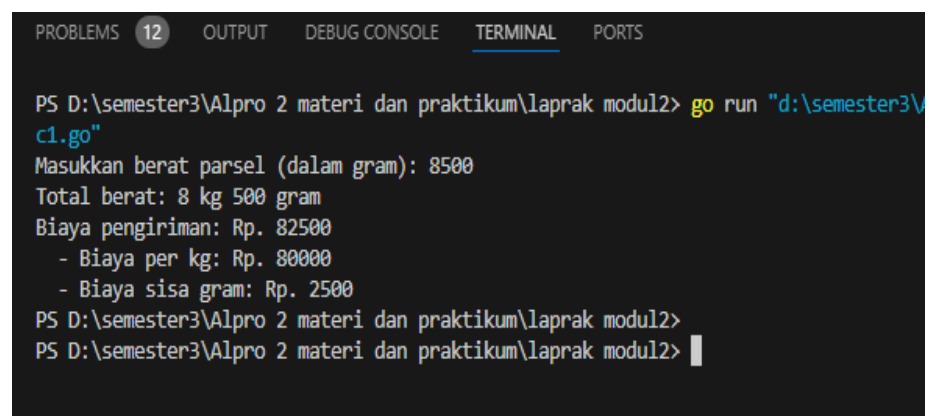
    func main() {
        var beratGram int
        fmt.Print("Masukkan berat parcel (dalam gram):")
        "\n"
        fmt.Scanln(&beratGram)

        biayaKg, biayaSisa, beratKg :=
            hitungBiaya(beratGram)

        fmt.Printf("Total berat: %d kg %d gram\n",
            beratKg, beratGram%1000)
        fmt.Printf("Biaya pengiriman: Rp. %d\n",
            biayaKg+biayaSisa)
        fmt.Printf(" - Biaya per kg: Rp. %d\n", biayaKg)
        fmt.Printf(" - Biaya sisa gram: Rp. %d\n",
            biayaSisa)
    }

```

### Sceenshot output :



```

PROBLEMS 12 OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS D:\semester3\Alpro 2 materi dan praktikum\laprak modul2> go run "d:\semester3\Alpro 2 materi dan praktikum\laprak modul2\c1.go"
Masukkan berat parcel (dalam gram): 8500
Total berat: 8 kg 500 gram
Biaya pengiriman: Rp. 82500
 - Biaya per kg: Rp. 80000
 - Biaya sisa gram: Rp. 2500
PS D:\semester3\Alpro 2 materi dan praktikum\laprak modul2>
PS D:\semester3\Alpro 2 materi dan praktikum\laprak modul2>

```

### **Deskripsi program :**

Program ini adalah aplikasi konsol yang dikembangkan menggunakan bahasa pemrograman Go untuk menghitung biaya pengiriman parcel berdasarkan berat. Program dapat menangani input berat dalam gram dan memberikan rincian biaya pengiriman yang terdiri dari biaya per kilogram dan biaya tambahan untuk sisa gram.

### **➤ Unguided 2c 2**

### **Sourch code :**

```
//fajar farizqi azmi
// 2311102192

package main

import "fmt"

func main() {
    var nam float64
    var nmk string

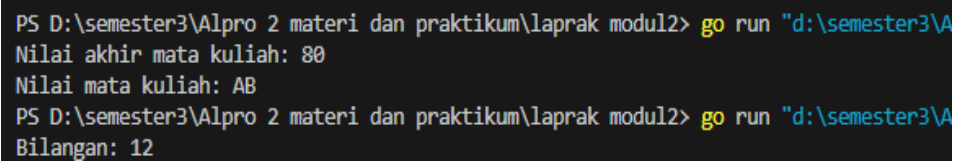
    fmt.Print("Nilai akhir mata kuliah: ")
    fmt.Scanln(&nam)

    if nam > 80 {
        nmk = "A"
    } else if nam > 72.5 {
        nmk = "AB"
    } else if nam > 65 {
        nmk = "B"
    } else if nam > 57.5 {
```

```
        nmk = "BC"
    } else if nam > 50 {
        nmk = "C"
    } else if nam > 40 {
        nmk = "D"
    } else {
        nmk = "E"
    }

    fmt.Println("Nilai mata kuliah:", nmk)
}
```

#### Screenshot output :



```
PS D:\semester3\Alpro 2 materi dan praktikum\laprak modul2> go run "d:\semester3\A
Nilai akhir mata kuliah: 80
Nilai mata kuliah: AB
PS D:\semester3\Alpro 2 materi dan praktikum\laprak modul2> go run "d:\semester3\A
Bilangan: 12
```

#### Deskripsi program :

Program ini adalah alat sederhana untuk mengonversi nilai akhir mata kuliah menjadi nilai huruf berdasarkan rentang nilai yang telah ditentukan.

Dengan menggunakan struktur kontrol if dan else if, program dapat menentukan nilai huruf yang tepat sesuai dengan input yang diberikan oleh pengguna.

#### ➤ Unguided 2c 3

**Source code :**

```
//fajar farizqi azmi
// 2311102192

package main

import (
    "fmt"
)

func main() {
    var b int

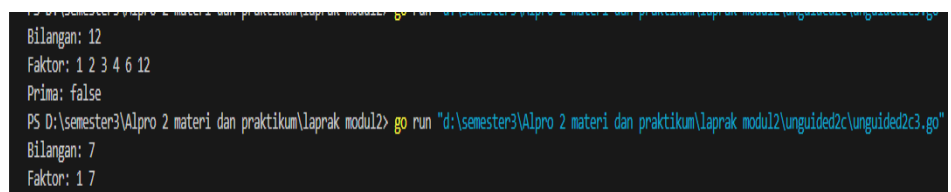
    fmt.Print("Bilangan: ")
    fmt.Scanln(&b)

    if b <= 1 {
        fmt.Println("Silakan masukkan bilangan bulat b > 1.")
        return
    }

    fmt.Print("Faktor: ")
    isPrima := true
    for i := 1; i <= b; i++ {
        if b%i == 0 {
            fmt.Print(i, " ")
            if i > 1 && i < b {
                isPrima = false
            }
        }
    }
}
```

```
}  
  
fmt.Println()  
  
fmt.Printf("Prima: %t\n", isPrima)  
}
```

### Screenshot output :



```
PS D:\semester3\Alpro 2 materi dan praktikum\laprak modul2> go run "d:\semester3\Alpro 2 materi dan praktikum\laprak modul2\unguided2c3.go"  
Billangan: 12  
Faktor: 1 2 3 4 6 12  
Prima: false  
PS D:\semester3\Alpro 2 materi dan praktikum\laprak modul2> go run "d:\semester3\Alpro 2 materi dan praktikum\laprak modul2\unguided2c3.go"  
Billangan: 7  
Faktor: 1 7
```

### Deskripsi program :

Program ini memberikan cara yang efektif untuk menghitung dan menampilkan faktor dari sebuah bilangan bulat serta menentukan apakah bilangan tersebut adalah bilangan prima.

Dengan menggunakan struktur kontrol seperti if dan for, program ini dapat memvalidasi input dan menghasilkan output yang sesuai.