

**LAPORAN PRAKTIKUM
ALGORITMA DAN PEMROGRAMAN 2**

**MODUL 2
REVIEW STRUKTUR KONTROL**



Disusun Oleh :

Zahra Tsuroyya Poetri / 2311102127

IF 11 - 05

Dosen Pengampu :

Arif Amrulloh

PROGRAM STUDI S1 TEKNIK INFORMATIKA

FAKULTAS INFORMATIKA

TELKOM UNIVERSITY PURWOKERTO

2024

I. DASAR TEORI

1. Pengenalan Golang

Golang adalah bahasa pemrograman sumber terbuka (opensource) yang berguna untuk memudahkan pembuatan perangkat lunak yang sederhana, andal, dan efisien. Golang (Go) disusun pada bulan September 2007 oleh Robert Griesemer, Rob Pike, dan Ken Thompson, semua kegiatannya dilakukan di Google, dan diumumkan pada bulan November 2009. Golang (Go) memiliki tujuan yaitu efisien dalam kompilasi dan eksekusi, dan efektif dalam menulis program.

Golang (Go) memiliki kemiripan dengan bahasa pemrograman C dalam bentuk penulisan dan pengkodeannya. Golang (Go) adalah proyek sumber terbuka (opensource), jadi kode sumber untuk kompiler, pustaka (library), dan alatnya tersedia secara gratis untuk siapa saja. Golang (Go) dapat berjalan pada Unix-like, Sistem Linux, FreeBSD, OpenBSD, Mac OS X, Plan 9, dan Microsoft Windows.

2. Kelebihan Golang

Golang memiliki kelebihan dibanding bahasa lainnya, beberapa di antaranya: Mendukung konkurensi di level bahasa dengan pengaplikasian cukup mudah Mendukung pemrosesan data dengan banyak prosesor dalam waktu yang bersamaan (parallel processing) Memiliki garbage collector Proses kompilasi sangat cepat Bukan bahasa pemrograman yang hirarkial, menjadikan developer tidak perlu ribet memikirkan segmen OOP-nya Package/modul yang disediakan terbilang lengkap. Karena bahasa ini open source, banyak sekali developer yang juga mengembangkan modul-modul lain yang bisa dimanfaatkan.

3. Command

Pengembangan aplikasi Golang tak jauh dari hal-hal yang berbau command line interface. Seperti kompilasi, testing, eksekusi program,

semua dilakukan lewat command line. Golang menyediakan command go untuk keperluan pengembangan aplikasi. Di bab ini kita akan belajar mengenai pemanfaatannya.

- Command go run

Command go run digunakan untuk eksekusi file program (file ber-ekstensi .go). Cara penggunaannya adalah dengan menuliskan command tersebut diikuti nama file.

- Command go test

Golang menyediakan package testing yang bisa dimanfaatkan untuk keperluan unit testing. File yang akan di-test harus ber-suffix _test.go .

- Command go build

Command ini digunakan untuk mengkompilasi file program. Sebenarnya ketika eksekusi program menggunakan go run , terjadi proses kompilasi juga, hanya saja file hasil kompilasi akan disimpan pada folder temporary untuk selanjutnya langsung dieksekusi. Berbeda dengan go build , command ini menghasilkan file executable pada folder yang sedang aktif.

- Command go install

Command go install memiliki fungsi yang sama dengan go build , hanya saja setelah proses kompilasi selesai, dilanjutkan ke proses instalasi program yang bersangkutan. Target eksekusi harus berupa folder proyek (bukan file .go), dan path folder tersebut dituliskan relatif terhadap \$GOPATH/src .

- Command go get

Command ini berbeda dengan command-command yang sudah dibahas di atas. go get digunakan untuk men-download package.

II. GUIDED

Guided 1

Soal Studi Case

Siswa kelas IPA di salah satu sekolah menengah atas di Indonesia sedang mengadakan praktikum kimia. Di setiap percobaan akan menggunakan 4 tabung reaksi, yang mana susunan warna cairan di setiap tabung akan menentukan hasil percobaan. Siswa diminta untuk mencatat hasil percobaan tersebut. Percobaan dikatakan berhasil apabila susunan warna zat cair pada gelas 1 hingga gelas 4 secara berturutan adalah 'merah', 'kuning', 'hijau', dan 'ungu' selama 5 kali percobaan berulang.

Buatlah sebuah program yang menerima input berupa warna dari ke 4 gelas reaksi sebanyak 5 kali percobaan. Kemudian program akan menampilkan true apabila urutan warna sesuai dengan informasi yang diberikan pada paragraf sebelumnya, dan false untuk warna sesuai dengan informasi yang iversity urutan warna lainnya.

Sourcecode

```
package main

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

func main() {
    // Urutan warna yang benar
    correctOrder := []string{"merah", "kuning", "hijau",
"ungu"}

    // Membaca input untuk 5 percobaan
    reader := bufio.NewReader(os.Stdin)
    success := true

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

        // Membaca input dari pengguna
        input, _ := reader.ReadString('\n')
        input = strings.TrimSpace(input)

        // Memisahkan input berdasarkan spasi
        colors := strings.Split(input, " ")

        // Mengecek apakah urutan warna sesuai
        for j := 0; j < 4; j++ {
            if colors[j] != correctOrder[j] {
                success = false
            }
        }
    }
}
```

```

        break
    }
}

// Jika ada percobaan yang tidak sesuai, keluar
dari loop
if !success {
    break
}

// Menampilkan hasil
if success {
    fmt.Println("BERHASIL : true")
} else {
    fmt.Println("BERHASIL : false")
}
}

```

Screenshoot Output

```

PS C:\berkas aya\kuliah\sem 3\Praktikum ALPRO 2\Minggu 2> go run "c:\berkas aya\kuliah\sem 3\Praktikum ALPRO 2\Minggu 2\guided1.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 C:\berkas aya\kuliah\sem 3\Praktikum ALPRO 2\Minggu 2> go run "c:\berkas aya\kuliah\sem 3\Praktikum ALPRO 2\Minggu 2\guided1.go"
Percobaan 1: merah kuning hijau ungu
Percobaan 2: merah kuning hijau ungu
Percobaan 3: merah kuning hijau ungu
Percobaan 4: ungu kuning hijau merah ungu
BERHASIL : false
PS C:\berkas aya\kuliah\sem 3\Praktikum ALPRO 2\Minggu 2>

```

Deskripsi Program

Program tersebut merupakan program praktikum kimia yang meminta pengguna menginputkan 5 kali percobaan dengan warna cairan “merah”, “kuning”, “hijau”, “ungu”. Pada percobaan pengguna menginputkan warna cairan tersebut secara berulang dan sama urutannya. Sehingga ketika pengguna menginputkan warna dengan bena, maka program akan

mengeluarkan output “BERHASIL: true”. Dan output akan berubah menjadi “BERHASIL: false” ketika pengguna salah memasukkan inputan atau tidak sesuai urutan.

Algoritma Program dan Cara Kerja Program

1. Menginisialisasi urutan warna yang benar, yaitu “merah”, “kuning”, “hijau”, “ungu”. Urutan warna yang benar disimpan menggunakan `correctOrder`
2. Looping sebanyak 5 kali percobaan menggunakan loop `for`
3. Membaca inputan pengguna menggunakan `bufio.NewReader(os.Stdin)` untuk membaca input berupa string dari pengguna. Fungsi `strings.TrimSpace()` menghapus spasi yang tidak diperlukan di awal dan akhir input.
4. Memisahkan input pengguna berdasarkan spasi dan urutan warna menggunakan `strings.Split(input, “ ”)` dan tersimpan ke dalam `colors`.
5. Membandingkan inputan dengan benar menggunakan loop `for` yang berjalan untuk setiap warna yang dimasukkan oleh pengguna.
6. Jika ada percobaan yang gagal atau input tidak sesuai urutan, variabel `success` akan bernilai `false`. Dan loop `for` akan menghentikan percobaan.
7. Setelah selesai melakukan 5 percobaan hasil akhir menentukan sesuai dengan variabel `success`. Jika nilai `true`, maka program akan menampilkan “BERHASIL: true”. Jika `false`, program menampilkan “BERHASIL: false”.

Guided 2

Sourcecode

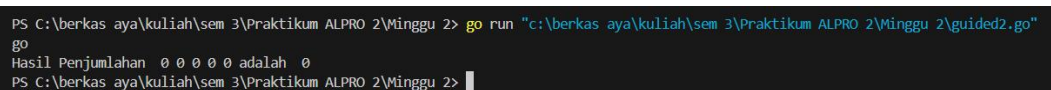
```
package main

import (
    "fmt"
)

func main(){
    var a, b, c, d, e int
    var hasil int
    fmt.Scanln(&a, &b, &c, &d, &e)

    hasil = a+b+c+d+e
    fmt.Println("Hasil Penjumlahan ",a,b,c,d,e, "adalah
",hasil)
}
```

Screenshoot Output



```
PS C:\berkas aya\kuliah\sem 3\Praktikum ALPRO 2\Minggu 2> go run "c:\berkas aya\kuliah\sem 3\Praktikum ALPRO 2\Minggu 2\guided2.go"
go
Hasil Penjumlahan  0 0 0 0 0 adalah 0
PS C:\berkas aya\kuliah\sem 3\Praktikum ALPRO 2\Minggu 2> █
```

Deskripsi Program

Program tersebut merupakan program untuk memberikan hasil penjumlahan bilangan bulat dari variabel a, b, c, d, dan e. Program ini langsung menampilkan output hasil penjumlahan.

Algoritma dan Cara Kerja Program

1. Menginisialisasi variabel bilangan bulat a, b, c, d, dan e untuk menyimpan angka. Program mendeklarasikan variabel hasil untuk menyimpan hasil penjumlahan dari lima angka.
2. Program membaca inputan pengguna menggunakan `fmt.ScanIn(&a, &b, &c, &d, &e)` untuk menerima bilangan bulat dari pengguna.
3. Setelah itu program menjumlahkan kelima bilangan tersebut dengan melakukan operasi $a + b + c + d + e$, dan menyimpan ke variabel hasil.
4. Kemudian program akan menampilkan hasil penjumlahan dengan output : “Hasil Penjumlahan a b c d e adalah hasil”.

Guided 3

Soal Studi Case

Diberikan sebuah nilai akhir mata kuliah (NAM) [0..100] dan standar penilaian nilai mata kuliah (NMK) sebagai berikut:

NAM	NMK
$NAM > 80$	A
$72.5 < NAM \leq 80$	AB

$65 < NAM \leq 72.5$	B
$57.5 < NAM \leq 65$	BC
$50 < NAM \leq 57.5$	C
$40 < NAM \leq 50$	D
$NAM \leq 40$	E

Sourcecode

```
package main

import "fmt"

func main() {
    var nam float32
    var nmk string

    // Meminta input nilai
    fmt.Print("Masukkan nilai : ")
    fmt.Scan(&nam)

    // Logika penentuan nilai huruf berdasarkan nilai
    numerik
    if nam > 80 {
        nmk = "A"
    } else if nam > 72.5 {
        nmk = "B"
    } else if nam > 65 {
        nmk = "C"
    } else if nam > 50 {
        nmk = "D"
    } else if nam > 40 {
        nmk = "E"
    } else {
        nmk = "F"
    }

    // Menampilkan hasil
    fmt.Printf("Nilai Indeks untuk nilai %.2f
    adalah %s\n", nam, nmk)
}
```

Screenshoot Output

```
PS C:\berkas aya\kuliah\sem 3\Praktikum ALPRO 2\Minggu 2> go run "c:\berkas aya\kuliah\sem 3\Praktikum ALPRO 2\Minggu 2\guided3.go"
Masukkan nilai : 80
Nilai Indeks untuk nilai 80.00 adalah B
PS C:\berkas aya\kuliah\sem 3\Praktikum ALPRO 2\Minggu 2> go run "c:\berkas aya\kuliah\sem 3\Praktikum ALPRO 2\Minggu 2\guided3.go"
Masukkan nilai : 81
Nilai Indeks untuk nilai 81.00 adalah A
PS C:\berkas aya\kuliah\sem 3\Praktikum ALPRO 2\Minggu 2> go run "c:\berkas aya\kuliah\sem 3\Praktikum ALPRO 2\Minggu 2\guided3.go"
Masukkan nilai : 30
Nilai Indeks untuk nilai 30.00 adalah F
PS C:\berkas aya\kuliah\sem 3\Praktikum ALPRO 2\Minggu 2> █
```

Deskripsi Program

Program tersebut merupakan program untuk mengecek hasil nilai akhir mata kuliah dengan rentang nilai 0-100. Pengguna dapat memasukkan nilai dan akan menghasilkan output sesuai tabel numerik yang telah ditentukan.

Algoritma Program dan Cara Kerja Program:

1. Program mendeklarasikan variabel `nam` bertipe `float32` untuk menyimpan input nilai numerik dari pengguna. Dan mendeklarasikan variabel `nmk` bertipe `string` untuk menyimpan nilai huruf atau indeks yang akan ditentukan berdasarkan numerik.
2. Program meminta pengguna untuk menginputkan nilai menggunakan `fmt.Print("Masukkan nilai : ")`. Dan program menggunakan `fmt.Scan(&nam)` untuk membacainputan nilai dari pengguna lalu disimpan dalam variabel `nam`.
3. Program melakukan pemeriksaan kondisi menggunakan `if-else` untuk menentukan nilai huruf (`nmk`) berdasarkan nilai numerik (`nam`) yang telah diinputkan.
4. Menampilkan hasil menggunakan `fmt.Printf` yang mencakup nilai numerik dan nilai indeks yang ditentukan.

III. UNGUIDED

Unguided 1

Soal Studi Case

Setiap hari Pak Andi membawa banyak barang belanjaan dari pasar dengan mengendarai sepeda motor. Barang belanjaan tersebut dibawa dalam kantong terpal di kiri-kanan motor. Sepeda motor tidak akan oleng jika selisih berat barang di kedua kantong sisi tidak lebih dari 9kg.

Buatlah program Pak Andi yang menerima input dua buah bilangan real positif yang menyatakan berat total masing-masing isi kantong terpal. Program akan terus meminta input bilangan tersebut hingga salah satu kantong terpal berisi 9Kg atau lebih

Perhatikan contoh sesi interaksi program seperti di bawah ini

Masukkan berat belanjaan di kedua kantong: 5.5 1.0
Masukkan berat belanjaan di kedua kantong: 7.1 8.5
Masukkan berat belanjaan di kedua kantong: 2 6
Masukkan berat belanjaan di kedua kantong: 9 5.8
Proses selesai

Pada modifikasi program tersebut, program akan menampilkan true jika selisih kedua isi kantong lebih dari atau sama dengan 9 kg. Program berhenti memproses apabila total berat isi kedua kantong melebihi 150 kg atau salah satu kantong beratnya negatif.

Perhatikan contoh sesi interaksi program seperti di bawah ini

```
Masukan berat belanjaan di kedua kantong: 5 10
Sepeda motor pak Andi akan oleng: false
Masukan berat belanjaan di kedua kantong: 55.6 70.2
Sepeda motor pak Andi akan oleng: true
Masukan berat belanjaan di kedua kantong: 72.3 66.9
Sepeda motor pak Andi akan oleng: false
Masukan berat belanjaan di kedua kantong: 59.5 98.7
Proses selesai.
```

Sourcecode

```
package main

import (
    "fmt"
)

func main() {
    var beratK1, beratK2 float64

    // Loop untuk meminta input hingga salah satu kantong
    berisi 9Kg atau lebih
    for {
        // Meminta input berat belanjaan untuk kedua
        kantong
        fmt.Print("Masukkan berat belanjaan di kedua
        kantong: ")
        fmt.Scanln(&beratK1, &beratK2)

        // Mengecek apakah salah satu kantong berisi 9Kg
        atau lebih
        if beratK1 >= 9 || beratK2 >= 9 {
```

```
        break
    }
}

// Menampilkan pesan proses selesai
fmt.Println("Proses selesai")
}
```

Sourcecode dimodifikasi

```
package main

import (
    "fmt"
)

func main() {
    var beratK1, beratK2 float64

    // Loop untuk meminta input berat belanjaan
    for {
        // Meminta input berat belanjaan untuk kedua
        kantong
        fmt.Print("Masukan berat belanjaan di kedua
        kantong: ")
        fmt.Scanln(&beratK1, &beratK2)

        // Mengecek jika salah satu kantong berisi angka
        negatif atau total berat > 150kg
        if beratK1 < 0 || beratK2 < 0 ||
        beratK1+beratK2 > 150 {
            break
        }

        // Menghitung selisih berat kedua kantong
    }
}
```

```

var selisih float64
if beratK1 > beratK2 {
    selisih = beratK1 - beratK2
} else {
    selisih = beratK2 - beratK1
}

// Mengecek apakah selisih 9 kg atau lebih (motor
pak Andi akan oleng)
oleng := selisih >= 9

// Menampilkan apakah motor oleng atau tidak
fmt.Printf("Sepeda motor pak Andi akan
oleng: %t\n", oleng)
}

// Menampilkan pesan proses selesai
fmt.Println("Proses selesai.")
}

```

Screenshoot Output

```

PS C:\berkas aya\kuliah\sem 3\Praktikum ALPRO 2\Minggu 2> go run "c:\berkas aya\kuliah\sem 3\Praktikum ALPRO 2\Minggu 2\unguided1.go"
Masukkan berat belanjaan di kedua kantong: 5.5 1.0
Masukkan berat belanjaan di kedua kantong: 7.1 8.5
Masukkan berat belanjaan di kedua kantong: 2 6
Masukkan berat belanjaan di kedua kantong: 9 5.8
Proses selesai
PS C:\berkas aya\kuliah\sem 3\Praktikum ALPRO 2\Minggu 2> go run "c:\berkas aya\kuliah\sem 3\Praktikum ALPRO 2\Minggu 2\unguided1modif.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 C:\berkas aya\kuliah\sem 3\Praktikum ALPRO 2\Minggu 2>

```

Deskripsi Program

Program tersebut merupakan program untuk menghitung selisih berat belanjaan dua sisi kantong pada motor pak Andi yang tidak lebih dari 9kg. Program akan meminta pengguna menginputkan dua sisi kantong tersebut

hingga berat selisih sisi kantong menyentuh 9kg atau lebih, kemudian program akan berhenti.

Lalu pada program yang dimodifikasi program akan menampilkan pesan true ketika selisih kedua sisi kantong mencapai 9kg atau lebih, dan program akan menampilkan pesan false ketika selisih kedua kantong tidak mencapai 9kg. Program akan berhenti apabila angka negatif dan total berat kedua sisi kantong lebih dari 150kg.

Algoritma Program dan Cara Kerja Program

1. Program mendeklarasikan dua variabel beratK1 dan beratK2 bertipe float64 untuk menyimpan berat belanjaan di kedua kantong
2. Program menggunakan loop for untuk meminta input dari pengguna sampai kondisi terpenuhi.
3. Program menampilkan pesan meminta pengguna untuk menginputkan berat dua kantong belanjaan menggunakan `fmt.ScanIn(&beratK1, &beratK2)`.
4. Program memeriksa salah satu berat apakah negatif atau total beratK1 + beratK2 lebih dari 150kg. Jika kondisi tersebut terpenuhi maka program selesai.
5. Program mendeklarasikan variabel selisih bertipe float64.
6. Mengecek selisih apakah lebih besar atau sama dengan 9kg, yang menunjukkan bahwa sepeda motor pak Andi akan oleng.
7. Menampilkan hasil program menggunakan `fmt.Printf` untuk menampilkan pesan : “Sepeda motor pak Andi akan oleng: [true/false]”.
8. Hasil akhir menampilkan pesan proses selesai.

Unguided 2

Soal Studi Case

Diberikan sebuah persamaan sebagai berikut ini.

$$f(k) = \frac{(4k + 2)^2}{(4k + 1)(4k + 3)}$$

Buatlah sebuah program yang menerima Input sebuah bilangan sebagai K, kemudian menghitung dan menampilkan nilai f(K) sesuai persamaan di atas.

Perhatikan contoh sesi interaksi program seperti di bawah ini

```
Nilai K 100
Nilai f(K) 1.0000061880
```

Sourcecode

```
package main

import (
    "fmt"
)

func main() {
    // Meminta input nilai K
    var k float64
    fmt.Print("Nilai K = ")
    fmt.Scanln(&k)

    // Menghitung bagian pembilang dan bagian penyebut
    dari persamaan
    pembilang := (4*k + 2) * (4*k + 2) // Menghitung
    (4k+2)^2
```

```

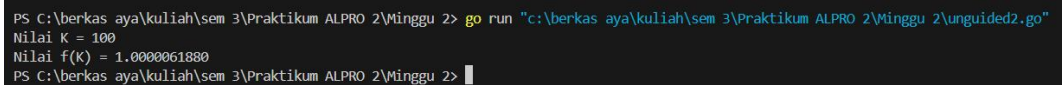
    penyebut := (4*k + 1) * (4*k + 3) // Menghitung
(4k+1) (4k+3)

    fk := pembilang / penyebut // Menghitung f(K)

    // Menampilkan hasil nilai K
    fmt.Printf("Nilai f(K) = %.10f\n", fk)
}

```

Screenshoot Output



```

PS C:\berkas aya\kuliah\sem 3\Praktikum ALPRO 2\Minggu 2> go run "c:\berkas aya\kuliah\sem 3\Praktikum ALPRO 2\Minggu 2\unguided2.go"
Nilai K = 100
Nilai f(K) = 1.0000061880
PS C:\berkas aya\kuliah\sem 3\Praktikum ALPRO 2\Minggu 2>

```

Deskripsi Program

Program tersebut merupakan program untuk menjalankan sebuah persamaan dan menghasilkan nilai K. Pengguna dapat menginputkan nilai K lalu nilai persamaan akan diproses oleh program dan menghasilkan output setelahnya.

Algoritma Program dan Cara Kerjanya

1. Program mendeklarasikan variabel k bertipe float64 untuk menyimpan input nilai K dari pengguna.
2. Program akan menampilkan pesan untuk pengguna memberikan inputan dan dibaca menggunakan `fmt.ScanIn(&k)`.
3. Program menghitung nilai pembilang dengan rumus yang telah ditentukan dan menyimpannya dalam variabel pembilang.
4. Program juga menghitung nilai penyebut dengan rumus yang telah ditentukan dan menyimpannya dalam variabel penyebut.
5. Hasil program dihitung dengan membagi pembilang dan penyebut, dan hasilnya disimpan dalam variabel `fk`.
6. Menampilkan hasil program yang telah dihitung menggunakan `fmt.printf`.

Unguided 3

Soal Studi Case

PT POS membutuhkan aplikasi perhitungan biaya kirim berdasarkan berat parcel. Maka, **buatlah program BlayaPos untuk menghitung biaya pengiriman tersebut dengan ketentuan sebagai berikut!**

Dari berat parcel (dalam gram), harus dihitung total berat dalam kg dan sisanya (dalam gram). Biaya jasa pengiriman adalah Rp. 10.000,- per kg. Jika sisa berat tidak kurang dari 500 gram, maka tambahan biaya kirim hanya Rp. 5,- per gram saja. Tetapi jika kurang dari 500 gram, maka tambahan biaya akan dibebankan sebesar Rp. 15,- per gram. Sisa berat (yang kurang dari 1kg) digratiskan biayanya apabila total berat ternyata lebih dari 10kg.

Perhatikan contoh sesi interaksi program seperti di bawah ini

1	Contoh #1 Berat parcel (gram) 8500 Detail berat: 8 kg 500 gr Detail biaya: Rp. 80000 Rp. 2500 Total biaya Rp 82500
2	Contoh #2 Berat parcel (gram) 9250 Detail berat: 9 kg 250 gr Detail biaya Rp. 90000 + Rp. 3750
3	Contoh #3

	Berat parsel (gram): 11750
	Detail berat: 11 kg + 750 gr
	Detail biaya: Rp. 110000 + Rp. 3750
	Total biaya: Rp. 110000

Sourcecode

```
package main

import (
    "fmt"
)

func main() {
    var beratParsel int
    var kg, sisaBerat int
    var biayaPerKg, biayaSisa int

    // Meminta input berat parsel dalam gram
    fmt.Print("Berat parsel (gram): ")
    fmt.Scanln(&beratParsel)

    // Menghitung berat dalam kg dan sisa berat
    kg = beratParsel / 1000
    sisaBerat = beratParsel % 1000

    // Menghitung biaya dasar berdasarkan per kg
    biayaPerKg = kg * 10000

    // Menghitung biaya tambahan berdasarkan sisa berat
    if kg > 10 {
        biayaSisa = 0 // Jika berat lebih dari 10 kg,
        sisa berat digratiskan
    }
```

```

    } else if sisaBerat >= 500 {
        biayaSisa = sisaBerat * 5 // Jika sisa >= 500
gram, biaya tambahan Rp. 5 per gram
    } else {
        biayaSisa = sisaBerat * 15 // Jika sisa < 500
gram, biaya tambahan Rp. 15 per gram
    }

    // Menampilkan detail berat
    fmt.Printf("Detail berat: %d kg %d gr\n", kg,
sisaBerat)

    // Menampilkan detail biaya
    fmt.Printf("Detail biaya: Rp. %d + Rp. %d\n",
biayaPerKg, biayaSisa)

    // Menampilkan total biaya
    fmt.Printf("Total biaya: Rp. %d\n",
biayaPerKg+biayaSisa)
}

```

Screenshoot Output

```

gu 2> go run "c:\berkas aya\kuliah\sem 3\Praktikum ALPRO 2\Minggu 2\unguided3.go"
Berat parcel (gram): 8500
Detail berat: 8 kg 500 gr
Detail biaya: Rp. 80000 + Rp. 2500
Total biaya: Rp. 82500
PS C:\berkas aya\kuliah\sem 3\Praktikum ALPRO 2\Minggu 2> go run "c:\berkas aya\kuliah\sem 3\Praktikum ALPRO 2\Minggu 2\unguided3.go"
Berat parcel (gram): 9250
Detail berat: 9 kg 250 gr
Detail biaya: Rp. 90000 + Rp. 3750
Total biaya: Rp. 93750
PS C:\berkas aya\kuliah\sem 3\Praktikum ALPRO 2\Minggu 2> go run "c:\berkas aya\kuliah\sem 3\Praktikum ALPRO 2\Minggu 2\unguided3.go"
Berat parcel (gram): 11750
Detail berat: 11 kg 750 gr
Detail biaya: Rp. 110000 + Rp. 0
Total biaya: Rp. 110000
PS C:\berkas aya\kuliah\sem 3\Praktikum ALPRO 2\Minggu 2>

```

Deskripsi Program

Program tersebut merupakan program untuk menghitung biaya pos untuk sebuah berat parcel dalam pengiriman. Berat akan dihitung per kg dan per

sis gram dengan ketentuan biaya yang telah ditentukan. Ketika pengguna memasukkan inputan, program akan menghitung dan menghasilkan output berat parcel diikuti dengan detail berat, detail biaya, dan total biaya.

Algoritma Program dan cara Kerja Program

1. Program mendeklarasikan variabel beratParcel, kg, sisaBerat, biayaPerKg, dan biayaSisa untuk menyimpan nilai yang diperlukan.
2. Program meminta pengguna untuk menginputkan dengan menampilkan pesan : “Berat parcel (gram): “. dan program akan membaca inputan menggunakan `fmt.ScanIn(&beratParcel)`.
3. Program menghitung berat dalam kilogram dengan membagi beratParcel dengan 1000 lalu menyimpannya dalam variabel kg.
4. Program juga menghitung sisa berat dalam berat gram dengan menggunakan modulus 1000, lalu menyimpan hasil di dalam biayaPerKg.
5. Porgram menghitung biaya tambahan berdasarkan sisa berat yang telah ditentukan.
6. Program menampilkan detail berat dengan format Detail berat: Rp. %d + Rp. %d\n menggunakan `fmt.Printf`.
7. Program menampilkan detail biaya dalam format Detail biaya: Rp. %d + Rp. %d\n menggunakan `fmt.Printf`.
8. Program menampilkan total biaya dengan menjumlahkan biayarPerKg dan biayaSisa dalam format Total biaya: Rp. %d\n.

Unguided 4

Soal Studi Case

Diberikan sebuah nilai akhir mata kuliah (NAM) [0..100] dan standar penilaian nilai mata kuliah (NMK) sebagai berikut:

NAM	NMK
NAM > 80	A
72.5 < NAM <= 80	AB

65 < NAM <= 72.5	B
57.5 < NAM <= 65	BC
50 < NAM <= 57.5	C
40 < NAM <= 50	D
NAM <= 40	E

Program berikut menerima input sebuah bilangan riil yang menyatakan NAM. Program menghitung NMK dan menampilkannya.

```
package main
import "fmt"
func main() {
    var nam float64
    var nmk string
    fmt.Print("Nilai akhir mata kuliah: ")
    fmt.Scanln(&nam)
    if nam > 80 {
        nam = "A"
    }
    if nam > 72.5 {
        nam = "AB"
    }
    if nam > 65 {
        nam = "B"
    }
    if nam > 57.5 {
```

```
nam = "BC"
}
if nam > 50 {
nam = "C"
}
if nam > 40 {
nam = "D"
}
else if nam <= 40 {
nam = "E"
}
fmt.Println("Nilai mata kuliah: ", nmk)
}
```

Jawablah pertanyaan-pertanyaan berikut:

- a. Jika nam diberikan adalah 80.1, apa keluaran dari program tersebut?
Apakah eksekusi program tersebut sesuai spesifikasi soal?
- b. Apa saja kesalahan dari program tersebut? Mengapa demikian?
Jelaskan alur program seharusnya
- c. Perbaiki program tersebut! Ujilah dengan masukan: 93.5; 70.6; dan 49.5.
Seharusnya keluaran yang diperoleh adalah 'A', 'B', dan 'D'.

Sourcecode

```
package main
import "fmt"
func main() {
    var nam float64
    var nmk string
    fmt.Print("Nilai akhir mata kuliah: ")
    fmt.Scanln(&nam)
    if nam > 80 {
```



```
    nam = "A"
}
if nam > 72.5 {
    nam = "AB"
}
if nam > 65 {
    nam = "B"
}
if nam > 57.5 {
    nam = "BC"
}
if nam > 50 {
    nam = "C"
}
if nam > 40 {
    nam = "D"
}
else if nam <= 40 {
    nam = "E"
}
fmt.Println("Nilai mata kuliah: ", nmk)
}
```

Sourcecode diperbaiki

```
package main

import "fmt"

func main() {
    var nam float64
    var nmk string

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

```
// Memeriksa nilai NAM dan menentukan NMK
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"
}

// Menampilkan hasil NMK
fmt.Println("Nilai mata kuliah:", nmk)
}
```

Screenshoot Output

```
PS C:\berkas aya\kuliah\sem 3\Praktikum ALPRO 2\Minggu 2> go run "c:\berkas aya\kuliah\sem 3\Praktikum ALPRO 2\Minggu 2\unguided4.go"
# command-line-arguments
.\unguided4.go:27:2: syntax error: unexpected else, expected }
PS C:\berkas aya\kuliah\sem 3\Praktikum ALPRO 2\Minggu 2>
```

Sesudah dimodifikasi

```
PS C:\berkas aya\kuliah\sem 3\Praktikum ALPRO 2\Minggu 2> go run "c:\berkas aya\kuliah\sem 3\Praktikum ALPRO 2\Minggu 2\unguided4modif.go"
Nilai akhir mata kuliah: 93.5
Nilai mata kuliah: A
PS C:\berkas aya\kuliah\sem 3\Praktikum ALPRO 2\Minggu 2> go run "c:\berkas aya\kuliah\sem 3\Praktikum ALPRO 2\Minggu 2\unguided4modif.go"
Nilai akhir mata kuliah: 70.6
Nilai mata kuliah: B
PS C:\berkas aya\kuliah\sem 3\Praktikum ALPRO 2\Minggu 2> go run "c:\berkas aya\kuliah\sem 3\Praktikum ALPRO 2\Minggu 2\unguided4modif.go"
Nilai akhir mata kuliah: 49.5
Nilai mata kuliah: D
PS C:\berkas aya\kuliah\sem 3\Praktikum ALPRO 2\Minggu 2> |
```

Deskripsi Program

Program tersebut merupakan program untuk mengisi nilai akhir kuliah dengan standar penilaian nam(numerik) dan nmk(nilai huruf). Pengguna

dapat memberikan inputan nilai akhir kuliah lalu program akan menghasilkan output nilai indeks sesuai ketentuan.

Algoritma Program dan Cara Kerja Program

1. Program mendeklarasikan variabel `nam` untuk menyimpan nilai akhir yang dimasukkan oleh pengguna dan variabel `nmk` untuk menyimpan nilai huruf yang akan ditentukan.
2. Program menampilkan pesan untuk pengguna dapat menginputkan nilai dan program membaca input dari pengguna menggunakan `fmt.ScanIn(&nam)`.
3. Program menggunakan struktur `if-else` untuk memeriksa nilai `nam` dan menentukan nilai huruf (`nmk`) berdasarkan ketentuan.
4. Program menampilkan hasil nilai mata kuliah sesuai inputan dengan menggunakan `fmt.Println`.

Jawaban

- a. Jika input adalah 80.1, program akan menghasilkan error karena penggunaan variabel yang terbalik. Keluaran program tidak sesuai dengan spesifikasi soal karena program seharusnya memberikan nilai huruf menggunakan string sesuai dengan rentang nilai, tetapi program gagal karena kesalahan tipe data. Sesuai spesifikasi soal ketika sudah diperbaiki.
- b. Kesalahan :
 - Salah dalam variabel: program menggunakan variabel `nam` untuk mengisi nilai huruf, padahal `nam` adalah variabel yang menyimpan nilai numerik.
 - Program tidak menggunakan `else if` dengan benar, sehingga kondisi tidak terpenuhi secara keseluruhan.

- c. (Perbaikan ada dalam source code dan hasil screenshoot)

Unguided 5

Soal Studi Case

Sebuah bilangan bulat b memiliki faktor bilangan $f > 0$ jika f habis membagi b . Contoh: 2 merupakan faktor dari bilangan 6 karena 6 habis dibagi 2.

Buatlah program yang menerima input sebuah bilangan bulat b dan $b > 1$. Program harus dapat mencari dan menampilkan semua faktor dari bilangan tersebut!

Perhatikan contoh sesi interaksi program seperti di bawah ini

Bilangan: 12	Bilangan: 7
Faktor: 12346 12	Faktor: 17

Bilangan bulat $b > 0$ merupakan bilangan prima p jika dan hanya jika memiliki persis dua faktor bilangan saja, yaitu 1 dan dirinya sendiri. Lanjutkan program sebelumnya. Setelah menerima masukan sebuah bilangan bulat $b > 0$.

Program tersebut mencari dan menampilkan semua faktor bilangan tersebut. Kemudian, program menentukan apakah b merupakan bilangan prima.

Perhatikan contoh sesi Interaksi program seperti di bawah ini

Bilangan: 12	Bilangan: 7
Faktor: 12346 12	Faktor: 17

Prima: false	Prima: true
--------------	-------------

Sourcecode

```
package main

import (
    "fmt"
)

func main() {
    var b int

    // Meminta input bilangan bulat b > 1
    fmt.Print("Bilangan: ")
    fmt.Scanln(&b)

    // Memastikan input valid
    if b <= 1 {
        fmt.Println("Bilangan harus lebih dari 1.")
        return
    }

    // Menampilkan faktor-faktor bilangan
    fmt.Print("Faktor: ")
    for i := 1; i <= b; i++ {
        if b%i == 0 {
            fmt.Print(i)
        }
    }
    fmt.Println() // Untuk memberikan newline di akhir
    output
}
```

Setelah dimodifikasi

```
package main

import (
    "fmt"
)

func main() {
    var b int

    // Meminta input bilangan bulat b > 1
    fmt.Print("Bilangan: ")
    fmt.Scanln(&b)

    // Memastikan input valid
    if b <= 1 {
        fmt.Println("Bilangan harus lebih dari 1.")
        return
    }

    // Array untuk menyimpan faktor
    var faktor [100]int
    hasilFaktor := 0

    // Mencari faktor bilangan
    for i := 1; i <= b; i++ {
        if b%i == 0 {
            faktor[hasilFaktor] = i // Memasukkan faktor
            ke dalam array
            jumlahFaktor++          // Menambah jumlah
            faktor
        }
    }

    // Menampilkan semua faktor
```

```

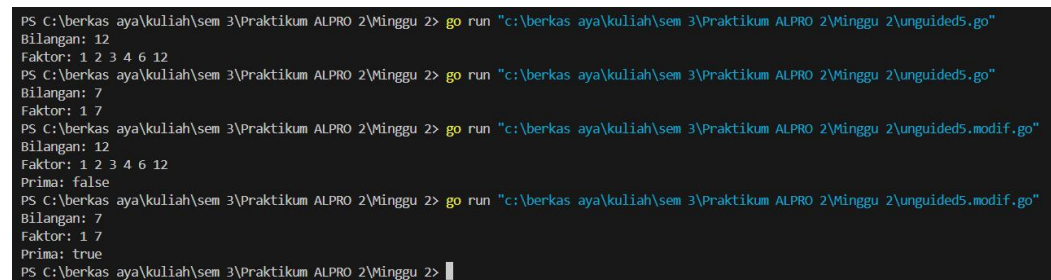
    fmt.Print("Faktor: ")
    for i := 0; i < hasilFaktor; i++ {
        fmt.Print(faktor[i], " ")
    }

    // Menentukan apakah bilangan tersebut prima
    prima := hasilFaktor == 2 // Bilangan prima hanya
memiliki 2 faktor

    // Menampilkan hasil apakah bilangan tersebut prima
atau tidak
    fmt.Printf("\nPrima: %t\n", prima)
}

```

Screenshoot Output



```

PS C:\berkas aya\kuliah\sem 3\Praktikum ALPRO 2\Minggu 2> go run "c:\berkas aya\kuliah\sem 3\Praktikum ALPRO 2\Minggu 2\unguided5.go"
Bilangan: 12
Faktor: 1 2 3 4 6 12
PS C:\berkas aya\kuliah\sem 3\Praktikum ALPRO 2\Minggu 2> go run "c:\berkas aya\kuliah\sem 3\Praktikum ALPRO 2\Minggu 2\unguided5.go"
Bilangan: 7
Faktor: 1 7
PS C:\berkas aya\kuliah\sem 3\Praktikum ALPRO 2\Minggu 2> go run "c:\berkas aya\kuliah\sem 3\Praktikum ALPRO 2\Minggu 2\unguided5.modif.go"
Bilangan: 12
Faktor: 1 2 3 4 6 12
Prima: false
PS C:\berkas aya\kuliah\sem 3\Praktikum ALPRO 2\Minggu 2> go run "c:\berkas aya\kuliah\sem 3\Praktikum ALPRO 2\Minggu 2\unguided5.modif.go"
Bilangan: 7
Faktor: 1 7
Prima: true
PS C:\berkas aya\kuliah\sem 3\Praktikum ALPRO 2\Minggu 2>

```

Deskripsi Program

Program tersebut merupakan program pemfaktora dari bilangan bulat dan bilang prima yang dapat diinputkan oleh pengguna. Program akan memproses ketika pengguna menginputkan bilangan, setelah itu jika bilangan tersebut merupakan bilangan prima maka akan true, jika bukan maka akan false.

Algoritma Program dan Cara kerja Program

1. Program mendeklarasikan variabel untuk menyimpan bilangan bulat yang diinputkan oleh pengguna.

2. Program menginisialisai array faktor untuk menyimpan faktor-faktor dari bilangan dan variabel hasilFaktor untuk menghitung jumlah faktor.
3. Program meminta inputan kepada pengguna dan membaca inputan dari pengguna menggunakan `fmt.ScanIn(&b)`.
4. Program memastikan inputan valid dengan b apakah kurang dari atau sama dengan 1.
5. Program menggunakan loop for dari 1 hingga b untuk mencari faktor.
6. Program memeriksa apakah b dapat dibagi dengan i tanpa sisa (`b % i == 0`). Jika ya, faktor tersebut akan disimpan dalam array faktor dan hasilFaktor.
7. Program menampilkan semua faktor yang ditemukan dengan menggunakan loop for untuk mencetak elemen dalam array faktor.
8. Program menampilkan hasil apakah bilangan tersebut adalah bilangan prima dengan menggunakan format Prima: %t.

Daftar Pustaka

- [1] M.Kom., Dr. Harja Santanapurba,, Dr. R. Ati Sukmawati, M.Kom, dan Ahmad Faisal. (2022). *DASAR-DASAR BAHASA PEMROGRAMAN GOLANG* Banjarmasin: Universitas Lambung Mangkurat.
<http://surl.li/lbiqzt>
- [2] Prayogo, Noval Agung. *Dasar Pemrograman Golang*.
<http://surl.li/euwyrh>