

LAPORAN PRAKTIKUM
ALGORITMA DAN PEMROGRAMAN 1
MODUL 9
“IF THEN”



DISUSUN OLEH:
RIZKINA AZIZAH
103112400082
S1 IF-12-01

DOSEN:
Yohani Setiya Rafika Nur, M. Kom.

PROGRAM STUDI S1 TEKNIK INFORMATIKA
FAKULTAS INFORMATIKA
TELKOM UNIVERSITY PURWOKERTO
2024/2025

DASAR TEORI

Dalam Go, kita dapat menggunakan pernyataan kondisional untuk mengeksekusi kode berdasarkan kondisi yang berbeda. Kita juga dapat menggunakan pernyataan `if`, `else if`, dan `if else` untuk mengeksekusi kode yang diberikan jika suatu kondisi benar atau salah.

- Pernyataan **`if`** - mengeksekusi beberapa kode jika satu kondisi benar
- Pernyataan **`if...else`** - mengeksekusi beberapa kode jika suatu kondisi benar dan kode lain jika kondisi tersebut salah
- Pernyataan **`if...else if....else`** - mengeksekusi kode yang berbeda untuk lebih dari dua kondisi
- Pernyataan **`switch...case`** - memilih salah satu dari banyak blok kode yang akan dieksekusi

Seleksi kondisi 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. Yang dijadikan acuan oleh seleksi kondisi adalah nilai bertipe `bool`, bisa berasal dari variabel, ataupun hasil operasi perbandingan. Nilai tersebut menentukan blok kode mana yang akan dieksekusi. Go memiliki 2 macam keyword untuk seleksi kondisi, yaitu **`if else`** dan **`switch`**.

a. Seleksi Kondisi Menggunakan Keyword `if`, `else 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.

b. Variabel Temporary Pada `if – else`

Variabel temporary adalah variabel yang hanya bisa digunakan pada deretan blok seleksi kondisi di mana ia ditempatkan. Penggunaan variabel ini membawa beberapa manfaat, antara lain:

- Scope atau cakupan variabel jelas, hanya bisa digunakan pada blok seleksi kondisi itu saja
- Kode menjadi lebih rapi
- Ketika nilai variabel tersebut didapat dari sebuah komputasi, perhitungan tidak perlu dilakukan di dalam blok masing-masing kondisi.

c. Seleksi Kondisi Menggunakan Keyword switch – case

Switch merupakan seleksi kondisi yang sifatnya fokus pada satu variabel, lalu kemudian di-cek nilainya. Contoh sederhananya seperti penentuan apakah nilai variabel `x` adalah: 1, 2, 3, atau lainnya.

d. Pemanfaatan case Untuk Banyak Kondisi

Sebuah case dapat menampung banyak kondisi. Cara penerapannya yaitu dengan menuliskan nilai pembanding-pembanding variabel yang di-switch setelah keyword case dipisah tanda koma (,).

e. Kurung Kurawal Pada Keyword case & default

Tanda kurung kurawal ({ }) bisa diterapkan pada keyword case dan default. Tanda ini opsional, boleh dipakai boleh tidak. Bagus jika dipakai pada blok kondisi yang di dalamnya ada banyak statement, dengannya kode akan terlihat lebih rapi.

f. Switch Dengan Gaya if - else

Uniknya di Go, switch bisa digunakan dengan gaya ala if-else. Nilai yang akan dibandingkan tidak dituliskan setelah keyword switch, melainkan akan ditulis langsung dalam bentuk perbandingan dalam keyword case.

g. Penggunaan Keyword fallthrough Dalam switch

Seperti yang sudah dijelaskan sebelumnya, bahwa switch pada Go memiliki perbedaan dengan bahasa lain. Ketika sebuah case terpenuhi, pengecekan kondisi tidak akan diteruskan ke case-case setelahnya.

Keyword fallthrough digunakan untuk memaksa proses pengecekan tetap diteruskan ke case selanjutnya dengan **tanpa menghiraukan nilai kondisinya**, efeknya adalah case di pengecekan selanjutnya selalu dianggap true (meskipun aslinya bisa saja kondisi tersebut tidak terpenuhi, akan tetap dianggap true).

h. Seleksi Kondisi Bersarang

Seleksi kondisi bersarang adalah seleksi kondisi, yang berada dalam seleksi kondisi, yang mungkin juga berada dalam seleksi kondisi, dan seterusnya. Seleksi kondisi bersarang bisa dilakukan pada if - else, switch, ataupun kombinasi keduanya.

CONTOH SOAL

1. Latihan1

Sebuah program digunakan untuk menghitung nilai absolut atau mutlak dari suatu bilangan.

Masukan terdiri dari suatu bilangan bulat.

Keluaran berupa nilai absolut atau mutlak dari bilangan yang diberikan.

Contoh masukan dan keluaran:

NO	Masukan	Keluaran
1	10	10
2	-3	3
3	5	5
4	0	0
5	-9999	9999

Source Code:

```
package main

import "fmt"

func main (){
    var bilangan int
    fmt.Scan(&bilangan)
    if bilangan < 0 {
        bilangan = -bilangan
    }
    fmt.Print(bilangan)
}
```

Output:

```
PS C:\Users\HP\Desktop\tugas> go run "c:\Users\HP\Desktop\tugas\modul 9\1.go"
10
10
PS C:\Users\HP\Desktop\tugas> go run "c:\Users\HP\Desktop\tugas\modul 9\1.go"
-3
3
PS C:\Users\HP\Desktop\tugas> go run "c:\Users\HP\Desktop\tugas\modul 9\1.go"
5
5
PS C:\Users\HP\Desktop\tugas> go run "c:\Users\HP\Desktop\tugas\modul 9\1.go"
0
0
PS C:\Users\HP\Desktop\tugas> go run "c:\Users\HP\Desktop\tugas\modul 9\1.go"
-9999
9999
```

Deskripsi Program:

- Program ini digunakan untuk menghitung nilai absolut atau mutlak dari suatu bilangan.
- Mendeklarasikan satu variabel yaitu **bilangan** dengan tipe data integer
- Intruksi Scan digunakan untuk mendeklarasikan input **bilangan**
- Menggunakan paradigma percabangan yaitu if-then :
if sebagai kondisi atau syarat = jika bilangan kurang 0
then ({}) sebagai aksi = maka bilangan * -1
- Intruksi Println digunakan untuk mendaklarasikan output bilangan

2. Latihan1

Buatlah program Go yang digunakan untuk menentukan apakah suatu bilangan yang diberikan adalah bilangan positif atau bukan.

Masukan terdiri dari suatu bilangan bulat.

Keluaran terdiri dari teks yang menyatakan bilangan bulat adalah "positif" atau "bukan positif".

Contoh masukan dan keluaran:

NO	Masukan	Keluaran
1	10	positif
2	-3	bukan positif
3	5	positif
4	0	bukan positif

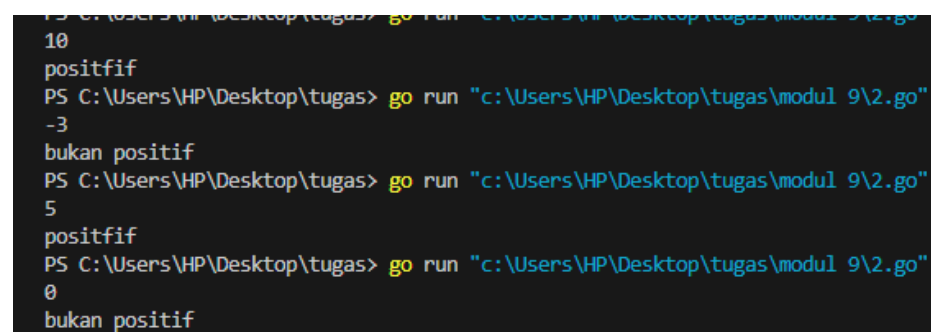
Source Code:

```
package main

import "fmt"

func main () {
    var a int
    var teks string
    fmt.Scan(&a)
    teks = "bukan positif"
    if a >= 0 {
        teks = "positif"
    }
    fmt.Println(teks)
}
```

Output:



```
PS C:\Users\HP\Desktop\tugas> go run "c:\Users\HP\Desktop\tugas\modul 9\2.go"
10
positif
PS C:\Users\HP\Desktop\tugas> go run "c:\Users\HP\Desktop\tugas\modul 9\2.go"
-3
bukan positif
PS C:\Users\HP\Desktop\tugas> go run "c:\Users\HP\Desktop\tugas\modul 9\2.go"
5
positif
PS C:\Users\HP\Desktop\tugas> go run "c:\Users\HP\Desktop\tugas\modul 9\2.go"
0
bukan positif
```

Deskripsi Program:

- Program ini digunakan untuk menentukan apakah suatu bilangan yang diberikan adalah bilangan positif atau bukan.
- Mendeklarasikan dua variabel yaitu `a` dengan tipe data integer dan `teks` dengan tipe data string
- Intruksi Scan digunakan untuk mendeklarasikan input `a`

- Menggunakan paradigma percabangan yaitu if-then :
if sebagai kondisi atau syarat = jika a lebih sama dengan 0
then ({}) sebagai aksi = maka positif
- Intruksi Println digunakan untuk mendaklarasikan output teks, jika output tidak sesuai syarat maka akan menghasilkan output kebalikannya (“bukan positif”)

3. Latihan1

Buatlah suatu program dengan menggunakan bahasa pemrograman Go, yang mana program digunakan untuk menentukan apakah bilangan yang diberikan adalah bilangan genap negatif atau bukan.

Masukan terdiri dari suatu bilangan bulat.

Keluaran terdiri dari suatu boolean yang menyatakan true apabila bilangan yang diberikan adalah genap negatif, atau false apabila bukan genap negatif.

Contoh masukan dan keluaran:

NO	Masukan	Keluaran
1	10	false
2	-3	false
3	-4	true
4	0	false
5	-2	true

Source Code:

```
package main

import "fmt"

func main () {
    var a int
    var hasil bool
    fmt.Scan(&a)
    if a < 0 && a%2 == 0 {
        hasil = true
    }
}
```

```
    fmt.Println(hasil)
}
```

Output:

```
10
false
PS C:\Users\HP\Desktop\tugas> go run "c:\Users\HP\Desktop\tugas\modul 9\tempCodeRunnerFile.go"
\Desktp\tugas\modul 9\tempCodeRunnerFile.go"
-4
true
PS C:\Users\HP\Desktop\tugas> go run "c:\Users\HP\Desktop\tugas\modul 9\tempCodeRunnerFile.go"
0
false
PS C:\Users\HP\Desktop\tugas> go run "c:\Users\HP\Desktop\tugas\modul 9\tempCodeRunnerFile.go"
-2
true
PS C:\Users\HP\Desktop\tugas> █
```

Deskripsi Program:

- Program ini digunakan untuk menentukan apakah bilangan yang diberikan adalah bilangan genap negatif atau bukan.
- Mendeklarasikan dua variabel yaitu `a` dengan tipe data integer dan `hasil` dengan tipe data boolean
- Intruksi Scan digunakan untuk mendeklarasikan input `a`
- Menggunakan paradigma percabangan yaitu if-then :
if sebagai kondisi atau syarat = jika `a` kurang dari 0 dan habis dibagi 2
then (`{}`) sebagai aksi = maka true
- Intruksi Println digunakan untuk mendaklarasikan output `hasil`, jika output tidak sesuai syarat maka akan menghasilkan output kebalikannya (false)

SOAL LATIHAN

Statement perulangan

1. Latihan 1

Sebuah program Go digunakan untuk menentukan jumlah motor yang diperlukan seseorang untuk melakukan touring. Satu motor hanya cukup untuk dua orang, yaitu satu pengemudi dan satu tumpangan. Setiap motor diprioritaskan untuk terisi dua orang.

Masukan terdiri dari bilangan bulat yang menyatang jumlah orang yang akan melakukan touring.

Keluaran berupa jumlah motor yang diperlukan peserta touring.

Contoh masukan dan keluaran:

NO	Masukan	Keluaran
1	10	5
2	1	1
3	25	13
4	9	5

Source Code:

```
package main
import "fmt"

func main(){
    var n int
    fmt.Scan(&n)
    motor := n/2+1
    if n%2 == 0 {
        motor = n/2
    }
    fmt.Println(motor)
}
```

Output:

```
PS C:\Users\HP\Desktop\tugas> go run "c:\Users\HP\Desktop\tugas\modul 9\latsol1.go"
1
PS C:\Users\HP\Desktop\tugas> go run "c:\Users\HP\Desktop\tugas\modul 9\latsol1.go"
25
13
PS C:\Users\HP\Desktop\tugas> go run "c:\Users\HP\Desktop\tugas\modul 9\latsol1.go"
9
5
```

Deskripsi Program:

- Program ini digunakan untuk menentukan jumlah motor yang diperlukan seseorang untuk melakukan touring. Satu motor hanya cukup untuk dua orang, yaitu satu pengemudi dan satu tumpangan. Setiap motor diprioritaskan untuk terisi dua orang.

- Mendeklarasikan dua variabel yaitu `n` dengan tipe data integer dan `motor` dengan tipe data integer
- Intruksi `Scan` digunakan untuk mendeklarasikan input `n`
- Menggunakan paradigma percabangan yaitu `if-then` :
if sebagai kondisi atau syarat = jika `n` habis dibagi 2
then (`{ }`) sebagai aksi = maka `n/2`
- Intruksi `Println` digunakan untuk mendeklarasikan output `motor`, jika output tidak sesuai syarat maka akan menghasilkan output kebalikannya (`n/2+1`)

2. Latihan 2

Buatlah suatu program dengan menggunakan bahasa pemrograman Go, yang mana program digunakan untuk menentukan apakah bilangan yang diberikan adalah bilangan genap negatif atau bukan.

Masukan terdiri dari suatu bilangan bulat.

Keluaran terdiri dari suatu teks yang menyatakan bilangan adalah "genap negatif" atau "bukan".

Contoh masukan dan keluaran

NO	Masukan	Keluaran
1	10	bukan
2	-4	genap negatif
3	0	bukan
4	-2	genap negative

Source Code:

```
package main

import "fmt"

func main(){
    var n int
    fmt.Scan(&n)
    teks := "bukan"
    if n<0 && n%2 == 0 {
        teks = "genap negatif"
```

```
}  
    fmt.Println(teks)  
  
}
```

Output:

```
PS C:\Users\HP\Desktop\tugas> go run "c:\Users\HP\Desktop\tugas\modul 9\latsol2.go"  
10  
bukan  
PS C:\Users\HP\Desktop\tugas> go run "c:\Users\HP\Desktop\tugas\modul 9\latsol2.go"  
-4  
genap negatif  
PS C:\Users\HP\Desktop\tugas> go run "c:\Users\HP\Desktop\tugas\modul 9\latsol2.go"  
0  
bukan  
PS C:\Users\HP\Desktop\tugas> go run "c:\Users\HP\Desktop\tugas\modul 9\latsol2.go"  
-2  
genap negatif
```

Deskripsi Program:

- Program ini digunakan untuk menentukan apakah bilangan yang diberikan adalah bilangan genap negatif atau bukan.
- Mendeklarasikan dua variabel yaitu **n** dengan tipe data integer dan **teks** dengan tipe data string
- Intruksi Scan digunakan untuk mendeklarasikan input **n**
- Menggunakan paradigma percabangan yaitu if-then :
if sebagai kondisi atau syarat = jika n kurang dari 0 dan n habis dibagi 2
then ({}) sebagai aksi = maka genap negatif
- Intruksi Println digunakan untuk mendeklarasikan output **teks**, jika output tidak sesuai syarat maka akan menghasilkan output kebalikannya ("bukan")

3. Latihan 3

Buatlah program yang digunakan untuk menentukan suatu bilangan adalah faktor dari bilangan yang lain. Suatu bilangan a adalah faktor dari b apabila bilangan a habis membagi bilangan b.

Masukan terdiri dari dua bilangan bulat positif x dan y.

Keluaran terdiri dari dua baris boolean, yang mana baris pertama adalah boolean yang menyatakan x adalah faktor dari y. Baris kedua adalah boolean yang menyatakan y adalah faktor dari x.

Contoh masukan dan keluaran:

NO	Masukan	Keluaran
1	10 5	false true
2	3 21	true false
3	4 4	true true

Source Code:

```
package main

import "fmt"

func main(){
    var x, y int
    var faktorX, faktorY bool
    fmt.Scan(&x, &y)

    if y%x == 0 {
        faktorX = true
    }
    if x%y == 0 {
        faktorY = true
    }

    fmt.Println(faktorX)
    fmt.Print(faktorY)
}
```

Output:

```

10 5
false
true
PS C:\Users\HP\Desktop\tugas> go run "c:\Users\HP\Desktop\tugas\modul 9\latsol3.go"
3 21
true
false
PS C:\Users\HP\Desktop\tugas> go run "c:\Users\HP\Desktop\tugas\modul 9\latsol3.go"
4 4
true
true
PS C:\Users\HP\Desktop\tugas> 

```

Deskripsi Program:

- Program ini digunakan untuk menentukan suatu bilangan adalah faktor dari bilangan yang lain. Suatu bilangan a adalah faktor dari b apabila bilangan a habis membagi bilangan b.
- Mendeklarasikan empat variabel yaitu `x,y` dengan tipe data integer dan `faktorX,faktorY` dengan tipe data boolean
- Intruksi Scan digunakan untuk mendeklarasikan input `x, y`
- Menggunakan paradigma percabangan yaitu if-then :
 - if pertama sebagai kondisi atau syarat = jika y habis dibagi x
then ({}) sebagai aksi = maka true
 - if kedua sebagai kondisi atau syarat = jika x habis dibagi y
then ({}) sebagai aksi = maka true
- Intruksi Println digunakan untuk mendaklarasikan output `faktorX` dan `factor Y`, jika output tidak sesuai syarat maka akan menghasilkan output kebalikannya (false)

DAFTAR PUSTAKA

Golang Program, Golang If...Else...Else If Statements

(<https://www.golangprograms.com/golang-if-else-statements.html>)

Novalagung(2024), A.13. Seleksi Kondisi

(<https://dasarpemrogramangolang.novalagung.com/A-seleksi-kondisi.html>,2024)

Akankah Velida(2022), Pernyataan If/Else dan Kondisional dalam Go

(<https://dev.to/willvelida/ifelse-and-conditional-statements-in-go-2k9>,2024)