

LAPORAN PRAKTIKUM
ALGORITMA DAN PEMROGRAMAN 1
MODUL IX
“IF-THEN”



DISUSUN OLEH:
FEROS PEDROSA VALENTINO
103112400055
S1 IF-12-01

DOSEN:
Yohani Setiya Rafika Nur, M. Kom.

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

DASAR TEORI

Percabangan:

- Setiap baris instruksi dieksekusi satu persatu
- Adanya instruksi bersyarat (kondisi). Instruksi dieksekusi apabila memenuhi kondisi atau syarat tertentu
- Konstruksi dalam algoritma yang memungkinkan kita untuk melakukan pilihan instruksi yang berbeda-beda sesuai dengan kondisi yang dihadapi

Kondisi:

- Ekspresi yang bernilai TRUE atau FALSE (Boolean) Operasi Perbandingan atau Logika

Aksi:

- Kumpulan instruksi/ekspresi yang akan dieksekusi apabila kondisi bernilai TRUE
- Antara kondisi dan aksi terdapat suatu hubungan berpasangan

If expression akan sering digunakan dalam membuat program. If expression digunakan untuk logika pengkondisian (percabangan). If expression akan mengeksekusi kode di dalamnya apabila kondisi yang ditentukan bernilai benar (true). If expression biasa digunakan bersamaan dengan operator perbandingan untuk mendapatkan nilai boolean.

Selain if expression, ada juga else expression. Jika if expression mengeksekusi blok kode apabila kondisinya benar, maka else expression adalah kebalikan dari if expression. Else expression akan mengeksekusi blok kode di dalamnya apabila kondisinya bernilai salah. Else expression tidak bisa berdiri sendiri. Ia harus dipasangkan dengan if expression.

Karakteristik If-Then

Penulisan struktur kontrol percabangan dengan menggunakan if-then pada dasarnya terdiri dari dua bagian, yaitu:

1. Kondisi, yaitu sesuatu syarat atau ketentuan dari suatu percabangan. Kondisi ini harus bernilai boolean, baik itu variabel ataupun operasi tipe data.
2. Aksi, yaitu kumpulan instruksi yang akan dilakukan apabila kondisi terpenuhi atau bernilai true. Artinya baris aksi tidak secara default dieksekusi, tetapi ada syarat yang harus terpenuhi terlebih dahulu sehingga aksi dapat dieksekusi.

CONTOH SOAL

1. Coso1

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:\semester1_alpro1\alpro1_week9> go run "c:\semester1_alpro1\alpro1_week9\coso1\coso1.go"
10
10
• PS C:\semester1_alpro1\alpro1_week9> go run "c:\semester1_alpro1\alpro1_week9\coso1\coso1.go"
-3
3
• PS C:\semester1_alpro1\alpro1_week9> go run "c:\semester1_alpro1\alpro1_week9\coso1\coso1.go"
5
5
• PS C:\semester1_alpro1\alpro1_week9> go run "c:\semester1_alpro1\alpro1_week9\coso1\coso1.go"
0
0
• PS C:\semester1_alpro1\alpro1_week9> go run "c:\semester1_alpro1\alpro1_week9\coso1\coso1.go"
-9999
9999
```

Deskripsi program:

Program di atas merupakan program sederhana dalam bahasa Go yang dirancang untuk menerima input berupa bilangan bulat dan selanjutnya mencetak nilai absolut dari bilangan tersebut. Program dimulai dengan mendeklarasikan variabel bilangan sebagai tipe data integer. Kemudian, program akan mengambil masukan dari pengguna dengan menggunakan fungsi `fmt.Scan` dan menyimpannya ke variabel yang sesuai. Selanjutnya, diperiksa apakah nilai bilangan tersebut lebih kecil dari nol. Apabila benar, nilai bilangan akan diubah menjadi positif dengan dikalikan dengan `-1`. Terakhir, program mencetak nilai bilangan setelah diubah menjadi nilai absolutnya.

2. Coso2

Source code:

```
package main

import "fmt"
```

```

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

```

Output:

```

PS C:\semester1_alpro1\alpro1_week9> go run "c:\semester1_alpro1\alpro1_week9\coso2\coso2.go"
10
positif
PS C:\semester1_alpro1\alpro1_week9> go run "c:\semester1_alpro1\alpro1_week9\coso2\coso2.go"
-3
bukan positif
PS C:\semester1_alpro1\alpro1_week9> go run "c:\semester1_alpro1\alpro1_week9\coso2\coso2.go"
5
positif
PS C:\semester1_alpro1\alpro1_week9> go run "c:\semester1_alpro1\alpro1_week9\coso2\coso2.go"
0
bukan positif

```

Deskripsi program:

Program ini merupakan program sederhana dengan bahasa pemrograman Go untuk menentukan apakah bilangan bulat yang dimasukkan pengguna termasuk bilangan positif atau bukan positif. Program dimulai dengan mendeklarasikan variabel *bilangan* yang bertipe integer untuk menyimpan angka yang diinput, dan variabel *teks* yang bertipe string untuk menyimpan hasil evaluasinya. Program akan membaca input pengguna dengan fungsi `fmt.Scan` dan menetapkan nilai awal dari variabel *teks* sebagai "bukan positif". Setelah itu, program akan mengecek apakah nilai *bilangan* tersebut lebih besar dari nol. Jika itu begitu, *teks* akan diubah menjadi "positif". Pada akhirnya, program akan mencetak nilai dari variabel *teks*, yang berisi hasil evaluasi apakah bilangan tersebut positif atau bukan positif.

3. Coso3

Source code:

```

package main

import "fmt"

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

```

Output:

```
PS C:\semester1_alpro1\alpro1_week9> go run "c:\semester1_alpro1\alpro1_week9\coso3\coso3.go"
10
false
PS C:\semester1_alpro1\alpro1_week9> go run "c:\semester1_alpro1\alpro1_week9\coso3\coso3.go"
-3
false
PS C:\semester1_alpro1\alpro1_week9> go run "c:\semester1_alpro1\alpro1_week9\coso3\coso3.go"
-4
true
PS C:\semester1_alpro1\alpro1_week9> go run "c:\semester1_alpro1\alpro1_week9\coso3\coso3.go"
0
false
PS C:\semester1_alpro1\alpro1_week9> go run "c:\semester1_alpro1\alpro1_week9\coso3\coso3.go"
-2
true
```

Deskripsi program:

Program di atas merupakan program sederhana yang ditulis dalam bahasa Go untuk mengevaluasi apakah bilangan bulat yang dimasukkan oleh pengguna termasuk dalam kategori bilangan genap negatif. Program diawali dengan mendeklarasikan variabel bilangan yang berjenis integer untuk menampung input angka, serta variabel hasil yang berjenis boolean untuk menyimpan hasil evaluasi logika. Program ini melakukan pembacaan input bilangan langsung dari pengguna menggunakan `fmt.Scan`. Lalu, ekspresi logika apakah $\text{bilangan} \% 2 == 0$ (modulo atau dilambangkan dengan simbol `%`) dan apakah bilangan kurang dari 0 dievaluasi. Pengecekan ini memverifikasi apakah bilangan tersebut genap (dengan sisa pembagian 0) dan negatif. Evaluasi tersebut telah disimpan di dalam variabel hasil. Pada akhirnya, program akan mencetak nilai hasil yang akan menjadi `true` jika bilangan genap negatif, dan `false` jika bukan genap negatif.

Catatan: Baris komentar dalam program menunjukkan cara penulisan lain yang dapat menghasilkan hasil yang serupa.

LATIHAN SOAL

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 menyatakan jumlah orang yang akan melakukan touring.

Keluaran berupa jumlah motor yang diperlukan peserta touring.

Source code:

```
package main

import "fmt"

func main() {
    var jumlahOrang, jumlahMotor int
    fmt.Print("Masukkan jumlah orang yang akan touring: ")
    fmt.Scan(&jumlahOrang)

    if jumlahOrang%2 == 0 {
        jumlahMotor = jumlahOrang / 2
    } else {
        jumlahMotor = (jumlahOrang / 2) + 1
    }

    fmt.Println("Jumlah motor yang diperlukan:", jumlahMotor)
}
```

Output:

```
PS C:\semester1_alpro1\alpro1_week9> go run "c:\semester1_alpro1\alpro1_week9\latsol1\latsol1.go"
Masukkan jumlah orang yang akan touring: 10
Jumlah motor yang diperlukan: 5
PS C:\semester1_alpro1\alpro1_week9> go run "c:\semester1_alpro1\alpro1_week9\latsol1\latsol1.go"
Masukkan jumlah orang yang akan touring: 1
Jumlah motor yang diperlukan: 1
PS C:\semester1_alpro1\alpro1_week9> go run "c:\semester1_alpro1\alpro1_week9\latsol1\latsol1.go"
Masukkan jumlah orang yang akan touring: 25
Jumlah motor yang diperlukan: 13
PS C:\semester1_alpro1\alpro1_week9> go run "c:\semester1_alpro1\alpro1_week9\latsol1\latsol1.go"
Masukkan jumlah orang yang akan touring: 9
Jumlah motor yang diperlukan: 5
```

Deskripsi program:

Program di atas adalah program yang ditulis dalam bahasa Go yang berfungsi untuk menghitung jumlah motor yang diperlukan untuk kegiatan touring berdasarkan jumlah peserta. Program diawali dengan meminta input berupa jumlah orang yang akan bergabung dalam tur melalui variabel `jumlahOrang` dengan jenis integer. Setelah itu, langkah menghitung jumlah motor dilakukan dengan memperhitungkan apakah jumlah peserta berada dalam angka genap atau ganjil.

Apabila jumlah peserta adalah bilangan genap (ditetapkan dengan syarat `jumlahOrang%2 == 0`), maka setiap motor bisa mengangkut dua orang. Oleh sebab itu, jumlah motor dihitung dengan membagi jumlah orang dengan 2. Akan

tetapi, bila jumlah peserta berjumlah ganjil, diperlukan satu motor ekstra untuk membawa satu orang yang tersisa. Dalam situasi ini, jumlah motor dikalkulasikan dengan rumus $(\text{jumlahOrang} / 2)$ ditambah 1.

Jumlah motor yang diperlukan akan ditampilkan kepada pengguna setelah perhitungan selesai. Program ini memastikan semua peserta tur dapat diakomodasi dengan menggunakan motor secara maksimal, baik ketika jumlah penumpang genap maupun ganjil.

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".

Source code:

```
package main

import "fmt"

func main() {
    var bilangan int
    fmt.Print("Masukkan bilangan: ")
    fmt.Scan(&bilangan)

    if bilangan < 0 && bilangan%2 == 0 {
        fmt.Println("genap negatif")
    } else {
        fmt.Println("bukan")
    }
}
```

Output:

```
PS C:\semester1_alpro1\alpro1_week9> go run "c:\semester1_alpro1\alpro1_week9\latsol2\latsol2.go"
Masukkan bilangan: 10
bukan
PS C:\semester1_alpro1\alpro1_week9> go run "c:\semester1_alpro1\alpro1_week9\latsol2\latsol2.go"
Masukkan bilangan: -4
genap negatif
PS C:\semester1_alpro1\alpro1_week9> go run "c:\semester1_alpro1\alpro1_week9\latsol2\latsol2.go"
Masukkan bilangan: 0
bukan
PS C:\semester1_alpro1\alpro1_week9> go run "c:\semester1_alpro1\alpro1_week9\latsol2\latsol2.go"
Masukkan bilangan: -2
genap negatif
```

Deskripsi program:

Program di atas adalah program dalam bahasa Go yang digunakan untuk menentukan apakah suatu bilangan merupakan bilangan genap negatif. Program dimulai dengan meminta input dari pengguna berupa sebuah bilangan bulat, yang disimpan dalam variabel bilangan berjenis integer.

Setelah bilangan dimasukkan, program memeriksa dua kondisi menggunakan struktur if-else. Pertama program mengecek apakah bilangan tersebut kurang dari

nol. Kedua program mengecek apakah bilangan tersebut saat di modulus dengan dua hasilnya nol.

Jika kedua kondisi tersebut terpenuhi, maka program mencetak "genap negatif". Namun, jika salah satu atau kedua kondisi tidak terpenuhi, program akan mencetak "bukan".

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.

Source code:

```
package main

import "fmt"

func main() {
    var x, y int
    var isFactorXY, isFactorYX bool

    fmt.Print("Masukkan bilangan bulat positif x: ")
    fmt.Scan(&x)
    fmt.Print("Masukkan bilangan bulat positif y: ")
    fmt.Scan(&y)

    if y%x == 0 {
        isFactorXY = true
    } else {
        isFactorXY = false
    }

    if x%y == 0 {
        isFactorYX = true
    } else {
        isFactorYX = false
    }

    fmt.Println(isFactorXY)
    fmt.Println(isFactorYX)
}
```


Output:

```
PS C:\semester1_alpro1\alpro1_week9> go run "c:\semester1_alpro1\alpro1_week9\latsol3\latsol3.go"
Masukkan bilangan bulat positif x: 10
Masukkan bilangan bulat positif y: 5
false
true
PS C:\semester1_alpro1\alpro1_week9> go run "c:\semester1_alpro1\alpro1_week9\latsol3\latsol3.go"
Masukkan bilangan bulat positif x: 3
Masukkan bilangan bulat positif y: 21
true
false
PS C:\semester1_alpro1\alpro1_week9> go run "c:\semester1_alpro1\alpro1_week9\latsol3\latsol3.go"
Masukkan bilangan bulat positif x: 4
Masukkan bilangan bulat positif y: 4
true
true
```

Deskripsi program:

Program di atas adalah program sederhana dalam bahasa Go yang digunakan untuk menentukan apakah suatu bilangan merupakan faktor dari bilangan lainnya. Program dimulai dengan mendeklarasikan dua variabel bilangan bulat positif x dan y berjenis integer untuk menyimpan input dari pengguna, serta dua variabel boolean yaitu `isFactorXY` dan `isFactorYX` untuk menyimpan hasil evaluasi faktor. Pertama program meminta pengguna untuk memasukkan dua bilangan bulat positif, yaitu x dan y . Kemudian, program memeriksa dua kondisi:

1. Apakah x merupakan faktor dari y dengan mengecek apakah hasil sisa pembagian y dengan x ($y \% x$) sama dengan nol. Jika ya, maka `isFactorXY` diatur menjadi `true`; jika tidak, diatur menjadi `false`.
2. Apakah y merupakan faktor dari x dengan mengecek apakah hasil sisa pembagian x dengan y ($x \% y$) sama dengan nol. Jika ya, maka `isFactorYX` diatur menjadi `true`; jika tidak, diatur menjadi `false`.

Akhirnya, program mencetak nilai dari `isFactorXY` dan `isFactorYX` dalam dua baris terpisah, yang masing-masing menunjukkan apakah x adalah faktor dari y dan apakah y adalah faktor dari x .

DAFTAR PUSTAKA

A.13. Seleksi Kondisi. (n.d.). Retrieved from
<https://dasarpemrogramangolang.novalagung.com/A-seleksi-kondisi.html>

#16: If Expression - Belajar Golang Dari Dasar. (2022). Retrieved from
<https://blog.ruangdeveloper.com/golang-if-expression/>