

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



Disusun oleh :
NAMA : Felix Pedrosa Valentino
NIM : 103112400056
S1 IF – 12 - 01

Dosen Pengampu :
Yohani Setiya Rafika Nur, M. Kom.

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

DASAR TEORI

Paradigma Percabangan

Sebelumnya telah dipelajari bahwa setiap baris kode program akan dieksekusi satu persatu secara sekuensial. Artinya kode program dari baris ke-1 hingga baris terakhir akan dieksekusi satu persatu. Bagaimana jika kita ingin baris kode program yang dieksekusi itu berdasarkan syarat atau suatu ketentuan tertentu? Misalnya baris pertama dan terakhir saja yang dijalankan. Maka di dalam pemrograman hal tersebut mungkin untuk dilakukan, struktur kontrol yang digunakan adalah if-then.

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

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.) Contoh Soal 1

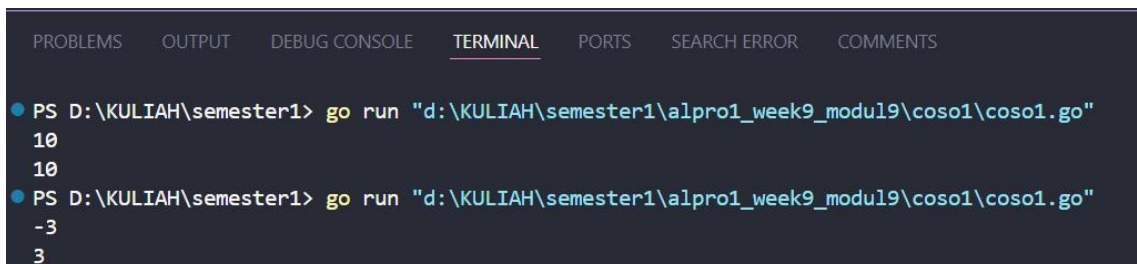
Source Code :

```
package main

import "fmt"

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

Output :



```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  SEARCH ERROR  COMMENTS

PS D:\KULIAH\semester1> go run "d:\KULIAH\semester1\alpro1_week9_modul9\coso1\coso1.go"
10
10
PS D:\KULIAH\semester1> go run "d:\KULIAH\semester1\alpro1_week9_modul9\coso1\coso1.go"
-3
3
```

Deskripsi Program :

Program di atas berfungsi menerima input bilangan bulat dari pengguna dan selanjutnya menampilkan nilai absolutnya. Program diawali dengan mendeklarasikan variabel bernama "bilangan" dengan tipe data integer yang akan digunakan untuk menyimpan nilai yang dimasukkan oleh pengguna. Setelah pengguna memasukkan sebuah bilangan, program akan memeriksa apakah bilangan tersebut negatif. Jika bilangan bersifat negatif (kurang dari nol), program akan mengubahnya menjadi positif dengan mengalikan bilangan tersebut dengan -1. Kemudian, program akan menampilkan nilai absolut dari bilangan tersebut. Program ini dibuat dengan tujuan untuk memastikan bahwa hasil keluarannya selalu positif

atau nol, tanpa memperdulikan apakah input dari pengguna pada mulanya negatif atau positif.

2.) Contoh Soal 2

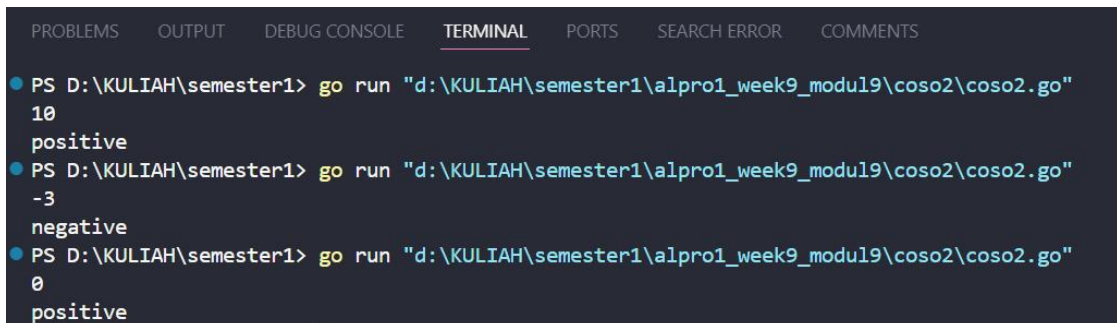
Source Code :

```
package main

import "fmt"

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

Output :



```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  SEARCH ERROR  COMMENTS
PS D:\KULIAH\semester1> go run "d:\KULIAH\semester1\alpro1_week9_modul9\coso2\coso2.go"
10
positive
PS D:\KULIAH\semester1> go run "d:\KULIAH\semester1\alpro1_week9_modul9\coso2\coso2.go"
-3
negative
PS D:\KULIAH\semester1> go run "d:\KULIAH\semester1\alpro1_week9_modul9\coso2\coso2.go"
0
positive
```

Deskripsi Program :

Program di atas merupakan program yang digunakan untuk menentukan apakah bilangan bulat yang dimasukkan oleh pengguna merupakan nilai positif atau negatif. Program ini dimulai dengan mendeklarasikan variabel *a* yang bertipe *int* untuk menyimpan bilangan yang akan diinput pengguna, serta variabel *teks* yang bertipe *string* untuk menyimpan hasil evaluasi ("positive" atau "negative"). Setelah pengguna memasukkan nilai, variabel *teks* akan diatur kembali ke nilai asal "negative". Setelah itu,

program akan memastikan apakah nilai bilangan a lebih besar atau sama dengan nol. Jika begitu, variabel teks ditukar dengan "positif". Setelah itu, program mencetak nilai teks untuk menunjukkan apakah bilangan yang dimasukkan oleh pengguna merupakan "positif" atau "negatif".

3.) Contoh Soal

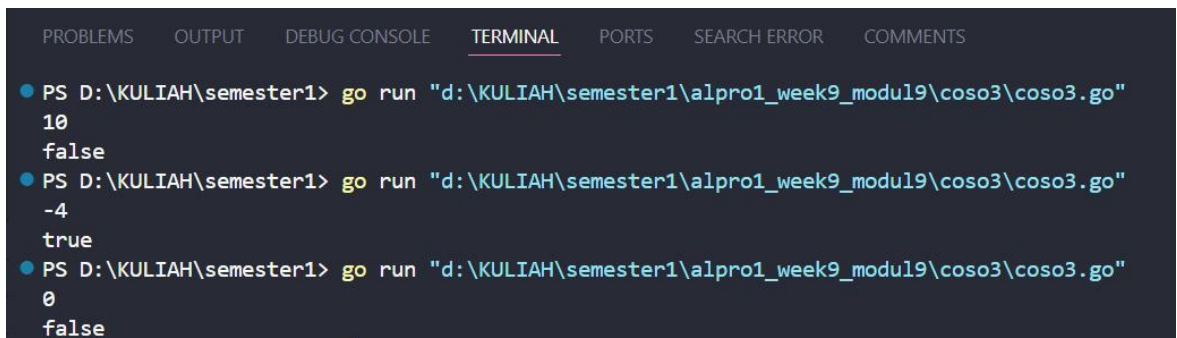
Source Code :

```
package main

import "fmt"

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

Output :



```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  SEARCH ERROR  COMMENTS

PS D:\KULIAH\semester1> go run "d:\KULIAH\semester1\alpro1_week9_modul9\coso3\coso3.go"
10
false
PS D:\KULIAH\semester1> go run "d:\KULIAH\semester1\alpro1_week9_modul9\coso3\coso3.go"
-4
true
PS D:\KULIAH\semester1> go run "d:\KULIAH\semester1\alpro1_week9_modul9\coso3\coso3.go"
0
false
```

Deskripsi Program :

Program ini berfungsi untuk memeriksa apakah angka bulat yang dimasukkan pengguna merupakan angka negatif dan juga genap. Program diawali dengan mendeklarasikan variabel a sebagai int untuk menampung input pengguna dan variabel hasil sebagai bool untuk menyimpan hasil

pengecekan kondisi. Setelah pengguna memasukkan nilai, program akan mengevaluasi apakah bilangan tersebut memenuhi dua syarat: bernilai negatif (kurang dari 0) dan genap (dapat dibagi habis dengan 2 atau hasil bagi 2 nya adalah 0). Hasil pengecekan ini telah disimpan dalam variabel yang bernama hasil. Apabila kedua kondisi terpenuhi, maka hasilnya akan true; jika tidak, hasilnya menjadi false. Setelah program berjalan, nilai hasil akan dicetak untuk menampilkan apakah bilangan yang dimasukkan oleh pengguna adalah negatif dan genap atau tidak.

LATIHAN SOAL

1.) Latihan Soal 1

Source code :

```
package main

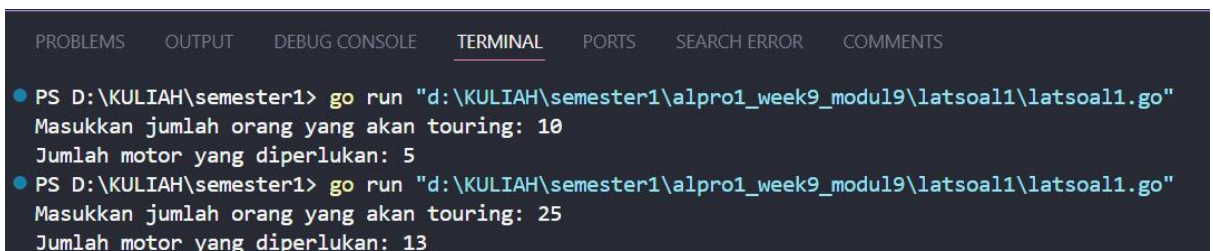
import "fmt"

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 D:\KULIAH\semester1> go run "d:\KULIAH\semester1\alpro1_week9_modul9\latsoal1\latsoal1.go"
Masukkan jumlah orang yang akan touring: 10
Jumlah motor yang diperlukan: 5
PS D:\KULIAH\semester1> go run "d:\KULIAH\semester1\alpro1_week9_modul9\latsoal1\latsoal1.go"
Masukkan jumlah orang yang akan touring: 25
Jumlah motor yang diperlukan: 13
```

Deskripsi Program :

Program ini akan menghitung jumlah motor yang diperlukan untuk membawa sekelompok orang dalam kegiatan touring, dengan asumsi bahwa setiap motor dapat membawa maksimal dua orang, yaitu satu pengemudi dan satu penumpang. Program dimulai dengan mendeklarasikan variabel `jumlahOrang`, `jumlahMotor` bertipe integer

untuk menyimpan input dari pengguna. Kemudian program meminta pengguna untuk memasukkan jumlah orang yang akan mengikuti touring. Setelah itu program memeriksa apakah jumlahOrang memiliki nilai genap atau ganjil. Apabila jumlah orang adalah genap, maka jumlah motor yang dibutuhkan adalah $\text{jumlahOrang} / 2$, karena setiap motor mampu mengangkut dua orang. Jika jumlah orang adalah ganjil, maka jumlah motor dihitung dengan rumus $(\text{jumlahOrang} / 2) + 1$, untuk memastikan ada satu motor tambahan yang membawa satu orang. Setelah itu, program menunjukkan total motor yang diperlukan (jumlahMotor).

2.) Latihan Soal 2

Source code :

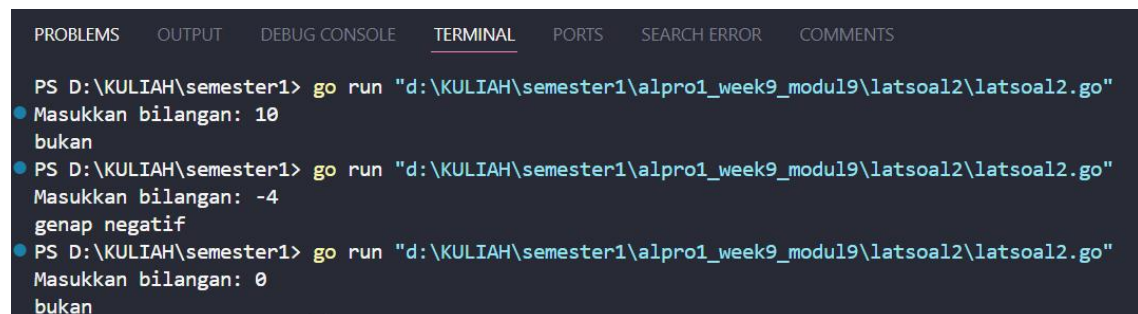
```
package main

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 :



```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  SEARCH ERROR  COMMENTS

PS D:\KULIAH\semester1> go run "d:\KULIAH\semester1\alpro1_week9_modul9\latsoal2\latsoal2.go"
Masukkan bilangan: 10
bukan
PS D:\KULIAH\semester1> go run "d:\KULIAH\semester1\alpro1_week9_modul9\latsoal2\latsoal2.go"
Masukkan bilangan: -4
genap negatif
PS D:\KULIAH\semester1> go run "d:\KULIAH\semester1\alpro1_week9_modul9\latsoal2\latsoal2.go"
Masukkan bilangan: 0
bukan
```

Deskripsi Program :

Program diatas bertujuan untuk menentukan apakah bilangan yang dimasukkan oleh pengguna adalah "genap negatif" atau "bukan". Program dimulai dengan mendeklarasikan variabel bilangan sebagai integer untuk menyimpan input dari pengguna. Setelah itu, program akan minta pengguna memasukkan satu bilangan bulat dan akan disimpan dalam variabel bilangan. Selanjutnya program akan mengecek dua kondisi : apakah bilangan tersebut negatif ($\text{bilangan} < 0$) dan apakah bilangan tersebut genap ($\text{bilangan} \% 2 == 0$). Apabila dua syarat terpenuhi, program akan mencetak "genap negatif", yang menandakan bahwa bilangan yang dimasukkan adalah bilangan genap dan negatif. Apabila salah satu atau kedua syarat tidak terpenuhi, program akan mencetak kata "bukan", yang menunjukkan bahwa bilangan tersebut tidak termasuk dalam kategori genap negatif. Program ini dapat membantu mengenali bilangan bulat tertentu yang memiliki atribut khusus, seperti bilangan genap dan negatif.

3.) Latihan Soal 3

Source code :

```
package main

import "fmt"

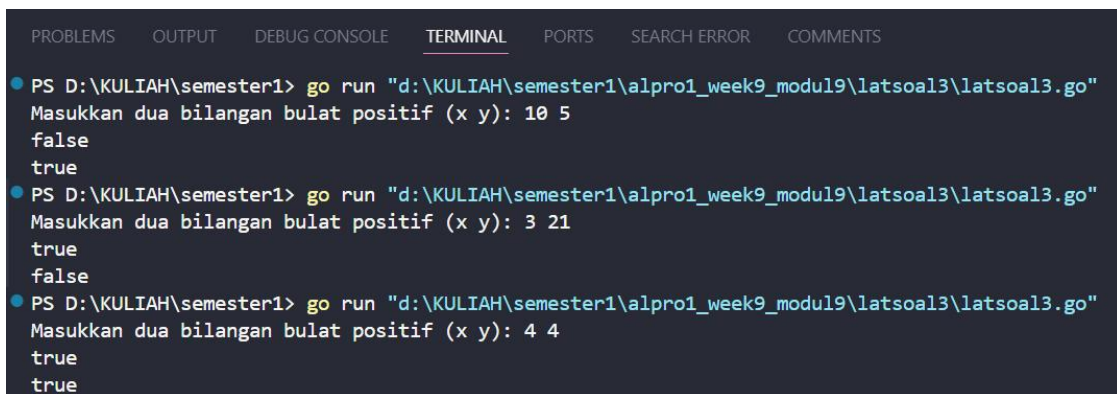
func main() {
    var x, y int
    var isFactorXY, isFactorYX bool
    fmt.Print("Masukkan dua bilangan bulat positif (x y): ")
    fmt.Scanln(&x, &y)

    // Menentukan apakah x adalah faktor dari y
    if y%x == 0 {
        isFactorXY = true
    } else {
        isFactorXY = false
    }

    // Menentukan apakah y adalah faktor dari x
    if x%y == 0 {
        isFactorYX = true
    }
```

```
} else {  
    isFactorYX = false  
}  
  
fmt.Println(isFactorXY)  
fmt.Println(isFactorYX)  
}
```

Output :



```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  SEARCH ERROR  COMMENTS  
● PS D:\KULIAH\semester1> go run "d:\KULIAH\semester1\alpro1_week9_modul9\latsoal3\latsoal3.go"  
Masukkan dua bilangan bulat positif (x y): 10 5  
false  
true  
● PS D:\KULIAH\semester1> go run "d:\KULIAH\semester1\alpro1_week9_modul9\latsoal3\latsoal3.go"  
Masukkan dua bilangan bulat positif (x y): 3 21  
true  
false  
● PS D:\KULIAH\semester1> go run "d:\KULIAH\semester1\alpro1_week9_modul9\latsoal3\latsoal3.go"  
Masukkan dua bilangan bulat positif (x y): 4 4  
true  
true
```

Deskripsi Program :

Program ini bertujuan untuk menentukan apakah bilangan bulat positif `x` dan `y` adalah faktor satu sama lain. Program dimulai dengan mendeklarasikan variabel `x`, `y` bertipe integer untuk menyimpan input dari pengguna dan variabel `isFactorXY`, `isFactorYX` bertipe boolean. Kemudian, program meminta input dari pengguna berupa dua bilangan bulat positif (`x` dan `y`) yang akan disimpan dalam variabel `x` dan `y`. Selanjutnya, program melakukan dua pengecekan untuk menentukan hubungan faktor antara kedua bilangan tersebut. Yang pertama, program akan menentukan apakah `x` merupakan faktor bagi `y` dengan menggunakan operator modulus (`y % x`). Apabila nilai `y` dapat terbagi secara utuh oleh `x` (hasil modulus adalah 0), maka status `isFactorXY` akan diubah menjadi `true`, yang menunjukkan bahwa `x` adalah faktor dari `y`. Kalau tidak, `isFactorXY` akan menjadi `false`.

Kemudian, program menentukan apakah `y` merupakan faktor dari `x` dengan menggunakan operator modulus (`x % y`). Apabila `x` habis dibagi oleh `y`, maka `isFactorYX` akan diatur sebagai `true`, menunjukkan bahwa `y` adalah faktor dari `x`. Jika tidak, maka nilai `isFactorYX` akan menjadi `false`.

Setelah itu, hasil pencetakan boolean `'isFactorXY'` dan `'isFactorYX'` akan ditampilkan dalam dua baris berbeda pada akhir program. Baris pertama berfungsi mengecek apakah `'x'` adalah faktor dari `'y'`, sementara baris kedua mengecek apakah `'y'` adalah faktor dari `'x'`. Program ini menghasilkan informasi yang jelas terkait kedua bilangan yang dimasukkan oleh pengguna.

DAFTAR PUSTAKA

(N.d.). Retrieved from https://telkomuniversityofficial-my.sharepoint.com/:p:/r/personal/gppras_telkomuniversity_ac_id/_layouts/15/Doc.aspx?sourcedoc=%7B11720F8E-4AD7-4ADC-BAEE-236F70E838CF%7D&file=%5BWeek%2009%5D%20CAK1BAB3-ALPRO_1.ppsx&action=edit&mobileredirect=true

Go if else. (n.d.). Retrieved from <https://www.programiz.com/golang/if-else>

Golang If...Else...Else If Statements. (n.d.). Retrieved from <https://www.golangprograms.com/golang-if-else-statements.html>

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