

**LAPORAN PRAKTIKUM
ALGORITMA DAN PEMROGRAMAN 2**

MODUL 2

MATERI



Oleh:

ARIEL AHNAF KUSMA

103112400050

12-IF-01

**S1 TEKNIK INFORMATIKA
TELKOM UNIVERSITY PURWOKERTO
2025**

I. DASAR TEORI

Dasar Teori

Golang, atau Go adalah bahasa pemrograman yang lahir di tahun 2009. Golang memiliki banyak kelebihan, terbukti dengan banyaknya perusahaan besar yang menggunakan bahasa ini dalam pengembangan produk-produk mereka, hingga level production tentunya

PERULANGAN

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.

KARAKTERISTIK FOR-LOOP

Dalam pemrograman, loop digunakan untuk mengulang blok kode. Misalnya,

Jika kita ingin mencetak suatu pernyataan sebanyak 100 kali, daripada menulis pernyataan cetak yang sama sebanyak 100 kali, kita dapat menggunakan perulangan untuk mengeksekusi kode yang sama sebanyak 100 kali.

Dalam Golang, kita menggunakan forloop untuk mengulang blok kode hingga kondisi yang ditentukan terpenuhi.

KARAKTERISTIK IF ELSE

Cara penerapan if-else di Go sama seperti pada bahasa pemrograman lain. Yang membedakan hanya tanda kurungnya (*parentheses*), di Go tidak perlu ditulis. Kode berikut merupakan contoh penerapan seleksi kondisi if else, dengan jumlah kondisi 4 buah.

PERCABANGAN

Percabangan adalah fitur dari bahasa pemrograman yang melakukan perhitungan atau tindakan yang berbeda tergantung pada apakah kondisi boolean yang ditentukan pemrogram mengevaluasi benar atau salah. Jika dalam golang

Percabangan digunakan untuk mengontrol alur eksekusi flow program.

Analoginya mirip seperti fungsi rambu lalu lintas di jalan raya. Kapan kendaraan diperbolehkan melaju dan kapan harus berhenti diatur oleh rambu tersebut.

Seleksi kondisi pada program juga kurang lebih sama, kapan sebuah blok kode dieksekusi dikontrol.

I. GUIDED

SOURCE CODE

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

OUTPUT

```
PS C:\ARIEL AHNAF KUSUMA\LAPRAK 1 ALPRO 2> go run "c:\ARIEL AHNAF KUSUMA\LAPRAK 1 ALPRO 2
\PRAKTIKUM 1\GUIDED 1\contoh.go"
Selamat datang di dunia DAP
6 5
6 + 5 = 11
PS C:\ARIEL AHNAF KUSUMA\LAPRAK 1 ALPRO 2> 
```

DESKRIPSI CODE

SOURCE CODE

```
package main

import "fmt"

func main() {
    for i := 1; i <= 5; i++ {
        fmt.Println("iterasi ke-", i)
    }
}
```

OUTPUT

```
PS C:\ARIEL AHNAF KUSUMA\LAPRAK 1 ALPRO 2> go run "c:\ARIEL AHNAF KUSUMA\LAPRAK 1 ALPRO 2
\PRAKTIKUM 1\GUIDED 2\contoh2.go"
iterasi ke- 1
iterasi ke- 2
iterasi ke- 3
iterasi ke- 4
iterasi ke- 5
PS C:\ARIEL AHNAF KUSUMA\LAPRAK 1 ALPRO 2>
```

DESKRIPSI CODE

SOURCE CODE

```
1 package main
2
3 import "fmt"
4
5 func main() {
6     // Contoh nilai (bisa diganti sesuai ketentuan)
7     nilai := 80
8     pctHadir := 0.75
9     adaTubes := true
10
11     var indeks string
12     if nilai > 75 && adaTubes {
13         indeks = "A"
14     } else if nilai > 65 {
15         indeks = "B"
16     } else if nilai > 55 {
17         indeks = "C"
18     } else {
19         indeks = "F"
20     }
21
22     fmt.Printf("Nilai %d dengan kehadiran %.2f dan buat tubes=%t, mendapat indeks %s\n", nilai, pctHadir*100, adaTubes, indeks)
23 }
24
```

OUTPUT

```
PS C:\ARIEL AHNAF KUSUMA\LAPRAK 1 ALPRO 2> go run "c:\ARIEL AHNAF KUSUMA\LAPRAK 1 ALPRO 2\PRAKTI
KUM 1\GUIDED 3\contoh3.go"
Nilai 80 dengan kehadiran 75.00 dan buat tubes=true, mendapat indeks A
PS C:\ARIEL AHNAF KUSUMA\LAPRAK 1 ALPRO 2>
```

DESKRIPSI CODE

II. UNGUIDED

SOURCE CODE

```
1  package main
2
3  //ARIEL AHNAF KUSUMA 103112400050
4  import "fmt"
5
6  func main() {
7      var tahun int
8      var kabisat bool
9      fmt.Print("Masukkan Tahun: ")
10     fmt.Scan(&tahun)
11     kabisat = (tahun%4 == 0 && tahun%100 != 0)
12     fmt.Println("Tahun Kabisat:", kabisat)
13 }
14
```

OUTPUT

```
PS C:\ARIEL AHNAF KUSUMA\LAPRAK 1 ALPRO 2> go run "c:\ARIEL AHNAF KUSUMA\LAPRAK 1 ALPRO 2\PRAKTI
KUM 1\2A NO 2\unguided1.go"
Masukkan Tahun: 2016
Tahun Kabisat: true
PS C:\ARIEL AHNAF KUSUMA\LAPRAK 1 ALPRO 2> go run "c:\ARIEL AHNAF KUSUMA\LAPRAK 1 ALPRO 2\PRAKTI
KUM 1\2A NO 2\unguided1.go"
Masukkan Tahun: 2000
Tahun Kabisat: true
PS C:\ARIEL AHNAF KUSUMA\LAPRAK 1 ALPRO 2> go run "c:\ARIEL AHNAF KUSUMA\LAPRAK 1 ALPRO 2\PRAKTI
KUM 1\2A NO 2\unguided1.go"
Masukkan Tahun: 2018
Tahun Kabisat: false
PS C:\ARIEL AHNAF KUSUMA\LAPRAK 1 ALPRO 2> █
```

DESKRIPSI CODE

Program ini menentukan suatu tahun apakah tahun tersebut kabisat atau bukan, tahun kabisat adalah tahun yang habis dibagi 4 maka dari itu program ini akan bekerja untuk mencari tahun kabisat dan yang bukan termasuk tahun kabisat.

SOURCE CODE

```
1  package main
2
3  //ARIEL AHNAF KUSUMA 103112400050
4  import "fmt"
5
6  func main() {
7      var k int
8      fmt.Print("Nilai K = ")
9      fmt.Scan(&k)
10
11     akar2 := 1.0
12     for k := 0; k <= k; k++ {
13         x := 4*float64(k) + 2
14         y := 4*float64(k) + 1
15         akar2 *= (x * x) / (y * (y + 2))
16     }
17
18     fmt.Printf("Nilai akar 2 = %.10f\n", akar2)
19 }
20
```


OUTPUT

```
PS C:\ARIEL AHNAF KUSUMA\LAPRAK 1 ALPRO 2> go run "c:\ARIEL AHNAF KUSUMA\LAPRAK 1 ALPRO 2\PRAKTI
KUM 1\2B NO 4 (FOR)\unguided2.go"
Nilai K = 100
Nilai akar 2 = 1.4133387072
PS C:\ARIEL AHNAF KUSUMA\LAPRAK 1 ALPRO 2> █
```

DESKRIPSI CODE

Program ini meminta pengguna memasukkan nilai k dan kemudian menggunakan perulangan for untuk menghitung hasil rumus. Dalam loop, variabel x dan y dihitung berdasarkan nilai k, dan hasil perhitungan dikalikan secara bertahap ke dalam akar2. Setelah perulangan selesai, hasil akhirnya dicetak dengan format desimal yang memiliki 10 angka di belakang koma.. Semisal jika k = 100, maka hasilnya 1.4133387072.

SOURCE CODE

```

1  package main
2
3  //ARIEL AHNAF KUSUMA 103112400050
4  import "fmt"
5
6  func main() {
7      var beratgram int
8      fmt.Print("Berat parcel (gram): ")
9      fmt.Scanln(&beratgram)
10
11      kg, gram := beratgram/1000, beratgram%1000
12      biayakg, biayagram := kg*10000, 0
13
14      if gram >= 500 {
15          biayagram = gram * 5
16      } else {
17          biayagram = gram * 15
18      }
19
20      fmt.Printf("Detail berat: %d kg + %d gr\n", kg, gram)
21      fmt.Printf("Detail biaya: Rp. %d + Rp. %d\n", biayakg, biayagram)
22      fmt.Printf("Total biaya: Rp. %d\n", biayakg+biayagram)
23  }
24

```

OUTPUT

```

PS C:\ARIEL AHNAF KUSUMA\LAPRAK 1 ALPRO 2> go run "c:\ARIEL AHNAF KUSUMA\LAPRAK 1 ALPRO 2\PRAKTI
KUM 1\2C NO 1 (IF ELSE)\unguided3.go"
Berat parcel (gram): 8500
Detail berat: 8 kg + 500 gr
Detail biaya: Rp. 80000 + Rp. 2500
Total biaya: Rp. 82500
PS C:\ARIEL AHNAF KUSUMA\LAPRAK 1 ALPRO 2> go run "c:\ARIEL AHNAF KUSUMA\LAPRAK 1 ALPRO 2\PRAKTI
KUM 1\2C NO 1 (IF ELSE)\unguided3.go"
Berat parcel (gram): 9250
Detail berat: 9 kg + 250 gr
Detail biaya: Rp. 90000 + Rp. 3750
Total biaya: Rp. 93750
PS C:\ARIEL AHNAF KUSUMA\LAPRAK 1 ALPRO 2> go run "c:\ARIEL AHNAF KUSUMA\LAPRAK 1 ALPRO 2\PRAKTI
KUM 1\2C NO 1 (IF ELSE)\unguided3.go"
Berat parcel (gram): 11750
Detail berat: 11 kg + 750 gr
Detail biaya: Rp. 110000 + Rp. 3750
Total biaya: Rp. 113750
PS C:\ARIEL AHNAF KUSUMA\LAPRAK 1 ALPRO 2>

```

DEKRIPISI CODE

Program ini menghitung biaya pengiriman dengan mengonversi berat parcel ke kilogram dan gram. Biaya dasar ditetapkan Rp. 10.000 per kilogram, sedangkan biaya tambahan dikenakan Rp. 5 per gram jika sisa gram ≥ 500 , dan Rp. 15 per gram jika kurang dari 500.

III. KESIMPULAN

Struktur kontrol dalam pemrograman—terutama perulangan dan percabangan—sangat penting untuk pengembangan algoritma. Menghitung jumlah, menentukan tahun kabisat, menghitung fungsi matematika, dan menghitung biaya pengiriman adalah beberapa contoh tugas yang diterapkan oleh program yang dijelaskan.

Program dapat beroperasi secara dinamis dan mengubah outputnya sesuai dengan kondisi saat ini dengan menggunakan perulangan for dan percabangan if-else. Secara keseluruhan, pemahaman dan penerapan struktur kontrol ini sangat penting untuk membuat program yang efektif yang dapat menyelesaikan berbagai masalah yang rumit.

IV. REFERENSI

DASAR PEMROGRAMAN GOLANG BY NOVAL AGUNG

PROGRAMIZ.COM

WIKIPEDIA.COM