

LAPORAN PRAKTIKUM
ALGORITMA DAN PEMROGRAMAN 2
MODUL 3
“FUNGSI”



DISUSUN OLEH:
SHEILA STEPHANIE ANINDYA
103112400086
S1 IF-12-01
S1 TEKNIK INFORMATIKA
TELKOM UNIVERSITY PURWOKERTO
2025

I. DASAR TEORI

A. Fungsi

Dalam konteks pemrograman, fungsi adalah sekumpulan blok kode yang dibungkus dengan nama tertentu. Penerapan fungsi yang tepat akan menjadikan kode lebih modular dan juga *dry* (singkatan dari *don't repeat yourself*) yang artinya tidak perlu menuliskan banyak kode untuk kegunaan yang sama berulang kali. Cukup deklarasikan sekali saja blok kode sebagai suatu fungsi, lalu panggil sesuai kebutuhan.

Oleh karena itu, fungsi selalu menghasilkan/mengembalikan nilai. Suatu subprogram dikatakan fungsi apabila:

1. Ada deklarasi tipe nilai yang dikembalikan, dan
2. Terdapat kata kunci `return` dalam badan subprogram.

Maka fungsi digunakan jika suatu nilai biasanya diperlukan, seperti:

- Assignment nilai ke suatu variabel
- Bagian dari ekspresi
- Bagian dari argumen suatu subprogram, dsb.

Karena itu selalu pilih nama fungsi yang menggambarkan nilai, seperti kata benda dan kata sifat. Contoh nama-nama fungsi : median, rerata, nilaiTerbesar, ketemu, selesai,

B. Penerapan Fungsi

Fungsi main() merupakan fungsi utama pada program Go, yang akan dieksekusi ketika program dijalankan. Selain fungsi `main()`, terdapat fungsi lainnya dalam Go Language. Untuk mengetahuinya dengan menuliskan keyword `func` kemudian diikuti nama fungsi, lalu kurung `()` (yang bisa diisi parameter), dan diakhiri dengan kurung kurawal untuk membungkus blok kode.

Parameter merupakan variabel yang menempel di fungsi yang nilainya ditentukan saat pemanggilan fungsi tersebut. Parameter sifatnya opsional, suatu fungsi bisa tidak memiliki parameter, atau bisa saja memiliki satu atau banyak parameter (tergantung kebutuhan).

Data yang digunakan sebagai value parameter saat pemanggilan fungsi biasa disebut dengan argument parameter (atau argument).

C. Fungsi Sebagai Parameter

Dalam pemrograman, fungsi dengan parameter memungkinkan kita untuk memberikan nilai-nilai input ke dalam fungsi sehingga bisa melakukan operasi yang lebih spesifik dan dinamis. Parameter ini berperan sebagai penampung nilai-nilai tersebut.

- **Kegunaan Parameter:**

- **Dinamisitas:** Dengan parameter, sebuah fungsi menjadi fleksibel dan dapat menangani berbagai nilai input.
- **Re-usability:** Fungsi dengan parameter dapat digunakan kembali dengan mudah untuk berbagai kasus dengan mengubah nilai argumennya.
- **Modularitas:** Parameter membantu memisahkan logic kode dan membuatnya lebih terstruktur.

II. GUIDED

1. Source Code:

```
package main

import "fmt"

func main() { //main redeclared in this block
    var a, b int
    fmt.Scan(&a, &b)

    if a >= b {
        fmt.Println(permutasi(a, b))
    } else {
        fmt.Println(permutasi(b, a))
    }
}

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

func permutasi(n, r int) int {
    if r > n {
        return 0
    }
    return faktorial(n) / faktorial(n-r)
}
```

Output:

```
PS C:\Sheila Stephanie Anindya\SMT 2\Alpro 2\code> go run "c:\Sheila
5 3
60
PS C:\Sheila Stephanie Anindya\SMT 2\Alpro 2\code> █
```

Penjelasan :

Program ini menghitung permutasi dengan rumus $P(n, r) = \frac{n!}{(n-r)!}$ Dimana n tidak boleh lebih kecil dari r. Jika r lebih besar dari n, hasilnya akan 0.

2. Source Code:

```
package main

import "fmt"

// konversi suhu dari celcius ke fahrenheit
func celsiusToFahrenheit(celsius float64) float64 {
    return (9.0/5.0)*celsius + 32
}

func main() {
    var N int
    fmt.Print("Masukkan jumlah data: ")
    _, err := fmt.Scan(&N)
    if err != nil || N <= 0 {
        fmt.Println("Input tidak valid, pastikan memasukkan angka positif.")
        return
    }
    temperatures := make([]float64, N)
    // Membaca suhu dalam Celsius
```

```

fmt.Println("Masukkan suhu dalam Celsius:")
for i := 0; i < N; i++ {
    _, err := fmt.Scan(&temperatures[i])
    if err != nil {
        fmt.Println("Input tidak valid, pastikan memasukkan angka.")
        return
    }
}

// Mengonversi ke Fahrenheit dan mencetak hasil
fmt.Println("Suhu dalam Fahrenheit:")
for _, temp := range temperatures {
    fmt.Printf("%.2f\n", celsiusToFahrenheit(temp))
}
}

```

Output:

```

PS C:\Sheila Stephanie Anindya\SMT 2\Alpro 2\code> go run "c:\Sheila Stephanie Anindya
Masukkan jumlah data: 3
Masukkan suhu dalam Celsius:
16
25
32
Suhu dalam Fahrenheit:
60.80
77.00
89.60
PS C:\Sheila Stephanie Anindya\SMT 2\Alpro 2\code>

```

Penjelasan :

Program mengkonversi suhu dari celcius ke Fahrenheit.

3. Source Code:

```
package main
```

```
import (  
    "fmt"  
    "math"  
)  
  
// Fungsi untuk menghitung luas permukaan tabung  
func luasPermukaanTabung(r, t float64) float64 {  
    return 2 * math.Pi * r * (r + t)  
}  
  
// Fungsi untuk menghitung volume tabung  
func volumeTabung(r, t float64) float64 {  
    return math.Pi * math.Pow(r, 2) * t  
}  
  
func main() {  
    var r, t float64  
  
    // Input jari-jari dan tinggi tabung dengan validasi  
    fmt.Print("Masukkan jari-jari tabung: ")  
    _, errR := fmt.Scan(&r)  
    fmt.Print("Masukkan tinggi tabung: ")  
    _, errT := fmt.Scan(&t)  
  
    // Memeriksa apakah input valid  
    if errR != nil || errT != nil {  
        fmt.Println("Input tidak valid! Harap masukkan angka yang  
benar.")  
        return  
    }  
}
```

```

// Memeriksa apakah jari-jari dan tinggi bernilai positif
if r <= 0 || t <= 0 {
    fmt.Println("Jari-jari dan tinggi tabung harus lebih dari nol.")
    return
}

// Menghitung luas permukaan dan volume
luas := luasPermukaanTabung(r, t)
volume := volumeTabung(r, t)

// Menampilkan hasil
fmt.Println("=====")
fmt.Printf("Luas Permukaan Tabung: %.2f satuan²\n", luas)
fmt.Printf("Volume Tabung: %.2f satuan³\n", volume)
fmt.Println("=====")
}

```

Output:

```

PS C:\Sheila Stephanie Anindya\SMT 2\Alpro 2\code> go run "c:\Sheila Stephanie Anindya\SMT 2\Alpro 2\code\main.go"
Masukkan jari-jari tabung: 14
Masukkan tinggi tabung: 30
=====
Luas Permukaan Tabung: 3870.44 satuan²
Volume Tabung: 18472.56 satuan³
=====
PS C:\Sheila Stephanie Anindya\SMT 2\Alpro 2\code>

```

Penjelasan :

Programnya menghitung luas permukaan tabung dengan rumus $2\pi r(r + t)$

dan volume tabung dengan rumus $\pi r^2 t$.

III. UNGUIDED

1. Source Code:

```
package main

import "fmt"

func main() {
    var a, b, c, d, p1, p2, c1, c2 int
    fmt.Scan(&a, &b, &c, &d)

    p1 = permutasi(a, c)
    c1 = combination(a, c)

    p2 = permutasi(b, d)
    c2 = combination(b, d)

    fmt.Println(p1, c1)
    fmt.Println(p2, c2)
}

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

func permutasi(n, r int) int {
```

```

    if r > n {
        return 0
    }
    return faktorial(n) / faktorial(n-r)
}

func combination(n, r int) int {
    if r > n {
        return 0
    }
    return faktorial(n) / (faktorial(r) * faktorial(n-r))
}

```

Output:

```

PS C:\Sheila Stephanie Anindya\SMT 2\Alpro 2\code> go run "c:\Sheila
5 10 3 10
60 10
3628800 1
PS C:\Sheila Stephanie Anindya\SMT 2\Alpro 2\code>

```

Penjelasan : Program ini menghitung permutasi dengan rumus $P(n, r) =$

$\frac{n!}{(n-r)!}$ dimana n tidak boleh lebih kecil dari r. Jika r lebih besar dari n, hasilnya

akan 0. Serta menghitung kombinasi dengan rumus $C(n, r) = \frac{n!}{r! (n-r)!}$.

2. Source Code:

```
package main

import "fmt"

var a, b, c int

func f(x int) int {
    return x * x
}

func g(x int) int {
    return x - 2
}

func h(x int) int {
    return x + 1
}

func fogog(x int) int {
    return f(g(h(x)))
}

func gohof(x int) int {
    return g(h(f(x)))
}

func hofog(x int) int {
    return h(f(g(x)))
}
```

```

func main() {
    fmt.Scan(&a, &b, &c)

    fmt.Printf("fogog(%d) = %d\n", a, fogog(a))
    fmt.Printf("gohof(%d) = %d\n", b, gohof(b))
    fmt.Printf("hofog(%d) = %d\n", c, hofog(c))
}

```

Output:

```

PS C:\Sheila Stephanie Anindya\SMT 2\Alpro 2\code> go run "c:\Sheila Stephanie Anindya\SMT 2\Alpro 2\code\main.go"
7 2 10
fogoh(7) = 36
gohof(2) = 3
hofog(10) = 65
PS C:\Sheila Stephanie Anindya\SMT 2\Alpro 2\code>

```

Penjelasan :

Program ini menghitung ketiga bilangan yang diinput, inputan pertama untuk menghitung fogog(x), kedua menghitung gohof(x), ketiga menghitung hofog(x).

Contoh perhitungannya : fogog (7) = f (g (h (7)))

- $h(7) = 7 + 1 = 8$
- $g(8) = 8 - 2 = 6$
- $f(6) = 6 \times 6 = 36$

3. Source Code:

```
package main

import (
    "fmt"
    "math"
)

func jarak(a, b, c, d int) float64 {
    return math.Sqrt(float64((a-c)*(a-c) + (b-d)*(b-d)))
}

func didalam(cx, cy, r, x, y int) bool {
    return jarak(cx, cy, x, y) <= float64(r)
}

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

    fmt.Scanln(&cx1, &cy1, &r1)
    fmt.Scanln(&cx2, &cy2, &r2)
    fmt.Scanln(&x, &y)

    // mengecek titik di lingkran 1 ap 2
    inside1 := didalam(cx1, cy1, r1, x, y)
    inside2 := didalam(cx2, cy2, r2, x, y)

    if inside1 && inside2 {
        fmt.Println("Titik di dalam lingkaran 1 dan 2")
    } else if inside1 {
```

```
        fmt.Println("Titik di dalam lingkaran 1")
    } else if inside2 {
        fmt.Println("Titik di dalam lingkaran 2")
    } else {
        fmt.Println("Titik di luar lingkaran 1 dan 2")
    }
}
```

Output:

```
PS C:\Sheila Stephanie Anindya\SMT 2\Alpro 2\code> go run "c:\Sheila Stephanie Anindya
"
1 1 5
8 8 4
2 2
Titik di dalam lingkaran 1
PS C:\Sheila Stephanie Anindya\SMT 2\Alpro 2\code> 
```

Penjelasan :

Program ini mengecek apakah titik berada di dalam lingkaran 1, 2, atau kedua lingkaran yang diberikan, dengan menghitung jaraknya. Jika jarak antara titik ke pusat lebih kecil atau sama dengan jari-jari, maka titik ada di dalam lingkaran.

IV. KESIMPULAN

Dalam pemrograman, fungsi adalah blok kode yang diberi nama tertentu. Menggunakan fungsi membuat kode lebih modular dan mengurangi pengulangan kode. Fungsi harus mengembalikan nilai dan memiliki deklarasi tipe nilai yang dikembalikan, serta menggunakan kata kunci return. Fungsi diperlukan ketika nilai dibutuhkan, seperti untuk mengassign nilai ke variabel atau dalam ekspresi. Nama fungsi harus jelas dan menggambarkan nilai, seperti median atau nilaiTerbesar.

Fungsi main() adalah fungsi utama dalam program Go yang dieksekusi saat program berjalan. Fungsi lain bisa ditulis dengan keyword func diikuti nama fungsi dan parameter dalam kurung. Parameter adalah variabel yang ditentukan saat memanggil fungsi dan bersifat opsional. Data yang dipakai untuk parameter saat pemanggilan disebut argumen.

Fungsi dengan parameter memungkinkan input nilai yang lebih spesifik. Parameter membantu fungsi menjadi lebih fleksibel, dapat digunakan kembali, dan membuat kode lebih terstruktur.

V. REFERENSI

1. Telkom University. (n.d.). *Modul 3 Fungsi Praktikum Algoritma dan Pemrograman* 2. Fakultas Informatika, Telkom University.
2. Novalagung. (n.d.). *Fungsi & Closure dalam Golang*.
<https://dasarpemrogramangolang.novalagung.com/A-fungsi-closure.html>.
3. Universitas Pembangunan Jaya. (n.d.). *Handout IFA105 - MP5 Fungsi*.
<https://ocw.upj.ac.id/files/Handout-IFA105-MP5-Fungsi.pdf>.
4. Dicoding. (n.d.). *Mengenal Parameter pada Bidang Pemrograman*.
<https://www.dicoding.com/blog/mengenal-parameter-pada-bidang-pemrograman/>.