

**LAPORAN PRAKTIKUM
ALGORITMA PEMROGRAMAN 2**

MODUL 3

FUNGSI



Oleh:

NAMA : Alberta Aurora Adiswari

NIM : 2311102220

KELAS : S1-IF-11-02

**S1 TEKNIK INFORMATIKA
INSTITUT TEKNOLOGI TELKOM PURWOKERTO**

2024

I. DASAR TEORI

Fungsi merupakan satu kesatuan rangkaian instruksi yang memberikan dan menghasilkan suatu nilai yang lain. Oleh karena itu, fungsi selalu menghasilkan/mengembalikan nilai. Suatu subprogram dikatakan fungsi apabila:

- **Ada** deklarasi tipe nilai yang dikembalikan, dan
- **Terdapat** kata kunci **return** dalam badan subprogram.

Maka fungsi digunakan jika suatu nilai biasanya diperlukan, seperti:

- Assignment nilai ke suatu variable
- Bagian dari ekspresi
- Bagian dari argument suatu subprogram, dsb.

Karena itu selalu pilih nama fungsi yang menggambarkan nilai, seperti kata benda dan kata sifat.

Deklarasi fungsi sama dengan prosedur, yaitu berada pada blok yang terpisah dengan program utama.

	Notasi Algoritma
1	function <nama function> (<params>) -> <type>
2	kamus
3	{deklarasi variabel lokal dari fungsi}
4	...
5	algoritma
6	{badan algoritma fungsi}
7	...
8	return <value/variabel>
9	endfunction
	Notasi dalam bahasa Go
10	func <nama function> (<params>) <type> {
11	/* deklarasi variabel lokal dari fungsi */
12	...
13	/* badan algoritma fungsi*/
14	...
15	return <value/variabel>
16	
17	}

Pada deklarasi setelah parameter terdapat tipe data dari nilai yang dikembalikan, sedangkan pada bagian badan fungsi terdapat return dari nilai yang dikembalikan.

Sama halnya dengan prosedur, pemanggilan fungsi cukup dilakukan dengan penulisan nama fungsi beserta argument yang diminta oleh parameter dari

fungsi. Perbedaannya dengan prosedur adalah fungsi bisa di assign ke suatu variabel, menjadi bagian dari ekspresi, dan argument dari suatu subprogram.

Notasi Algoritma	
1	program ContohProsedur
2	kamus
3	r,t : integer
4	v1,v2 : real
5	algoritma
6	r <- 5;
7	t <- 10
8	v1 <- volumeTabung(r,t) {cara pemanggilan #1}
9	v2 <- volumeTabung(r,t) + volumeTabung(15,t) {cara pemanggilan #2}
10	output(volumeTabung(14,100)) {cara pemanggilan #3}
11	endprogram
Notasi dalam bahasa Go	
12	func main() {
13	var r,t int
14	var v1,v2 float64
15	r = 5
16	t = 10
17	v1 = volumeTabung(r,t) // cara pemanggilan #1
18	v2 = volumeTabung(r,t) + volumeTabung(15,t) // cara pemanggilan #2
19	fmt.Println(volumeTabung(14,100)) // cara pemanggilan #3
20	}

Pada contoh pemanggilan fungsi tersebut tidak ada perbedaan pada saat pemanggilan fungsi pada pseudocode ataupun GoLang. Disini fungsi bisa di-assign ke suatu variabel pada saat pemanggilan, bisa dioperasikan sesuai dengan tipe data yang dikembalikan, dan juga bisa langsung ditampilkan dengan perintah output ataupun print.

II. GUIDED

1. Latihan fungsi

Source code

```
package main

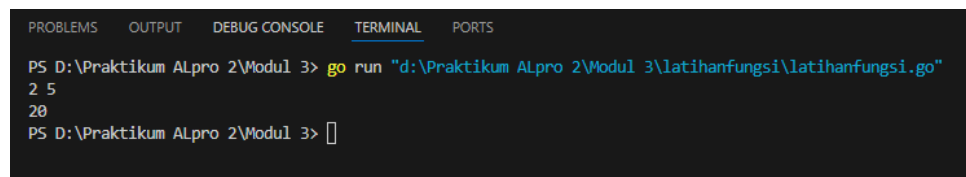
import "fmt"

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 {
    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))
}
```

Screenshoot program



```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS
PS D:\Praktikum ALpro 2\Modul 3> go run "d:\Praktikum ALpro 2\Modul 3\latihanfungsi\latihanfungsi.go"
2 5
20
PS D:\Praktikum ALpro 2\Modul 3> 
```

Deskripsi program

Pada program ini menggunakan Bahasa pemrograman Go untuk menghitung permutasi dari dua bilangan yang diinputkan oleh pengguna. Pengguna diminta untuk menginputkan dua bilangan dari a dan b, lalu program memeriksa apakah a lebih besar atau sama dengan b, jika iya, program menghitung permutasi menggunakan permutasi (a, b), namun jika tidak program menghitung permutasi (b,a). Fungsi permutasi(n, r) dihitung berdasarkan rumus. Pada program inputannya berupa angka 2 dan 5, lalu program menghitung $P(5,2)$, yang hasilnya adalah 20, menampilkan hasilnya pada layar.

2. Guided 1

Source code

```
package main

import "fmt"

func factorial(n int) int {
    if n == 0 || n == 1 {
        return 1
    }
    result := 1
    for i := 2; i <= n; i++ {
        result *= i
    }
    return result
}

func permutation(n, r int) int {
    return factorial(n) / factorial(n-r)
}

func combination(n, r int) int {
    return factorial(n) / (factorial(r) * factorial(n-r))
}

func main() {
    var a, b, c, d int
    fmt.Println("Masukkan bilangan a, b, c, d (dengan spasi): ")
    fmt.Scanf("%d %d %d %d", &a, &b, &c, &d)

    if a >= c && b >= d {
        permutasiAC := permutation(a, c)
        kombinasiAC := combination(a, c)

        permutasiBD := permutation(b, d)
        kombinasiBD := combination(b, d)

        fmt.Println("Permutasi(a, c) dan Kombinasi(a, c):", permutasiAC, kombinasiAC)
        fmt.Println("Permutasi(b, d) dan Kombinasi(b, d):", permutasiBD, kombinasiBD)
    } else {
        fmt.Println("Syarat a >= c dan b >= d tidak terpenuhi.")
    }
}
```

Screenshoot program

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS D:\Praktikum ALpro 2\Modul 3> go run "d:\Praktikum ALpro 2\Modul 3\guided1\guided1_rora_220.go"
Masukkan bilangan a, b, c, d (dengan spasi):
5 10 3 10
Permutasi(a, c) dan Kombinasi(a, c): 60 10
Permutasi(b, d) dan Kombinasi(b, d): 3628800 1
PS D:\Praktikum ALpro 2\Modul 3> go run "d:\Praktikum ALpro 2\Modul 3\guided1\guided1_rora_220.go"
Masukkan bilangan a, b, c, d (dengan spasi):
8 0 2 0
Permutasi(a, c) dan Kombinasi(a, c): 56 28
Permutasi(b, d) dan Kombinasi(b, d): 1 1
PS D:\Praktikum ALpro 2\Modul 3> |
```

Deskripsi program

Pada program ini menggunakan Bahasa pemrograman Go yang dimana program ini menghitung permutasi dan kombinasi dari dua pasang bilangan yang dimasukkan oleh pengguna, yaitu (a, c) dan (b, d). Pengguna diminta untuk menginputkan empat bilangan: a, b, c, dan d kemudian program memeriksa apakah syarat $a \geq c$ dan $b \geq d$ terpenuhi. Program akan menghitung nilai permutasi dan kombinasi menggunakan fungsi `permutation()` dan `combination()` sesuai dengan perhitungan factorial dari nilai-nilai yang diinputkan. Permutasi dihitung dengan rumus $P(n, r) = \frac{n!}{(n-r)!}$

Sedangkan kombinasi dihitung menggunakan rumus $C(n, r) = \frac{n!}{r!(n-r)!}$. Hasil

dari kedua perhitungan (a, c) dan (b, d) dicetak ke layar. Jika syarat tidak terpenuhi program akan menampilkan bahwa syarat tersebut tidak valid.

III. UNGUIDED

Source code

```
package main

import (
    "fmt"
    "math"
)

func jarak(x1, y1, x2, y2 int) float64 {
    return math.Sqrt(float64((x2-x1)*(x2-x1) + (y2-y1)*(y2-y1)))
}

func diDalamLingkaran_220(cx, cy, r, x, y int) bool {
    return jarak(cx, cy, x, y) <= float64(r)
}

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

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

    diDalamLing1 := diDalamLingkaran_220(cx1, cy1, r1,
x, y)
    diDalamLing2 := diDalamLingkaran_220(cx2, cy2, r2,
x, y)

    if diDalamLing1 && diDalamLing2 {
        fmt.Println("Titik di dalam lingkaran 1 dan 2")
    } else if diDalamLing1 {
        fmt.Println("Titik di dalam lingkaran 1")
    } else if diDalamLing2 {
        fmt.Println("Titik di dalam lingkaran 2")
    } else {
        fmt.Println("Titik di luar lingkaran 1 dan 2")
    }
}
```

Screenshoot program

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

\Modul 3\unguided\unguided_rora_220.go"
1 1 5
8 8 4
2 2
Titik di dalam lingkaran 1
PS D:\Praktikum ALpro 2\Modul 3> go run "d:\Praktikum ALpro 2\Modul 3\unguided\unguided_rora_220.go"
1 2 3
4 5 6
7 8
Titik di dalam lingkaran 2
PS D:\Praktikum ALpro 2\Modul 3> go run "d:\Praktikum ALpro 2\Modul 3\unguided\unguided_rora_220.go"
5 10 15
-15 4 20
0 0
Titik di dalam lingkaran 1 dan 2
PS D:\Praktikum ALpro 2\Modul 3> go run "d:\Praktikum ALpro 2\Modul 3\unguided\unguided_rora_220.go"
1 1 5
8 8 4
15 20
Titik di luar lingkaran 1 dan 2
PS D:\Praktikum ALpro 2\Modul 3> █
```

Deskripsi program

Pada program ini menggunakan Bahasa pemrograman Go yang dimana program ini menentukan posisi sebuah titik (x, y) terhadap dua lingkaran berdasarkan koordinat pusat dan jari-jari masing-masing lingkaran. Program dimulai dengan mendefinisikan fungsi jarak() untuk menghitung jarak antara dua titik, yang kemudian digunakan oleh fungsi diDalamLingkaran_220() untuk mengecek apakah titik tersebut berada di dalam lingkaran tertentu. Program menerima inputan dari pengguna berupa koordinat pusat jari-jari dua lingkaran serta koordinat titik yang ingin diperiksa, lalu program akan mengecek keberadaan titik di dalam lingkaran pertama, lingkaran kedua, kedua lingkaran, atau di luar kedua lingkaran, dan mencetak hasilnya. Outputan program diantaranya “Titik di dalam lingkaran 1”, “Titik di dalam lingkaran 2”, “Titik di dalam lingkaran 1 dan 2”, atau “Titik di luar lingkaran 1 dan 2”. Program lebih memperhitungkan jarak menggunakan rumus $jarak = \sqrt{(a - c)^2 + (b - d)^2}$ Untuk menentukan apakah titik berada dalam lingkaran, dibandingkan dengan jari-jari lingkaran.