

**LAPORAN PRAKTIKUM
ALGORITMA DAN PEMROGRAMAN 2**

MODUL III

FUNGSI



Disusun Oleh :

ARIHQ RADHITYA PRADANA

2311102260

IF 11 06

Dosen Pengampu :

ABEDNEGO DWI SEPTIADI

PROGRAM STUDI S1 TEKNIK INFORMATIKA

FAKULTAS INFORMATIKA

TELKOM UNIVERSITY PURWOKERTO

2024

I. DASAR TEORI

Bahasa pemrograman Go, atau Golang, adalah bahasa yang dikembangkan oleh Google dengan tujuan untuk menciptakan sistem yang efisien, sederhana, dan mudah dipelajari. Salah satu konsep dasar dalam Golang adalah **fungsi**. Fungsi dalam Golang adalah blok kode yang dirancang untuk melakukan tugas tertentu; mereka menerima input, memprosesnya, dan mengembalikan output. Fungsi ini sangat penting karena memungkinkan pengembang untuk menulis kode yang modular dan terorganisir, sehingga memudahkan pemeliharaan dan pengembangan aplikasi yang lebih besar.

Golang mendukung berbagai jenis fungsi, termasuk fungsi biasa dan fungsi generik. Fungsi biasa mengambil parameter dengan tipe data tertentu dan mengembalikan hasil setelah pemrosesan. Namun, Golang memiliki keterbatasan dalam mendukung fungsi generik, yang dapat bekerja dengan berbagai tipe data. Keterbatasan ini dapat menyebabkan pengulangan kode dan mengurangi efisiensi selama pengembangan. Meskipun demikian, kelebihan Golang dalam hal kecepatan eksekusi dan kemudahan penggunaan menjadikannya pilihan populer di kalangan pengembang untuk membangun aplikasi berskala besar, terutama dalam konteks layanan web dan cloud computing. Dengan sintaks yang bersih dan dukungan untuk concurrency melalui goroutine, Golang memungkinkan pengembang untuk menciptakan aplikasi yang berkinerja tinggi dengan lebih cepat dan efisien.

II. GUIDED

Guided 1

```
package main

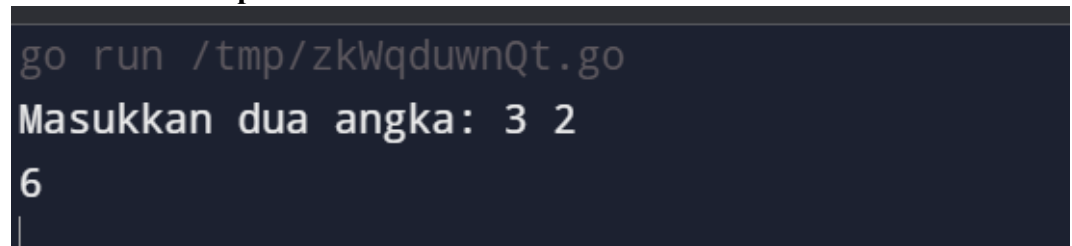
import "fmt"

func main() {
    var a, b int
    fmt.Print("Masukkan dua angka: ")
    fmt.Scan(&a, &b)
    fmt.Println(permutasi(a, b))
}

func faktorial(n int) int64 {
    var hasil int64 = 1
    for i := 1; i <= n; i++ {
        hasil *= int64(i)
    }
    return hasil
}

func permutasi(n, r int) int64 {
    if r > n {
        fmt.Println("r tidak boleh lebih besar dari n")
        return 0
    }
    return faktorial(n) / faktorial(n-r)
}
```

Screenshoot Output



```
go run /tmp/zkWqduwnQt.go
Masukkan dua angka: 3 2
6
|
```

Deskripsi Program

Kode di atas merupakan program Go yang menghitung nilai permutasi dari dua bilangan yang dimasukkan oleh pengguna. Program dimulai dengan fungsi `main()`, di mana pengguna diminta untuk memasukkan dua angka, yaitu `a` dan `b`, melalui input konsol menggunakan `fmt.Scan()`. Setelah kedua angka diinput, program memanggil fungsi `permutasi(a, b)`.

Fungsi `faktorial(n)` digunakan untuk menghitung faktorial dari angka `n`. Ia mendeklarasikan variabel `hasil` bertipe `int64` untuk menghindari overflow saat menghitung faktorial angka besar. Fungsi ini menghitung faktorial dengan mengalikan `hasil` dengan setiap angka dari 1 hingga `n`.

Fungsi `permutasi(n, r)` mengimplementasikan rumus permutasi $P(n, r) = \frac{n!}{(n-r)!}$. Sebelum perhitungan dilakukan, fungsi ini memeriksa apakah nilai `r` lebih besar dari `n`. Jika ya, program mencetak pesan kesalahan dan mengembalikan nilai `0`, karena permutasi tidak valid jika $r > n$. Jika tidak, fungsi akan menghitung permutasi dengan membagi faktorial `n` dengan faktorial `(n-r)`.

Hasil dari perhitungan permutasi ini kemudian dicetak di konsol oleh fungsi `main()`. Program ini memberikan pengguna cara sederhana untuk menghitung permutasi dua angka, dengan validasi input yang memastikan perhitungan permutasi hanya dilakukan jika input valid.

Guided 2

```
package main

import (
    "fmt"
)

func main() {
    balok()
}

func balok() {
    var panjang, lebar, tinggi float64

    fmt.Print("Panjang = ")
    fmt.Scanln(&panjang)

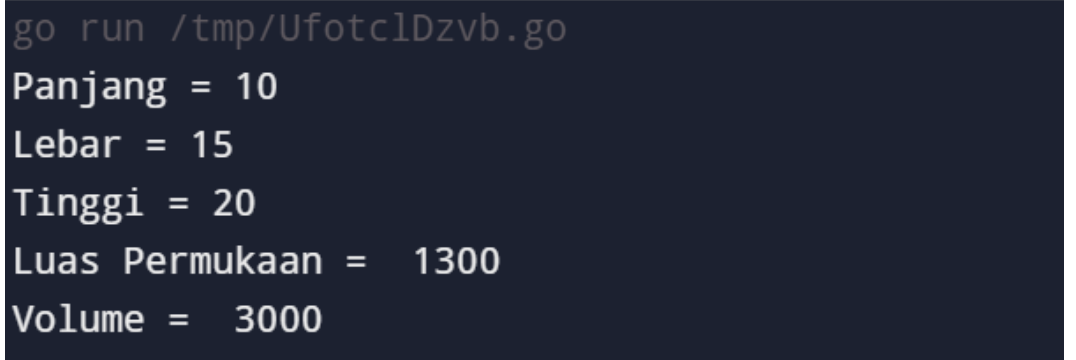
    fmt.Print("Lebar = ")
    fmt.Scanln(&lebar)

    fmt.Print("Tinggi = ")
    fmt.Scanln(&tinggi)

    luasPermukaan := 2 * (panjang*lebar + panjang*tinggi
+ lebar*tinggi)
    volume := panjang * lebar * tinggi

    fmt.Println("Luas Permukaan = ", luasPermukaan)
    fmt.Println("Volume = ", volume)
}
```

Screenshoot Output



```
go run /tmp/UfotclDzvb.go
Panjang = 10
Lebar = 15
Tinggi = 20
Luas Permukaan = 1300
Volume = 3000
```

Deskripsi Program

coding di atas merupakan sebuah program Go yang menghitung luas permukaan dan volume balok berdasarkan input dari pengguna. Berikut adalah penjelasan secara rinci dalam bentuk paragraf:

Program ini dimulai dengan mendefinisikan dua fungsi: ``main()`` dan ``balok()``. Fungsi ``main()`` adalah titik masuk utama dari program, yang memanggil fungsi ``balok()``. Dalam fungsi ``balok()``, program meminta pengguna untuk memasukkan nilai panjang, lebar, dan tinggi balok dengan menggunakan ``fmt.Scanln()`` untuk mengambil input dari pengguna. Setelah input diperoleh, program menghitung luas permukaan balok menggunakan rumus ``2 * (panjang * lebar + panjang * tinggi + lebar * tinggi)`` yang mewakili jumlah luas semua sisi balok. Selain itu, program juga menghitung volume balok dengan rumus ``panjang * lebar * tinggi``. Hasil perhitungan luas permukaan dan volume kemudian ditampilkan ke layar menggunakan ``fmt.Println()``, dengan format yang jelas dan mudah dipahami oleh pengguna.

Guided 3

```
package main

import "fmt"

var a, b, c, d int

// Fungsi untuk menghitung faktorial dari bilangan n
func faktorial(n int) int {
    hasil := 1
    // Loop untuk menghitung faktorial, dimulai dari 1
    hingga n
    for i := 1; i <= n; i++ {
        hasil = hasil * i
    }
    return hasil
}

// Fungsi untuk menghitung permutasi  $P(n, r) = n! / (n-r)!$ 
func permutasi(n, r int) int {
    return faktorial(n) / faktorial(n-r)
}

// Fungsi untuk menghitung kombinasi  $C(n, r) = n! / (r! * (n-r)!)$ 
func kombinasi(n, r int) int {
    return faktorial(n) / (faktorial(r) * faktorial(n-r))
}

func main() {

    fmt.Print("Masukkan input = ")
    fmt.Scan(&a, &b, &c, &d)

    if a >= c && b >= d {
        // Baris pertama: Permutasi dan Kombinasi a
        terhadap c
        fmt.Printf("%d, %d\n", permutasi(a, c),
            kombinasi(a, c))

        // Baris kedua: Permutasi dan Kombinasi b
        terhadap d
        fmt.Printf("%d, %d\n", permutasi(b, d),
            kombinasi(b, d))
    } else {
        fmt.Println("Syarat tidak terpenuhi: a harus
            >= c dan b harus >= d")
    }
}
```

Screenshoot Output

```
go run /tmp/QqdmWoyivw.go
Masukkan input = 5 10 3 10
60, 10
3628800, 1
```

Deskripsi Program

Program ini merupakan implementasi dalam bahasa Go yang bertujuan untuk menghitung **permutasi** dan **kombinasi** dari dua pasang bilangan input (a, b) dan (c, d). Pertama, pengguna diminta memasukkan empat angka: 'a', 'b', 'c', dan 'd'. Kemudian program memeriksa apakah nilai 'a' lebih besar atau sama dengan 'c' dan 'b' lebih besar atau sama dengan 'd'. Jika syarat ini terpenuhi, program akan menghitung permutasi dan kombinasi dari dua bilangan:

1. Permutasi 'P(a, c)' dan kombinasi 'C(a, c)'.
2. Permutasi 'P(b, d)' dan kombinasi 'C(b, d)'.

Permutasi dihitung menggunakan rumus $P(n, r) = n! / (n-r)!$, sementara kombinasi dihitung dengan rumus $C(n, r) = n! / (r! * (n-r)!)$. Faktorial 'n!' dihitung dengan fungsi 'faktorial(n)', yang mengalikan semua bilangan dari 1 hingga 'n'.

Jika nilai 'a' atau 'b' tidak memenuhi syarat ($a < c$ atau $b < d$), program akan menampilkan pesan bahwa syarat tidak terpenuhi dan tidak akan melakukan perhitungan.

Jadi, tujuan program ini adalah untuk menghitung dan menampilkan hasil permutasi dan kombinasi berdasarkan input pengguna, asalkan kondisi awal yang ditentukan terpenuhi.

III. UNGUIDED

Sourcecode

```
package main

import "fmt"

// Fungsi f(x) = x^2
func f(x int) int {
    return x * x
}

// Fungsi g(x) = x - 2
func g(x int) int {
    return x - 2
}

// Fungsi h(x) = x + 1
func h(x int) int {
    return x + 1
}

// Fungsi untuk menghitung (f o g o h) (x)
func fogoh(x int) int {
    return f(g(h(x)))
}

// Fungsi untuk menghitung (g o h o f) (x)
func gohof(x int) int {
    return g(h(f(x)))
}

// Fungsi untuk menghitung (h o f o g) (x)
func hofog(x int) int {
    return h(f(g(x)))
}

func main() {
    var a, b, c int
    // Masukkan input berupa 3 bilangan bulat
    fmt.Scan(&a, &b, &c)

    // Baris pertama: (f o g o h) (a)
    fogohResult := fogoh(a)
    fmt.Printf("(fogoh) (%d) = %d\n", a, fogohResult)

    // Baris kedua: (g o h o f) (b)
    gohofResult := gohof(b)
    fmt.Printf("(gohof) (%d) = %d\n", b, gohofResult)

    // Baris ketiga: (hofog) (c)
    hofogResult := hofog(c)
```

```
    fmt.Printf("(hofog) (%d) = %d\n", c, hofogResult)\n}\n}
```

Screenshoot Output

```
go run /tmp/NnNxx2L5iR.go\n7 2 10\n(fogoh)(7) = 36\n(gohof)(2) = 3\n(hofog)(10) = 65
```

Deskripsi Program

Kode di atas adalah program dalam bahasa Go yang mengimplementasikan tiga fungsi matematika: $f(x)$, $g(x)$, dan $h(x)$, serta menghitung komposisi dari fungsi-fungsi tersebut untuk tiga bilangan bulat yang dimasukkan oleh pengguna. Fungsi $f(x)$ mengembalikan kuadrat dari x (yaitu $f(x) = x^2$), sementara fungsi $g(x)$ mengurangi nilai x dengan 2 ($g(x) = x - 2$), dan fungsi $h(x)$ menambah nilai x dengan 1 ($h(x) = x + 1$). Program ini mendefinisikan tiga komposisi fungsi: $(f \circ g \circ h)(x)$, yang menghitung $f(g(h(x)))$; $(g \circ h \circ f)(x)$, yang menghitung $g(h(f(x)))$; dan $(h \circ f \circ g)(x)$, yang menghitung $h(f(g(x)))$. Dalam fungsi `main()`, pengguna diminta untuk memasukkan tiga bilangan bulat, `a`, `b`, dan `c`, yang kemudian digunakan untuk menghitung dan mencetak hasil dari setiap komposisi fungsi dalam format yang jelas, seperti `(fogoh)(a) = hasil`. Dengan demikian, program ini memungkinkan pengguna untuk melihat hasil dari komposisi fungsi matematika berdasarkan input yang diberikan.

unguided 2

```
package main

import (
    "fmt"
    "math"
)

func main() {
    var cx1, cy1, r1 int
    var cx2, cy2, r2 int
    var x, y int

    // Input untuk lingkaran 1
    fmt.Println("Masukkan data lingkaran 1 (format: x y radius):")
    fmt.Scan(&cx1, &cy1, &r1)

    // Input untuk lingkaran 2
    fmt.Println("Masukkan data lingkaran 2 (format: x y radius):")
    fmt.Scan(&cx2, &cy2, &r2)

    // Input untuk titik yang akan diperiksa
    fmt.Println("Masukkan koordinat titik (format: x y):")
    fmt.Scan(&x, &y)

    // Hitung jarak titik ke pusat lingkaran
    d1 := jarak(x, y, cx1, cy1)
    d2 := jarak(x, y, cx2, cy2)

    // Tentukan posisi titik
    if d1 <= float64(r1) && d2 <= float64(r2) {
        fmt.Printf("titik di bawah lingkaran 1 dan 2\n")
    } else if d1 <= float64(r1) {
        fmt.Printf("titik di bawah lingkaran 1\n")
    } else if d2 <= float64(r2) {
        fmt.Printf("titik di bawah lingkaran 2\n")
    } else {
        fmt.Printf("titik di luar lingkaran 1 dan 2\n")
    }
}

// Fungsi untuk menghitung jarak antara dua titik
func jarak(a, b, c, d int) float64 {
    return math.Sqrt(math.Pow(float64(a-c), 2) +
        math.Pow(float64(b-d), 2))
}
```

Screenshoot Output

```
Masukkan data lingkaran 1 (format: x y radius):  
1 1 5  
Masukkan data lingkaran 2 (format: x y radius):  
8 8 4  
Masukkan koordinat titik (format: x y):  
2 2  
titik di bawah lingkaran 1
```

Deskripsi Program

Program di atas merupakan implementasi dalam bahasa Go yang bertujuan untuk menentukan posisi suatu titik relatif terhadap dua lingkaran. Program meminta pengguna untuk memasukkan data dua lingkaran berupa koordinat pusat ('x' dan 'y') serta radius (jari-jari) masing-masing lingkaran. Selain itu, pengguna juga diminta untuk memasukkan koordinat suatu titik yang akan diperiksa.

Setelah data dimasukkan, program menghitung jarak antara titik tersebut dan pusat kedua lingkaran menggunakan rumus jarak Euclidean, yaitu $\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$. Hasil perhitungan jarak ini dibandingkan dengan radius lingkaran untuk menentukan apakah titik berada di dalam (di bawah) salah satu, kedua, atau di luar lingkaran.

Program kemudian mencetak hasil yang relevan: jika titik berada di dalam kedua lingkaran, hasilnya adalah "titik di bawah lingkaran 1 dan 2". Jika hanya berada di salah satu lingkaran, program akan menampilkan "titik di bawah lingkaran 1" atau "titik di bawah lingkaran 2". Jika titik berada di luar kedua lingkaran, output-nya adalah "titik di luar lingkaran 1 dan 2".