

**LAPORAN PRAKTIKUM
ALGORITME DAN PEMROGRAMAN 2**

**MODUL 3
FUNGSI**



Oleh:

RAIFANKA RAISA RAMADHAN

2311102205

IF - 11 - 07

**PROGRAM STUDI S1 TEKNIK INFORMATIKA
FAKULTAS INFORMATIKA
INSTITUT TEKNOLOGI TELKOM PURWOKERTO
2024**

I. Dasar Teori

Fungsi didefinisikan dengan kata kunci `func`, diikuti nama fungsi, parameter, tipe data hasil, dan isi fungsi. Fungsi dapat menerima parameter dan mengembalikan satu atau lebih nilai, dengan fitur pengembalian nilai ganda yang memudahkan penanganan kesalahan. Sebagai contoh, fungsi bagi menerima dua parameter dan mengembalikan hasil pembagian serta pesan error jika terjadi pembagian dengan nol. Selain itu, fungsi di Go dapat diperlakukan sebagai *first-class citizens*, yang berarti bisa disimpan dalam variabel, diteruskan sebagai parameter, atau dikembalikan dari fungsi lain. Fleksibilitas ini memungkinkan penggunaan konsep seperti *closure*, *callback*, dan *rekursi*.

II. Guided Guided 1

```
package main

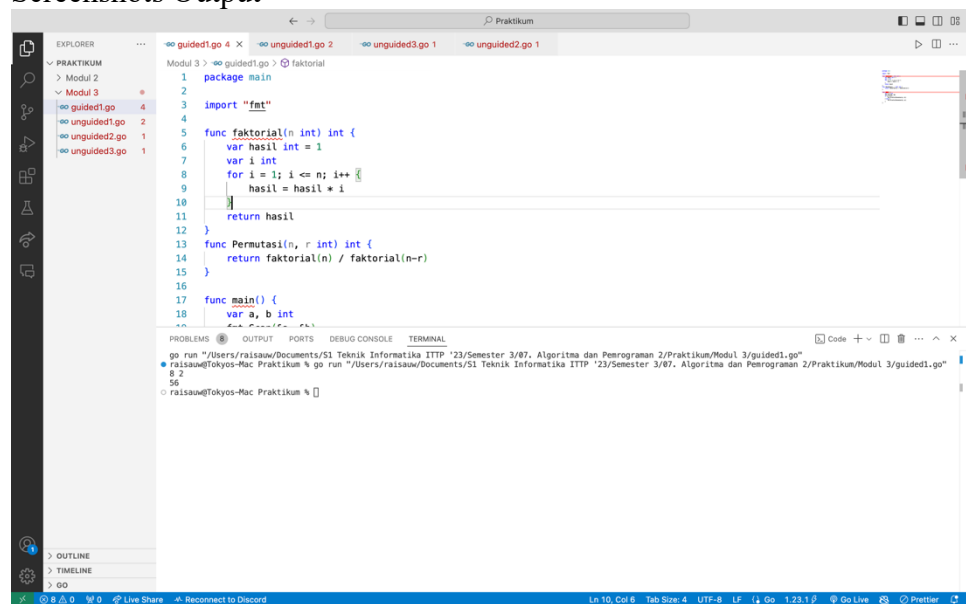
import "fmt"

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

func Permutasi(n, r int) int {
    return faktorial(n) / faktorial(n-r)
}

func main() {
    var a, b int
    fmt.Scan(&a, &b)
    if a >= b {
        fmt.Println(Permutasi(a, b))
    } else {
        fmt.Println(Permutasi(b, a))
    }
}
```

Screenshots Output



Deskripsi:

Program ini adalah program untuk menghitung permutasi dari dua bilangan n dan r . Fungsi faktorial menghitung nilai faktorial dari bilangan n dengan menggunakan perulangan. Fungsi Permutasi kemudian menghitung nilai permutasi menggunakan rumus $P(n, r) = n! / (n - r)!$. Di bagian `main()`, program menerima dua input dari pengguna, yaitu a dan b . Jika a lebih besar atau sama dengan b , maka program akan menghitung permutasi $P(a, b)$. Jika sebaliknya, program menghitung permutasi $P(b, a)$ dan menampilkan hasilnya.

Guided 2

```
package main

import "fmt"

func volumeTabung(jari_jari, tinggi int) float64 {
    var luasAlas, volume float64
    luasAlas = 3.14 * float64(jari_jari*jari_jari)
    volume = luasAlas * float64(tinggi)
    return volume
}

func main() {
    var r, t int
    var v1, v2 float64
    r = 5
    t = 10

    v1 = volumeTabung(r, t)
    v2 = volumeTabung(r, t) + volumeTabung(15, t)

    fmt.Println("volume tabung 14 x 100", volumeTabung(14,
100))
    fmt.Println("v1:", v1)
    fmt.Println("v2:", v2)
}
```

Screenshots Output



Deskripsi:

Program ini adalah program untuk menghitung volume tabung dengan menggunakan jari-jari dan tinggi sebagai input. Fungsi `volumeTabung` menghitung luas alas tabung berdasarkan rumus πr^2 (dengan $\pi \approx 3.14$) dan mengalikannya dengan tinggi untuk mendapatkan volume. Di dalam fungsi `main()`, program menghitung volume dari tabung dengan jari-jari 5 dan tinggi 10 (disimpan dalam variabel `v1`), serta menghitung dua volume lain (disimpan dalam `v2`), yang merupakan jumlah dari volume tabung dengan jari-jari 5 dan tabung dengan jari-jari 15, keduanya dengan tinggi 10. Hasil perhitungan untuk beberapa tabung lainnya juga dicetak ke layar.

III. Unguided Unguided 1

```
package main

import (
    "fmt"
)

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

func permutasi(n, r int) int {
    return faktorial(n) / faktorial(n-r)
}

func kombinasi(n, r int) int {
    return faktorial(n) / (faktorial(r) * faktorial(n-r))
}

func main() {
    var a, b, c, d int
    fmt.Print("input angka : ")
    fmt.Scanln(&a, &b, &c, &d)
    if a >= c && b >= d {
        fmt.Println(permutasi(a, c), kombinasi(a, c))
        fmt.Println(permutasi(b, d), kombinasi(b, d))
    } else {
        fmt.Print("MAAF ANDA TIDAK MEMENUHI PERSYARATAN")
    }
}
```

Screenshots Output

```
7 func faktorial(n int) int {
11 }
12 return hasil
13 }
14 func permutasi(n, r int) int {
15     return faktorial(n) / faktorial(n-r)
16 }
17 func kombinasi(n, r int) int {
18     return faktorial(n) / (faktorial(r) * faktorial(n-r))
19 }
20
21 func main() {
22     var a, b, c, d int
23     fmt.Println("Input angka : ")
24     fmt.Scanln(&a, &b, &c, &d)
25     if a >= c && b >= d {
26         fmt.Println(permutasi(a, c), kombinasi(a, c))
27         fmt.Println(permutasi(b, d), kombinasi(b, d))
28     } else {
29         fmt.Println("MAAF ANDA TIDAK MEMENUHI PERSYARATAN")
30     }
31 }
```

```
go run "/Users/raisaaw/Documents/S1 Teknik Informatika ITPP '23/Semester 3/07. Algoritma dan Pemrograman 2/Praktikum/Modul 3/unguided1.go"
raisaaw@Tokyo-Mac 07. Algoritma dan Pemrograman 2 % go run "/Users/raisaaw/Documents/S1 Teknik Informatika ITPP '23/Semester 3/07. Algoritma dan Pemrograman 2/Praktikum/Modul 3/unguided1.go"
input angka : 8 8 2 0
56 28
1 1
raisaaw@Tokyo-Mac 07. Algoritma dan Pemrograman 2 %
```

Deskripsi:

Program ini adalah program untuk menghitung permutasi dan kombinasi dari dua pasang bilangan. Fungsi faktorial menghitung nilai faktorial dari bilangan yang diberikan. Fungsi permutasi menghitung permutasi $P(n, r)$ menggunakan rumus $P(n, r) = n! / (n - r)!$, sementara fungsi kombinasi menghitung kombinasi $C(n, r)$ dengan rumus $C(n, r) = n! / (r!(n - r)!)$. Di dalam fungsi `main()`, program menerima empat input bilangan dari pengguna (`a`, `b`, `c`, dan `d`). Jika `a >= c` dan `b >= d`, maka program akan menghitung dan menampilkan permutasi serta kombinasi untuk kedua pasangan bilangan tersebut. Jika tidak memenuhi syarat, program akan menampilkan pesan "MAAF ANDA TIDAK MEMENUHI PERSYARATAN".

Unguided 2

```
package main

import (
    "fmt"
)

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 fogoh(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() {
    var a, b, c int

    fmt.Println("Masukkan bilangan bulat : ")
    fmt.Scanf("%d %d %d", &a, &b, &c)

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

Screenshots Output



```
1 package main
2
3 import (
4     "fmt"
5 )
6
7 func f(x int) int {
8     return x * x
9 }
10
11 func g(x int) int {
12     return x - 2
13 }
14
15 func h(x int) int {
16     return x + 1
17 }
18
19 func fogoh(a, b, c int) {
20     fmt.Println("Masukkan bilangan bulat :")
21     fmt.Println("a = ", a)
22     fmt.Println("b = ", b)
23     fmt.Println("c = ", c)
24     fogoh(f(g(h(a))), g(h(f(b))), h(f(g(c))))
25 }
26
27 func gohof(a, b, c int) {
28     gohof(g(h(f(a))), h(f(g(b))), f(g(h(c))))
29 }
30
31 func hofog(a, b, c int) {
32     hofog(h(f(g(a))), f(g(h(b))), g(h(f(c))))
33 }
34
35 func main() {
36     a, b, c := 7, 2, 18
37     fogoh(a, b, c)
38     gohof(a, b, c)
39     hofog(a, b, c)
40 }
```

Output:

```
go run "/Users/raisaaw/Documents/S1 Teknik Informatika ITPP '23/Semester 3/07. Algoritma dan Pemrograman 2/Praktikum/Modul 3/unguided2.go"
raisaaw@Tokyo-Mac 07. Algoritma dan Pemrograman 2 % go run "/Users/raisaaw/Documents/S1 Teknik Informatika ITPP '23/Semester 3/07. Algoritma dan Pemrograman 2/Praktikum/Modul 3/unguided2.go"
Masukkan bilangan bulat :
7 2 18
fogoh(7) = 36
gohof(2) = 3
hofog(18) = 65
raisaaw@Tokyo-Mac 07. Algoritma dan Pemrograman 2 %
```

Deskripsi:

P Program ini adalah program untuk menghitung komposisi fungsi dari tiga fungsi sederhana: $f(x) = x^2$, $g(x) = x - 2$, dan $h(x) = x + 1$. Program mendefinisikan tiga komposisi fungsi, yaitu $\text{fogoh}(x)$ yang menggabungkan fungsi $f(g(h(x)))$, $\text{gohof}(x)$ yang menggabungkan $g(h(f(x)))$, dan $\text{hofog}(x)$ yang menggabungkan $h(f(g(x)))$. Di bagian $\text{main}()$, program meminta pengguna untuk memasukkan tiga bilangan bulat (a , b , dan c). Kemudian, program akan menghitung dan menampilkan hasil dari ketiga komposisi fungsi tersebut untuk masing-masing bilangan input yang dimasukkan.

Unguided 3

```
package main

import (
    "fmt"
    "math"
)

func jarak(cx, cy, x, y int) float64 {
    return math.Sqrt(float64((cx-x)*(cx-x) + (cy-y)*(cy-y)))
}

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

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

    fmt.Println("Masukkan koordinat pusat dan radius lingkaran 1 (cx1 cy1 r1):")
    fmt.Scanf("%d %d %d", &cx1, &cy1, &r1)

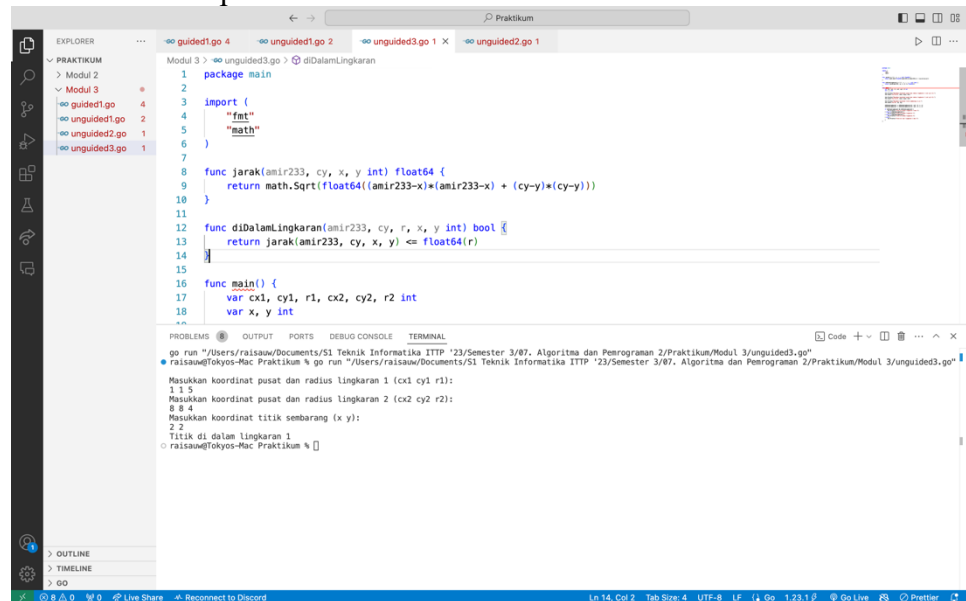
    fmt.Println("Masukkan koordinat pusat dan radius lingkaran 2 (cx2 cy2 r2):")
    fmt.Scanf("%d %d %d", &cx2, &cy2, &r2)

    fmt.Println("Masukkan koordinat titik sembarang (x y):")
    fmt.Scanf("%d %d", &x, &y)

    diDalamLingkaran1 := diDalamLingkaran(cx1, cy1, r1, x, y)
    diDalamLingkaran2 := diDalamLingkaran(cx2, cy2, r2, x, y)

    if diDalamLingkaran1 && diDalamLingkaran2 {
        fmt.Println("Titik di dalam lingkaran 1 dan 2")
    } else if diDalamLingkaran1 {
        fmt.Println("Titik di dalam lingkaran 1")
    } else if diDalamLingkaran2 {
        fmt.Println("Titik di dalam lingkaran 2")
    } else {
        fmt.Println("Titik di luar lingkaran 1 dan 2")
    }
}
```

Screenshots Output



```
Modul 3 > > unguided3.go > diDalamLingkaran
1 package main
2
3 import {
4     "fmt"
5     "math"
6 }
7
8 func jarak(amir233, cy, x, y int) float64 {
9     return math.Sqrt(float64((amir233-x)*(amir233-x) + (cy-y)*(cy-y)))
10 }
11
12 func diDalamLingkaran(amir233, cy, r, x, y int) bool {
13     return jarak(amir233, cy, x, y) <= float64(r)
14 }
15
16 func main() {
17     var cx1, cy1, r1, cx2, cy2, r2 int
18     var x, y int
19
20     go run "/Users/raisau/Documents/S1 Teknik Informatika ITPP '23/Semester 3/07. Algoritma dan Penrograman 2/Praktikum/Modul 3/unguided3.go"
21     raisau@Tokyo-Mac Praktikum % go run "/Users/raisau/Documents/S1 Teknik Informatika ITPP '23/Semester 3/07. Algoritma dan Penrograman 2/Praktikum/Modul 3/unguided3.go"
22     Masukkan koordinat pusat dan radius lingkaran 1 (cx1 cy1 r1):
23     1 1 5
24     Masukkan koordinat pusat dan radius lingkaran 2 (cx2 cy2 r2):
25     8 8 4
26     Masukkan koordinat titik sembarang (x y):
27     2 2
28     Titik di dalam lingkaran 1
29     raisau@Tokyo-Mac Praktikum %
```

Deskripsi:

Program ini adalah program untuk menentukan apakah suatu titik berada di dalam satu atau dua lingkaran yang diberikan. Fungsi jarak menghitung jarak Euclidean antara pusat lingkaran dan titik sembarang menggunakan rumus $\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$. Fungsi diDalamLingkaran memeriksa apakah jarak dari titik ke pusat lingkaran lebih kecil atau sama dengan jari-jari lingkaran, yang berarti titik tersebut berada di dalam lingkaran. Di dalam fungsi main(), program meminta input dari pengguna berupa koordinat pusat dan jari-jari dua lingkaran serta koordinat sebuah titik. Setelah itu, program memeriksa apakah titik tersebut berada di dalam salah satu, kedua, atau di luar kedua lingkaran, dan menampilkan hasilnya sesuai dengan kondisi yang ditemukan.