

xALGORITMA PEMOGRAMAN 2
MODUL 2
MATERI REVIEW STRUKTUR KONTROL



Oleh:

NAMA: Davi Ilyas Renaldo

NIM: 103112400062

KELAS: 12-IF-01

S1 TEKNIK INFORMATIKA
TELKOM UNIVERSITY PURWOKERTO
2025

DASAR TEORI

Go, bahasa pemrograman yang dikembangkan oleh Google, terkenal karena kesederhanaan dan efisiensinya. Salah satu fitur yang membuat Go menjadi pilihan yang populer di antara para pengembang adalah struktur kontrol yang kuat dan mudah digunakan. Dalam artikel ini, kita akan menjelajahi penggunaan struktur kontrol utama dalam Go: **if**, **for**, dan **switch**, serta bagaimana Anda dapat memanfaatkannya dalam pengembangan perangkat lunak.

If Statement

Statement if digunakan untuk mengevaluasi kondisi dan menjalankan blok kode jika kondisinya benar (true).

Go mendukung beberapa jenis loop, tetapi yang paling umum adalah loop **for**.

Loop **for** digunakan untuk mengulangi blok kode selama kondisi yang ditentukan benar.

Switch Statement

Pernyataan switch digunakan untuk memeriksa beberapa kondisi dan menjalankan blok kode yang sesuai dengan kondisi yang terpenuhi.

Dengan memahami penggunaan yang tepat dari if, for, dan switch dalam Go, Anda dapat menulis kode yang lebih bersih, mudah dimengerti, dan efisien. Teruslah eksplorasi dan praktikkan struktur kontrol ini dalam pengembangan perangkat lunak Anda untuk mencapai hasil yang optimal.

I. GUIDED

1.

```
GUIDED > 1.go > main
1  package main
2
3  import "fmt"
4
5  func main() {
6      var greetings = "selamat datang didunia DAP"
7      var a, b int
8      fmt.Println(greetings)
9      fmt.Scan(&a, &b)
10     fmt.Printf("%v + %v = %v\n", a, b, a+b)
11
12 }
```

OUTPUT:

```
PS C:\Users\ACER\OneDrive\alpro2\modul1> go run "c:\Users\ACER\OneDrive\alpro2\modul1\GUIDED\1.go"
selamat datang didunia DAP
2 8
2 + 8 = 10
PS C:\Users\ACER\OneDrive\alpro2\modul1> 
```

DESKRIPSI:

Program ini memberikan kata sambutan "selamat datang didunia DAP", kemudian meminta input kepada pengguna untuk memasukkan dua angka kemudian program tersebut akan menghitung dan menampilkan hasil penjumlahan dari angka yang diinputkan oleh pengguna.

2.

```
GUIDED > go 2.go > main
1  package main
2
3  import "fmt"
4
5  func main() {
6      for i := 1; i <= 5; i++ {
7          fmt.Println("iterasi ke-", i)
8      }
9  }
```

OUTPUT:

```
PS C:\Users\ACER\OneDrive\alpro2\modul1> go run "c:\Users\ACER\OneDrive\alpro2\modul1\GUIDED\2.go"
iterasi ke- 1
iterasi ke- 2
iterasi ke- 3
iterasi ke- 4
iterasi ke- 5
PS C:\Users\ACER\OneDrive\alpro2\modul1> |
```

DESKRIPSI:

Program ini menggunakan loop untuk mencetak pesan "iterasi ke- X" sebanyak 5 kali, dimulai dari X = 1 hingga X = 5.

3.

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

OUTPUT:

```
PS C:\Users\ACER\OneDrive\alpro2\modul1> go run "c:\Users\ACER\OneDrive\alpro2\modul1\GUIDED\3.go"
Nilai 80 dengan kehadiran 75.00% dan buat tubes=true, mendapat indeks A
PS C:\Users\ACER\OneDrive\alpro2\modul1>
```

DESKRIPSI:

Program ini menentukan indeks nilai berdasarkan nilai ujian, kehadiran dan tugas besar. Dimana berjalannya program ini sudah ditentukan berdasarkan angka yg diinputkan pada kode program tersebut.

II. UNGUIDED

1.

```
UNGUIDED > go 2a2.go > main
1 //M.DAVI ILYAS RENALDO
2 //103112400062
3
4 package main
5
6 import "fmt"
7
8 func main() {
9     var tahun int
10    fmt.Scan(&tahun)
11
12    kabisat := tahun %400 == 0 || (tahun %4 == 0 && tahun%100 !=0)
13    fmt.Printf("tahun: %d\nkabisat: %t\n", tahun, kabisat)
14 }
```

OUTPUT:

```
tahun: 2016
kabisat: true
PS C:\Users\ACER\OneDrive\alpro2\modul1> go run "c:\Users\ACER\OneDrive\alpro2\modul1\UNGUIDED\2a2.go"
2018
tahun: 2018
kabisat: false
PS C:\Users\ACER\OneDrive\alpro2\modul1> go run "c:\Users\ACER\OneDrive\alpro2\modul1\UNGUIDED\2a2.go"
2000
tahun: 2000
kabisat: true
```

DESKRIPSI:

Program ini adalah program untuk memeriksa apakah tahun yang dimasukkan oleh pengguna merupakan tahun kabisat atau bukan. Tahun kabisat merupakan tahun yang jumlah harinya habis dibagi 400 atau 4 tetapi tidak habis dibagi 100.

2.

```
UNGUIDED > go 2b4.go > ...
1 //M.DAVI ILYAS RENALDO
2 //103112400062
3 |
4 package main
5
6 import "fmt"
7
8 func akar2(K int) float64 {
9     hasil := 1.0
10    for k := 0; k <= K; k++ {
11        pembilang := float64((4*k + 2) * (4*k + 2))
12        penyebut := float64((4*k + 1) * (4*k + 3))
13        hasil *= pembilang / penyebut
14    }
15    return hasil
16 }
17
18 func main() {
19     var K int
20     fmt.Print("Nilai K = ")
21     fmt.Scan(&K)
22
23     hasil := akar2(K)
24     fmt.Printf("Nilai akar 2 = %.10f\n", hasil)
25 }
```

OUTPUT:

```
PS C:\Users\ACER\OneDrive\alpro2\modul1> go run "c:\Users\ACER\OneDrive\alpro2\modul1\UNGUIDED\2b4.go"
Nilai K = 10
Nilai akar 2 = 1.4062058441
PS C:\Users\ACER\OneDrive\alpro2\modul1> go run "c:\Users\ACER\OneDrive\alpro2\modul1\UNGUIDED\2b4.go"
Nilai K = 100
Nilai akar 2 = 1.413387072
PS C:\Users\ACER\OneDrive\alpro2\modul1> go run "c:\Users\ACER\OneDrive\alpro2\modul1\UNGUIDED\2b4.go"
Nilai K = 1000
Nilai akar 2 = 1.4141252651
PS C:\Users\ACER\OneDrive\alpro2\modul1> |
```

DESKRIPSI:

Program ini program yang meminta inputan angka bilangan bulat dari pengguna, lalu program ini akan menghitung secara berulang-ulang sampai mendapatkan nilai yang mendekati akar 2. Kemudian hasilnya akan ditampilkan dengan format 10 angka decimal.

3.

```
UNGUIDED > 2c1.go > main
1 //M.DAVI ILYAS RENALDO
2 //103112400062
3
4 package main
5
6 import "fmt"
7
8 func main() {
9     var berat_asli int
10    fmt.Print("berat parsel (gram): ")
11    fmt.Scanln(&berat_asli)
12
13    kg := berat_asli / 1000
14    gram := berat_asli % 1000
15
16    fmt.Printf("detail berat: %d kg + %d gr\n", kg, gram)
17
18    biaya_kg := kg * 10000
19    var biaya_gram int
20    if gram >= 500 {
21        biaya_gram = gram * 5
22    } else {
23        biaya_gram = gram * 15
24    }
25    fmt.Printf("detail biaya: Rp. %d + Rp. %d\n", biaya_kg, biaya_gram)
26    fmt.Printf("total biaya: Rp. %d", biaya_kg+biaya_gram)
27 }
```

OUTPUT:

```
berat parsel (gram): 8500
detail berat: 8 kg + 500 gr
detail biaya: Rp. 80000 + Rp. 2500
total biaya: Rp. 82500
PS C:\Users\ACER\OneDrive\alpro2\modul1> go run "c:\Users\ACER\OneDrive\alpro2\modul1\UNGUIDED\2c1.go"
berat parsel (gram): 9250
detail berat: 9 kg + 250 gr
detail biaya: Rp. 90000 + Rp. 3750
total biaya: Rp. 93750
PS C:\Users\ACER\OneDrive\alpro2\modul1> go run "c:\Users\ACER\OneDrive\alpro2\modul1\UNGUIDED\2c1.go"
```

DESKRIPSI:

Program diatas adalah suatu program untuk menentukan total biaya dari suatu parsel berdasarkan dari berat parsel yang diinputkan oleh pengguna.

III. KESIMPULAN

Berdasarkan laporan praktikum saya diatas , dapat disimpulkan bahwa struktur kontrol dalam Bahasa (Go if, for, switch) memungkinkan kita untuk mengatur alur eksekusi program dengan cara yang lebih jelas dan efisien.

If: digunakan untuk menjalankan kode berdasarkan kondisi yang diberikan.

Switch: digunakan untuk memeriksa beberapa kondisi sekaligus dengan cara yang lebih sederhana daripada menggunakan if.

For: digunakan untuk mengulang kode, bisa dengan kondisi tertentu atau dengan iterasi.

IV. REFERENSI

- Modul 2 review struktur control
- <https://buildwithangga.com/tips/struktur-kontrol-dalam-go-penggunaan-if-for-dan-switch>