

**LAPORAN PRAKTIKUM
ALGORITMA DAN PEMROGRAMAN 2**

**MODUL II
REVIEW STRUKTUR KONTROL**



Oleh:

FEROS PEDROSA VALENTINO

103112400055

IF-12-01

**S1 TEKNIK INFORMATIKA
TELKOM UNIVERSITY PURWOKERTO**

2025

I. DASAR TEORI

Program utama pada Go pasti selalu mempunyai komponen:

1. package main
berfungsi sebagai penanda bahwa file tersebut berisi program utama.
2. func main()
terdapat kode utama dari sebuah program Go.

GoLang hanya mempunyai kata kunci for untuk semua jenis perulangan yang dipelajari dalam notasi algoritma. Dalam konsep pemrograman terstruktur, setiap rancangan algoritma harus memenuhi syarat satu pintu masuk dan satu pintu keluar. Dalam Go, kita menggunakan loop while untuk mengeksekusi blok kode hingga kondisi tertentu terpenuhi. Go tidak memiliki kata kunci khusus untuk while loop. Namun, kita dapat menggunakan forloop untuk menjalankan fungsi while loop. Bentuk repeat-until di perulangan dilakukan terus menerus sampai kondisi keluar terpenuhi. Artinya selama kondisi belum terpenuhi (salah) maka perulangan akan terus dilakukan. Pada saat keluar dari loop maka nilai kondisi pasti benar.

GoLang mendukung dua percabangan yaitu if-else dan switch-case. Penulisan if else Go diawali dengan keyword if kemudian diikuti nilai seleksi kondisi dan blok kode ketika kondisi terpenuhi. Ketika kondisinya tidak terpenuhi akan blok kode else dipanggil (jika blok kode else tersebut ada). Ketika ada banyak kondisi, gunakan else if. Switch merupakan seleksi kondisi yang sifatnya fokus pada satu variabel, lalu kemudian di-cek nilainya. Di GoLang, ketika sebuah case terpenuhi, tidak akan dilanjutkan ke pengecekan case selanjutnya, meskipun tidak ada keyword break di situ. Sebuah case dapat menampung banyak kondisi. Cara penerapannya yaitu dengan menuliskan nilai pembanding-pembanding variabel yang di-switch setelah keyword case dipisah tanda koma.

Tipe Data yang ada di GoLang

1. Integer / int (Bilangan Bulat)
 - int: bergantung platform

- int8: -128 sampai 127
 - int32: -2147483648 sampai 2147483647
 - int64: -9223372036854775808 sampai 9223372036854775807
 - uint: bergantung platform
 - uint8: 0 sampai 255
 - uint32: 0 sampai 4294967295
 - uint64: 0 sampai 18446744073709551615
2. Real / float
 - float32: $\pm 1.18 \times 10^{-38}$ sampai $\pm 3.4 \times 10^{38}$
 - float64: $\pm 2.23 \times 10^{-308}$ sampai $\pm 1.80 \times 10^{308}$
 3. String

serangkaian karakter dengan lebar variabel, dengan setiap karakter diwakili oleh satu atau beberapa byte menggunakan pengodean UTF-8 .
 4. Boolean

Tipe data ini memiliki dua nilai yaitu true atau false. Nilai default tipe data ini adalah false.

II. GUIDED

1. Coso1

Source code:

```
1 //Feros Pedrosa
2
3 package main
4
5 import "fmt"
6
7 func main() {
8     var greetings = "Selamat datang di dunia DAP"
9     var a, b int
10    fmt.Println(greetings)
11    fmt.Scanln(&a, &b)
12    fmt.Printf("%v + %v = %v\n", a, b, a+b)
13 }
14
```

Output:

```
PS C:\ALPRO\semester2_alpro2> go run "c:\ALPRO\semester2_alpro2\alpro2_week2\coso1\coso1.go"
Selamat datang di dunia DAP
8 9
8 + 9 = 17
```

Deskripsi program:

Program diatas ditulis dalam bahasa Go dan berfungsi untuk mencetak pesan sambutan dan menerima dua angka sebagai inputan pengguna, lalu menjumlahkannya dan menampilkan hasilnya. Pertama deklarasikan variabel greetings yang berisi "Selamat datang di dunia DAP" lalu deklarasikan juga variabel a, b bertipe integer. Kemudian program akan menampilkan pesan "Selamat datang di dunia DAP" menggunakan `fmt.Println()`. Selanjutnya program akan membaca inputan pengguna yang disimpan dalam variabel a, b dan program akan menjumlahkan kedua bilangan tersebut dan program juga akan menampilkan hasilnya menggunakan `fmt.Printf()`.

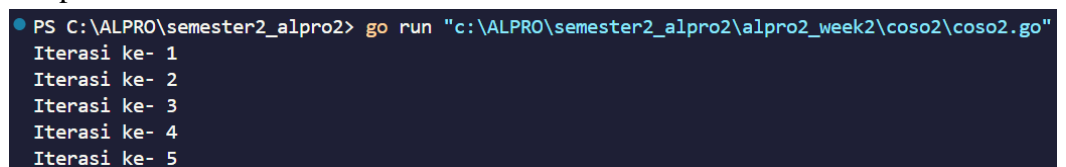
2. Coso2

Source code:



```
1 //Feros Pedrosa
2
3 package main
4
5 import "fmt"
6
7 func main() {
8     for i := 1; i <= 5; i++ {
9         fmt.Println("Iterasi ke-", i)
10    }
11 }
12
```

Output:



```
PS C:\ALPRO\semester2_alpro2> go run "c:\ALPRO\semester2_alpro2\alpro2_week2\coso2\coso2.go"
Iterasi ke- 1
Iterasi ke- 2
Iterasi ke- 3
Iterasi ke- 4
Iterasi ke- 5
```

Deskripsi program:

Program diatas ditulis dalam bahasa Go dan berguna untuk mencetak teks sebanyak 5 kali dengan perulangan for. Program dimulai dengan mendeklarasikan perulangan for dengan variabel i yang dimulai dari 1 dan akan terus bertambah hingga 5 karena ada tanda sama dengannya. Pada setiap iterasi, program akan menampilkan teks "Iterasi ke-" diikuti dengan nilai i, yang menunjukkan nomor iterasinya. Fungsi `fmt.Println` digunakan untuk mencetak output pada setiap perulangan.

3. Coso3

Source code:

```
1 //Feros Pedrosa
2
3 package main
4
5 import "fmt"
6
7 func main() {
8     // contoh nilai (bisa diganti sesuai kebutuhan)
9     nilai := 80
10    pctHadir := 0.75
11    adaTubes := true
12
13    var indeks string
14
15    if nilai > 75 && adaTubes {
16        indeks = "A"
17    } else if nilai > 65 {
18        indeks = "B"
19    } else if nilai > 50 && pctHadir > 0.7 {
20        indeks = "C"
21    } else {
22        indeks = "F"
23    }
24
25    fmt.Printf("Nilai %d dengan kehadiran %.2f%% dan buat tubes=%t, mendapat indeks %s\n",
26        nilai, pctHadir*100, adaTubes, indeks)
27 }
28
```

Output:

```
PS C:\ALPRO\semester2_alpro2> go run "c:\ALPRO\semester2_alpro2\alpro2_week2\coso3\coso3.go"
Nilai 80 dengan kehadiran 75.00% dan buat tubes=true, mendapat indeks A
```

Deskripsi prgram:

Program diatas ditulis dalam bahasa Go dan berfungsi untuk menentukan indeks berdasarkan nilai, persentase kehadiran, dan pengumpulan tugas besar (tubes). Pertama definisikan variabel nilai yang berisi 80, variabel pctHadir yang berisi 0.75, variabel adaTubes yang bernilai true, dan deklarasikan variabel indeks yang bertipe string. Selanjutnya gunakan percabangan if-else untuk menentukan indeks. Jika nilai lebih dari 75 dan mahasiswa mengumpulkan tugas besar maka indeksnya "A". Jika nilai lebih dari 65 maka mendapatkan indeks "B". Jika nilai lebih dari 50 dan persentase kehadiran lebih dari 70% maka indeksnya "C". Selain itu, mahasiswa mendapatkan indeks "F". Terakhir, program mencetak hasil evaluasi menggunakan fmt.Printf(), yang menampilkan nilai, persentase kehadiran dalam bentuk persen, status pengumpulan tugas besar, serta indeks akhir yang didapat.

III. UNGUIDED

1. Latihan soal 1

Source code:

```
1 //Feros Pedrosa
2
3 package main
4
5 import "fmt"
6
7 func main() {
8     var tahun int
9     fmt.Print("Tahun: ")
10    fmt.Scan(&tahun)
11
12    if (tahun%400 == 0) || (tahun%4 == 0 && tahun%100 != 0) {
13        fmt.Print("Kabisat: true\n")
14    } else {
15        fmt.Print("Kabisat: false\n")
16    }
17 }
18
```

Output:

```
PS C:\ALPRO\semester2_alpro2> go run "c:\ALPRO\semester2_alpro2\alpro2_week2\latsol1\latsol1.go"
Tahun: 2016
Kabisat: true
PS C:\ALPRO\semester2_alpro2> go run "c:\ALPRO\semester2_alpro2\alpro2_week2\latsol1\latsol1.go"
Tahun: 2000
Kabisat: true
PS C:\ALPRO\semester2_alpro2> go run "c:\ALPRO\semester2_alpro2\alpro2_week2\latsol1\latsol1.go"
Tahun: 2018
Kabisat: false
```

Deskripsi program:

Program diatas ditulis dalam bahasa Go dan berfungsi untuk menentukan apakah tahun yang diinputkan oleh pengguna merupakan tahun kabisat atau bukan. Pertama deklarasikan variabel tahun untuk menyimpan inputan pengguna. Lalu program akan membaca inputan pengguna dan program akan mengecek apakah inputannya habis dibagi 400, atau inputannya habis dibagi 4 tetapi dibagi 100 hasilnya bukan 0 jika inputannya memenuhi salah satu kondisi tersebut maka outputnya true. Selain itu, maka outputnya akan false.

2. Latihan soal 2

Source code:

```
1 //Feros Pedrosa
2
3 package main
4
5 import "fmt"
6
7 func main() {
8     var k float64
9
10    for {
11        fmt.Print("Nilai K = ")
12        _, err := fmt.Scanln(&k)
13        if err == nil {
14            break
15        }
16        fmt.Println("Input tidak valid.")
17    }
18
19    f_k := (4*k + 2) * (4*k + 2) / ((4*k + 1) * (4*k + 3))
20
21    fmt.Printf("Nilai f(K) = %.10f\n", f_k)
22 }
23
```

Output:

```
PS C:\ALPRO\semester2_alpro2> go run "c:\ALPRO\semester2_alpro2\latsol2fk\latsol2(f(k)).go"
Nilai K = 100
Nilai f(K) = 1.0000061880
```

Deskripsi program:

Program diatas ditulis dalam bahasa Go dan berfungsi untuk menghitung nilai fungsi k dengan cara mengganti k nya dengan inputan pengguna. Pertama deklarasikan variabel k dengan tipe data float64. Selanjutnya program akan masuk ke dalam loop for yang tidak terbatas. Jika input yang dimasukkan valid maka program akan keluar dari loop dan melanjutkan perhitungan. Jika input yang dimasukkan salah maka program akan mengeluarkan pesan "Input tidak valid." dan meminta pengguna untuk mencoba lagi. Setelah mendapatkan inputan yang valid program akan melanjutkan perhitungan dengan memakai rumus $(4*k + 2) * (4*k + 2) / ((4*k + 1) * (4*k + 3))$

$((4*k + 1) * (4*k + 3))$. Terakhir program akan menampilkan hasil perhitungan dengan format 10 angka dibelakang koma.

3. Latihan soal 3

Source code:

```
1 //Feros Pedrosa
2
3 package main
4
5 import "fmt"
6
7 func main() {
8
9     var berat, beratKg, sisaGram, biayaKg, biayaSisaGram, totalBiaya int
10
11     fmt.Println("Masukkan berat parsel (gram): ")
12     fmt.Scan(&berat)
13
14     beratKg = berat / 1000
15     sisaGram = berat % 1000
16
17     biayaKg = beratKg * 10000
18
19     if sisaGram > 0 {
20         if beratKg > 10 {
21             biayaSisaGram = 0
22         } else if sisaGram >= 500 {
23             biayaSisaGram = sisaGram * 5
24         } else {
25             biayaSisaGram = sisaGram * 15
26         }
27     }
28
29     totalBiaya = biayaKg + biayaSisaGram
30
31     fmt.Printf("Detail berat: %d kg + %d gr\n", beratKg, sisaGram)
32     fmt.Printf("Detail biaya: Rp. %d + Rp. %d\n", biayaKg, biayaSisaGram)
33     fmt.Printf("Total biaya: Rp. %d\n", totalBiaya)
34 }
35
```

Output:

```
PS C:\ALPRO\semester2_alpro2> go run "c:\ALPRO\semester2_alpro2\latsol3\latsol3.go"
Masukkan berat parcel (gram): 8500
Detail berat: 8 kg + 500 gr
Detail biaya: Rp. 80000 + Rp. 2500
Total biaya: Rp. 82500
PS C:\ALPRO\semester2_alpro2> go run "c:\ALPRO\semester2_alpro2\latsol3\latsol3.go"
Masukkan berat parcel (gram): 9250
Detail berat: 9 kg + 250 gr
Detail biaya: Rp. 90000 + Rp. 3750
Total biaya: Rp. 93750
PS C:\ALPRO\semester2_alpro2> go run "c:\ALPRO\semester2_alpro2\latsol3\latsol3.go"
Masukkan berat parcel (gram): 11750
Detail berat: 11 kg + 750 gr
Detail biaya: Rp. 110000 + Rp. 0
Total biaya: Rp. 110000
```

Deskripsi program:

Program diatas ditulis dalam bahasa Go dan berfungsi untuk menghitung biaya pengiriman parcel berdasarkan beratnya dalam gram. Pertama, deklarasikan variabel berat, beratKg, sisaGram, biayaKg, biayaSisaGram, totalBiaya sebagai tipe data integer. Pengguna diminta untuk memasukkan berat parcel, yang kemudian diolah untuk menentukan berat dalam kilogram dan sisa gram. Biaya pengiriman dihitung berdasarkan tarif per kilogram dan sisa gram, dengan ketentuan khusus: jika berat parcel lebih dari 10 kilogram, biaya untuk sisa gram akan menjadi nol; jika sisa gram lebih dari atau sama dengan 500 gram, biaya per gram adalah Rp. 5; dan jika kurang dari 500 gram, biaya per gram adalah Rp. 15. Setelah menghitung biaya untuk kilogram dan sisa gram, program akan menjumlahkan kedua biaya tersebut untuk mendapatkan total biaya pengiriman. Hasil akhir ditampilkan dalam format yang jelas, menunjukkan detail berat, detail biaya, dan total biaya pengiriman.

IV. KESIMPULAN

Terdapat dua jenis struktur kontrol yang dipakai untuk mengerjakan laporan praktikum ini yaitu percabangan dan perulangan. Di percabangan ada if-else dan switch-case. Penulisan if else di GoLang diawali dengan keyword if kemudian diikuti nilai dan blok kode ketika kondisi terpenuhi. Ketika kondisinya tidak terpenuhi akan blok kode else dipanggil (jika blok kode else tersebut ada). Ketika ada banyak kondisi, gunakan else if.

Pada laporan praktikum ini juga ada perulangan for untuk mengerjakan soalnya. Perulangan adalah proses mengulang-ulang eksekusi blok kode tanpa henti, selama kondisi yang dijadikan acuan terpenuhi. Biasanya disiapkan variabel untuk iterasi atau variabel penanda kapan perulangan akan diberhentikan. Ada dua statement yang bisa ditambahkan yaitu Init statement dan Post statement. Init statement adalah sebuah statement sebelum for loop dieksekusi. Post statement adalah statement yang akan selalu dieksekusi di akhir pada setiap perulangan. Init statement hanya dieksekusi sekali sehingga bisa digunakan untuk menginisialisasi variabel awal yang menentukan counter perulangan. Sedangkan post statement yang selalu dieksekusi pada akhir tiap perulangan bisa digunakan untuk menaikkan nilai counter perulangan.

REFERENSI

Modul 2 – Praktikum Alpro 2

A.13. Seleksi Kondisi. (n.d.). Retrieved from

<https://dasarpemrogramangolang.novalagung.com/A-seleksi-kondisi.html>