

**LAPORAN PRAKTIKUM
ALGORITMA DAN PEMROGRAMAN 2**

MODUL III

FUNGSI



Oleh :

NAMA : Felix Pedrosa Valentino

NIM : 103112400056

KELAS : IF – 12 – 01

**S1 TEKNIK INFORMATIKA
TELKOM UNIVERSITY PURWOKERTO**

2025

I. DASAR TEORI

Definisi Fungsi

Fungsi adalah suatu rangkaian instruksi yang terstruktur dengan baik, yang bertujuan untuk memberikan atau menghasilkan nilai tertentu. Secara umum, fungsi memetakan input ke nilai lain, sehingga dapat dikatakan bahwa fungsi selalu menghasilkan atau mengembalikan suatu nilai. Sebuah subprogram dikategorikan sebagai fungsi jika memenuhi syarat berikut:

1. Terdapat deklarasi mengenai tipe nilai yang akan dikembalikan.
2. Terdapat kata kunci `'return'` di dalam tubuh subprogram.

Penerapan Fungsi

Fungsi `'main()'` berfungsi sebagai titik awal dalam program Go dan akan dieksekusi saat program dijalankan.

Selain fungsi `'main()'`, pengguna juga dapat membuat fungsi lainnya dengan cukup mudah. Caranya adalah dengan menuliskan kata kunci `'func'`, diikuti dengan nama fungsi, kemudian tanda kurung `'()'` (yang dapat diisi dengan parameter), dan diakhiri dengan kurung kurawal yang membungkus blok kode.

Parameter merupakan variabel yang terkait dengan fungsi, dan nilainya akan ditentukan saat fungsi dipanggil. Parameter bersifat opsional; sebuah fungsi bisa tidak memiliki parameter sama sekali, atau sebaliknya, dapat memiliki satu atau lebih parameter sesuai dengan kebutuhan.

Fungsi seringkali digunakan ketika sebuah nilai dibutuhkan, misalnya dalam hal :

- Penugasan nilai ke dalam suatu variabel
- Sebagai bagian dari ekspresi
- Sebagai argumen dalam suatu subprogram, dan lain sebagainya.

Oleh karena itu, penting untuk memilih nama fungsi yang mencerminkan nilai yang dihasilkan, biasanya berupa kata benda atau kata sifat. Beberapa

contoh nama fungsi yang baik adalah: median, rerata, nilaiTerbesar, ketemu, selesai, dan lain-lain.

Deklarasi Fungsi

Deklarasi fungsi mirip dengan prosedur, karena biasanya ditempatkan dalam blok terpisah dari program utama. Pada bagian deklarasi, setelah parameter, terdapat tipe data yang menunjukkan nilai yang akan dikembalikan. Sementara itu, di dalam badan fungsi, terdapat pernyataan return yang mengembalikan nilai tersebut.

Cara Memanggil Fungsi

Pemanggilan fungsi mirip dengan prosedur, yaitu dengan menuliskan nama fungsi beserta argumen yang sesuai dengan parameter yang dibutuhkan. Namun, terdapat perbedaan penting: fungsi dapat di-assign ke dalam variabel, berfungsi sebagai bagian dari ekspresi, serta dapat digunakan sebagai argumen dalam subprogram lainnya.

II. GUIDED

1. Guided 1

Source Code :

```
// Felix Pedrosa V

package main

import "fmt"

func main() {
    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 *= i
    }
    return hasil
}

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

Output :

```

PS D:\KULIAH\SEMESTER 2\Algoritma & Pemrograman 2\CODING - GOLANG - Alpro 2> go run "d:\KULIAH\SEMESTER 2\Algoritma & Pemrograman 2\CODING - GOLANG - Alpro 2\alpro2_week3_modul13\10311240056_Guided1\10311240056_Guided1.go"
5 3
60
PS D:\KULIAH\SEMESTER 2\Algoritma & Pemrograman 2\CODING - GOLANG - Alpro 2> go run "d:\KULIAH\SEMESTER 2\Algoritma & Pemrograman 2\CODING - GOLANG - Alpro 2\alpro2_week3_modul13\10311240056_Guided1\10311240056_Guided1.go"
5 2
20

```

Penjelasan Program :

Program di atas ditulis menggunakan bahasa Go dan berfungsi untuk menghitung nilai permutasi dari dua bilangan bulat yang dimasukkan oleh pengguna. Pertama-tama, program ini meminta pengguna untuk memasukkan dua bilangan bulat, yang kemudian disimpan dalam variabel *a* dan *b*. Selanjutnya, program akan membandingkan kedua bilangan tersebut. Jika *a* lebih besar atau sama dengan *b*, maka program akan menghitung permutasi dengan memanggil fungsi `permutasi(a, b)`. Namun, jika *b* lebih besar dari *a*, program akan menghitung permutasi dengan memanggil fungsi `permutasi(b, a)`.

Fungsi permutasi sendiri menghitung nilai permutasi menggunakan rumus $nPr = n! / (n - r)$ dan mengandalkan fungsi faktorial untuk mendapatkan nilai faktorial dari suatu bilangan. Fungsi faktorial ini bertugas menghitung hasil perkalian semua bilangan bulat positif hingga *n*. Setelah itu, program akan mencetak hasil perhitungan permutasi.

2. Guided 2

Source Code :

```

// Felix Pedrosa V

package main

import "fmt"

// Fungsi untuk mengonversi suhu dari Celsius 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 D:\KULIAH\SEMESTER 2\Algoritma & Pemrograman 2\CODING - GOLANG - Alpro 2> go run "d:\KULIAH\SEMESTER 2\Algoritma & Pemrograman 2\CODING - GOLANG - Alpro 2\alpro2_week3_modul3\18311240056_Guided2\18311240056_Guided2.go"
Masukkan jumlah data: 2
Masukkan suhu dalam Celsius:
28
32
Suhu dalam Fahrenheit:
82.40
89.60

```

Penjelasan Program :

Program ini ditulis menggunakan bahasa Go dan berfungsi untuk mengonversi suhu dari Celsius ke Fahrenheit. Pertama-tama, program meminta pengguna untuk memasukkan jumlah data suhu yang ingin dikonversi, yang disimpan dalam variabel N. Untuk memastikan input yang diberikan valid, program akan memeriksa apakah N merupakan angka positif. Jika tidak, program akan menampilkan pesan kesalahan dan menghentikan proses.

Selanjutnya, program membuat sebuah slice bernama `temperatures` untuk menyimpan suhu dalam Celsius yang akan dimasukkan oleh pengguna. Setelah itu, pengguna diminta untuk memasukkan suhu satu per satu, dan setiap suhu yang dimasukkan akan disimpan dalam slice tersebut. Jika terjadi kesalahan saat proses input, program akan menampilkan pesan kesalahan dan menghentikan eksekusi.

Setelah semua suhu berhasil dimasukkan, program akan mengonversi setiap suhu dari Celsius ke Fahrenheit dengan menggunakan fungsi `celsiusToFahrenheit`, kemudian mencetak hasil konversi dengan format dua desimal.

3. Guided 3

Source Code :

```
// Felix Pedrosa V

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 D:\KULIAH\SEMESTER 2\Algoritma & Pemrograman 2\CODING - GOLANG - Alpro 2> go run "d:\KULIAH\SEMESTER 2\Algoritma & Pemrograman 2\CODING - GOLANG - Alpro 2\alpro2_week3_modul3\18311240056_Guided3\18311240056_Guided3.go"
Masukkan jari-jari tabung: 5
Masukkan tinggi tabung: 5
=====
Luas Permukaan Tabung: 314.16 satuan²
Volume Tabung: 392.70 satuan³
=====

```

Penjelasan Program :

Program di atas ditulis menggunakan bahasa Go dan berfungsi untuk menghitung luas permukaan serta volume tabung berdasarkan jari-jari dan tinggi yang diinputkan oleh pengguna. Langkah pertama, program meminta pengguna untuk memasukkan nilai jari-jari (r) dan tinggi (t) tabung. Setelah menerima input, program akan melakukan validasi untuk memastikan bahwa kedua nilai tersebut merupakan angka yang sah dan lebih besar dari nol. Jika input yang diberikan tidak valid, program akan

menampilkan pesan “Input tidak valid! Harap masukkan angka yang benar”. Namun, jika input valid, program akan menghitung luas permukaan tabung dengan memanfaatkan fungsi `luasPermukaanTabung` yang menerapkan rumus $(2 \pi r (r + t))$, serta volume tabung melalui fungsi `volumeTabung` yang menggunakan rumus $(\pi r^2 t)$. Setelah perhitungan tersebut selesai, program akan menampilkan hasil luas permukaan dan volume tabung dengan format dua desimal, sehingga pengguna mendapatkan informasi yang jelas mengenai karakteristik tabung yang telah dihitung. Dengan demikian, program ini memudahkan pengguna dalam melakukan perhitungan geometris secara akurat dan efisien.

III. UNGUIDED

1. UnGuided 1

Source Code :

```
// Felix Pedrosa V

package main

import "fmt"

// Fungsi untuk menghitung faktorial
func hitungFaktorial(n int) int {
    if n == 0 || n == 1 {
        return 1
    }
    hasil := 1
    for i := 2; i <= n; i++ {
        hasil *= i
    }
    return hasil
}

// Fungsi untuk menghitung permutasi
func hitungPermutasi(n, r int) int {
    return hitungFaktorial(n) / hitungFaktorial(n-r)
}

// Fungsi untuk menghitung kombinasi
func hitungKombinasi(n, r int) int {
    return hitungFaktorial(n) / (hitungFaktorial(r) * hitungFaktorial(n-r))
}

func main() {
    var x, y, z, w int
    fmt.Scan(&x, &y, &z, &w)

    if x >= z && y >= w {
        fmt.Println(hitungPermutasi(x, z), hitungKombinasi(x, z))
        fmt.Println(hitungPermutasi(y, w), hitungKombinasi(y, w))
    }
}
```

```

    } else {
        fmt.Println("Input tidak valid, pastikan x >= z dan y >= w")
    }
}

```

Output :

```

PS D:\KULIAH\SEMESTER 2\Algoritma & Pemrograman 2\CODING - GOLANG - Alpro 2> go run "d:\KULIAH\SEMESTER 2\Algoritma & Pemrograman 2\CODING -
GOLANG - Alpro 2\alpro2_week3_modul3\103112400056_Unguided1\103112400056_Unguided1.go"
5 10 3 10
60 10
3628800 1
PS D:\KULIAH\SEMESTER 2\Algoritma & Pemrograman 2\CODING - GOLANG - Alpro 2> go run "d:\KULIAH\SEMESTER 2\Algoritma & Pemrograman 2\CODING -
GOLANG - Alpro 2\alpro2_week3_modul3\103112400056_Unguided1\103112400056_Unguided1.go"
8 0 2 0
56 28
1 1

```

Penjelasan Program :

Program diatas ditulis menggunakan bahasa Go dan berfungsi untuk menghitung nilai permutasi serta kombinasi dari dua pasangan bilangan bulat yang dimasukkan oleh pengguna. Pertama-tama, program mendefinisikan fungsi `hitungFaktorial` yang bertugas menghitung faktorial dari bilangan bulat n . Fungsi ini akan mengembalikan nilai 1 jika n sama dengan 0 atau 1, dan akan menghitung hasil perkalian dari semua bilangan bulat positif hingga n untuk nilai n yang lebih besar.

Selanjutnya, terdapat fungsi `hitungPermutasi` yang menghitung permutasi dari n terhadap r menggunakan rumus $P(n, r) = \frac{n!}{(n-r)!}$, serta fungsi `hitungKombinasi` yang menghitung kombinasi dari n terhadap r dengan rumus $C(n, r) = \frac{n!}{r!(n-r)!}$.

Dalam fungsi utama, program meminta pengguna untuk memasukkan empat bilangan bulat, yaitu x , y , z , dan w . Program kemudian memeriksa apakah kondisi $x \geq z$ dan $y \geq w$ terpenuhi. Jika kondisi tersebut terpenuhi, program akan menghitung dan mencetak hasil permutasi dan kombinasi untuk kedua pasangan bilangan tersebut. Jika tidak, program akan menampilkan pesan kesalahan yang menyatakan bahwa input yang diberikan tidak valid.

2. UnGuided 2

Source Code :

```
// Felix Pedrosa V
```

```
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
}

// Komposisi  $f(g(h(x)))$ 
func fogoh(x int) int {
    return f(g(h(x)))
}

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

// Komposisi  $h(f(g(x)))$ 
func hofog(x int) int {
    return h(f(g(x)))
}

func main() {
    var a, b, c int
    fmt.Scan(&a, &b, &c)
    fmt.Println(fogoh(a))
    fmt.Println(gohof(b))
    fmt.Println(hofog(c))
}
```

```
}
```

Output :

```
PS D:\KULIAH\SEMESTER 2\Algoritma & Pemrograman 2\CODING - GOLANG - Alpro 2> go run "d:\KULIAH\SEMESTER 2\Algoritma & Pemrograman 2\CODING - GOLANG - Alpro 2\alpro2_week3_modul3\103112400056_Unguided2\103112400056_Unguided2.go"
7 2 10
36
3
65
PS D:\KULIAH\SEMESTER 2\Algoritma & Pemrograman 2\CODING - GOLANG - Alpro 2> go run "d:\KULIAH\SEMESTER 2\Algoritma & Pemrograman 2\CODING - GOLANG - Alpro 2\alpro2_week3_modul3\103112400056_Unguided2\103112400056_Unguided2.go"
5 5 5
16
24
10
PS D:\KULIAH\SEMESTER 2\Algoritma & Pemrograman 2\CODING - GOLANG - Alpro 2> go run "d:\KULIAH\SEMESTER 2\Algoritma & Pemrograman 2\CODING - GOLANG - Alpro 2\alpro2_week3_modul3\103112400056_Unguided2\103112400056_Unguided2.go"
3 8 4
4
63
5
```

Penjelasan Program :

Program diatas ditulis menggunakan bahasa Go dan berfungsi untuk menghitung nilai dari beberapa komposisi fungsi matematis. Terdapat tiga fungsi dasar yang telah didefinisikan: fungsi $f(x) = x^2$, fungsi $g(x) = x - 2$, dan fungsi $h(x) = x + 1$. Selanjutnya, program ini mengimplementasikan komposisi dari fungsi-fungsi tersebut dalam tiga bentuk yang berbeda, yaitu $\text{fogoh}(x) = f(g(h(x)))$, $\text{gohof}(x) = g(h(f(x)))$, dan $\text{hofog}(x) = h(f(g(x)))$.

Pengguna diharuskan untuk memasukkan tiga bilangan bulat, yaitu a, b, dan c, yang dipisahkan oleh spasi. Setelah input diterima, program akan menampilkan hasil dari masing-masing komposisi fungsi untuk ketiga bilangan tersebut. Hasil keluaran akan ditampilkan dalam tiga baris, menunjukkan output dari $\text{fogoh}(a)$, $\text{gohof}(b)$, dan $\text{hofog}(c)$.

3. UnGuided 3

Source Code :

```
// Felix Pedrosa V

package main

import (
    "fmt"
    "math"
)

// Fungsi untuk menghitung jarak antara dua titik
```

```

func hitungJarak(x1, y1, x2, y2 float64) float64 {
    return math.Sqrt(math.Pow(x1-x2, 2) + math.Pow(y1-y2, 2))
}

// Fungsi untuk memeriksa apakah titik berada di dalam lingkaran
func titikDalamLingkaran(centerX, centerY, radius, pointX, pointY
float64) bool {
    return hitungJarak(centerX, centerY, pointX, pointY) <= radius
}

func main() {
    var centerX1, centerY1, radius1 float64
    var centerX2, centerY2, radius2 float64
    var koordinatX, koordinatY float64

    fmt.Scan(&centerX1, &centerY1, &radius1)
    fmt.Scan(&centerX2, &centerY2, &radius2)
    fmt.Scan(&koordinatX, &koordinatY)

    // Memeriksa apakah titik berada di dalam lingkaran
    dalamLingkaran1 := titikDalamLingkaran(centerX1, centerY1, radius1,
koordinatX, koordinatY)
    dalamLingkaran2 := titikDalamLingkaran(centerX2, centerY2, radius2,
koordinatX, koordinatY)

    // Menampilkan hasil
    if dalamLingkaran1 && dalamLingkaran2 {
        fmt.Println("Titik di dalam lingkaran 1 dan 2")
    } else if dalamLingkaran1 {
        fmt.Println("Titik di dalam lingkaran 1")
    } else if dalamLingkaran2 {
        fmt.Println("Titik di dalam lingkaran 2")
    } else {
        fmt.Println("Titik di luar lingkaran 1 dan 2")
    }
}

```

Output :

```

PS D:\KULIAH\SEMESTER 2\Algoritma & Pemrograman 2\CODING - GOLANG - Alpro 2> go run "d:\KULIAH\SEMESTER 2\Algoritma & Pemrograman 2\CODING -
GOLANG - Alpro 2\alpro2_week3_modul3\103112400056_Unguided3\103112400056_Unguided3.go"
1 1 5
8 8 4
2 2
Titik di dalam lingkaran 1
PS D:\KULIAH\SEMESTER 2\Algoritma & Pemrograman 2\CODING - GOLANG - Alpro 2> go run "d:\KULIAH\SEMESTER 2\Algoritma & Pemrograman 2\CODING -
GOLANG - Alpro 2\alpro2_week3_modul3\103112400056_Unguided3\103112400056_Unguided3.go"
1 2 3
4 5 6
7 8
Titik di dalam lingkaran 2
PS D:\KULIAH\SEMESTER 2\Algoritma & Pemrograman 2\CODING - GOLANG - Alpro 2> go run "d:\KULIAH\SEMESTER 2\Algoritma & Pemrograman 2\CODING -
GOLANG - Alpro 2\alpro2_week3_modul3\103112400056_Unguided3\103112400056_Unguided3.go"
5 10 15
-15 4 20
0 0
Titik di dalam lingkaran 1 dan 2
PS D:\KULIAH\SEMESTER 2\Algoritma & Pemrograman 2\CODING - GOLANG - Alpro 2> go run "d:\KULIAH\SEMESTER 2\Algoritma & Pemrograman 2\CODING -
GOLANG - Alpro 2\alpro2_week3_modul3\103112400056_Unguided3\103112400056_Unguided3.go"
1 1 5
8 8 4
15 20
Titik di luar lingkaran 1 dan 2

```

Penjelasan Program :

Program diatas ditulis menggunakan bahasa Go dan berfungsi untuk menentukan posisi suatu titik relatif terhadap dua lingkaran yang ditentukan oleh pusat dan radiusnya. Pertama-tama, program mendefinisikan sebuah fungsi bernama `hitungJarak` yang berfungsi untuk menghitung jarak antara dua titik dalam ruang dua dimensi dengan menggunakan rumus jarak Euclidean. Fungsi ini akan menerima koordinat dari kedua titik tersebut dan mengembalikan nilai jarak dalam bentuk bilangan pecahan.

Selanjutnya, terdapat fungsi lain yang disebut `titikDalamLingkaran`, yang berfungsi untuk memeriksa apakah suatu titik berada di dalam lingkaran. Hal ini dilakukan dengan membandingkan jarak titik tersebut dari pusat lingkaran dengan radius lingkaran itu sendiri. Di dalam fungsi main, program akan meminta pengguna untuk memasukkan koordinat pusat dan radius dari kedua lingkaran, serta koordinat titik yang ingin diperiksa.

Setelah menerima semua input yang diperlukan, program akan menggunakan fungsi `titikDalamLingkaran` untuk menentukan apakah titik yang dimaksud berada di dalam lingkaran pertama, lingkaran kedua, atau di luar kedua lingkaran tersebut. Hasil dari pemeriksaan ini kemudian ditampilkan kepada pengguna melalui pesan yang jelas, sehingga memberikan informasi apakah titik tersebut berada di dalam salah satu dari lingkaran, kedua lingkaran, atau di luar keduanya.

IV. KESIMPULAN

Modul ini membahas konsep dasar mengenai fungsi dalam pemrograman serta cara mengimplementasikannya. Fungsi merupakan subprogram yang menghasilkan nilai tertentu berdasarkan input yang diterima. Berikut adalah poin-poin utama yang dibahas dalam modul ini:

1. Definisi dan Deklarasi Fungsi

Setiap fungsi harus memiliki tipe data yang akan dikembalikan dan menggunakan kata kunci `return`. Penamaan fungsi sebaiknya mencerminkan nilai yang akan dihasilkan. Deklarasi fungsi dilakukan terpisah dari program utama agar lebih terstruktur.

2. Implementasi dalam Algoritma dan Bahasa Go

Modul ini juga menyediakan format deklarasi fungsi dalam notasi algoritma, serta menjelaskan bagaimana cara implementasinya dalam bahasa Go. Sebagai contoh, diberikan penjelasan mengenai fungsi sederhana, seperti cara menghitung volume tabung.

3. Cara Memanggil Fungsi

Fungsi dapat dipanggil dengan memberikan argumen yang sesuai. Selain itu, fungsi juga dapat digunakan dalam berbagai operasi matematika, ekspresi, atau langsung dicetak untuk tujuan tertentu.

V. REFERENSI

Modul 3 - Praktikum Alpro 2

A.18. Fungsi. (n.d.). Retrieved from
<https://dasarpemrogramangolang.novalagung.com/A-fungsi.html>