

**LAPORAN PRAKTIKUM
ALGORITMA DAN PEMROGRAMAN 2**

MODUL 3

FUNGSI



Disusun Oleh :

M. Faleno Albar Firjatulloh / 2311102297

S1-IF-11-05

Dosen Pengampu :

Arif Amrulloh, S.Kom., M.Kom

PROGRAM STUDI S1 TEKNIK INFORMATIKA

FAKULTAS INFORMATIKA

TELKOM UNIVERSITY PURWOKERTO

2024

I. DASAR TEORI

A. Definisi Fungsi

I. Fungsi adalah kumpulan blok kode yang diberi nama untuk mempermudah penggunaannya. Dengan menerapkan fungsi dengan baik, kode menjadi lebih modular dan mengikuti prinsip DRY (Don't Repeat Yourself). Artinya, kita tidak perlu menulis ulang kode yang sama berkali-kali, cukup mendefinisikan fungsi tersebut sekali, lalu memanggilnya saat diperlukan.

Suatu subprogram dikatakan fungsi apabila:

1. Ada deklarasi tipe nilai yang dikembalikan.
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.

B. Deklarasi Fungsi

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

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

Berikut adalah contoh fungsi untuk menghitung volume dari tabung apabila jari – jari alas dan tinggi tabung diketahui.

	Notasi Algoritma
1	function volumeTabung(jari_jari,tinggi : integer) -> real
2	kamus
3	luasAlas, volume: real
4	algoritma
5	luasAlas <- 3.14 * (jari_jari * jari_jari)
6	volume <- luasAlas * tinggi
7	return volume
8	endfunction
	Notasi dalam bahasa Go
10	func volumeTabung(jari_jari,tinggi int) float64 {
11	var luasAlas,volume float64
12	luasAlas = 3.14 * float64(jari_jari * jari_jari)
13	volume = luasAlas * tinggi
14	return volume
15	}

C. Cara Pemanggilan Fungsi

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 variable, 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	}

II. GUIDED

1. Guided 1

Soal Studi Case

Contoh program dengan function

Sourcecode

```
package main

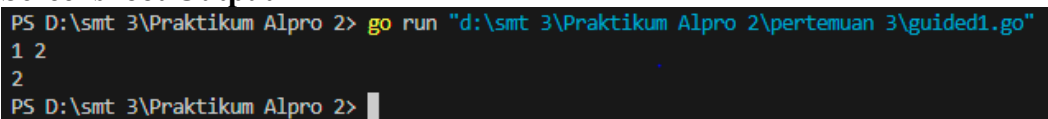
import "fmt"

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 Output



```
PS D:\smt 3\Praktikum Alpro 2> go run "d:\smt 3\Praktikum Alpro 2\pertemuan 3\guided1.go"
1 2
2
PS D:\smt 3\Praktikum Alpro 2> |
```

Deskripsi Program

Program ini menghitung **permutasi** dari dua angka yang dimasukkan pengguna. Jika $a \geq b$, maka menghitung $P(a, b)$, jika tidak, menghitung $P(b, a)$ menggunakan rumus permutasi $P(n, r) = \frac{n!}{(n-r)!}$.

2. Guided 2

Soal Studi Case

Membuat program dengan bahasa go untuk mencari sebuah Luas dan Keliling Persegi.

Sourcecode

```
package main

import (
    "fmt"
)

// Fungsi untuk menghitung luas persegi panjang
func hitungLuas(panjang, lebar float64) float64 {
    return panjang * lebar
}

// Fungsi untuk menghitung keliling persegi
// panjang
func hitungKeliling(panjang, lebar float64) float64 {
    return 2 * (panjang + lebar)
}

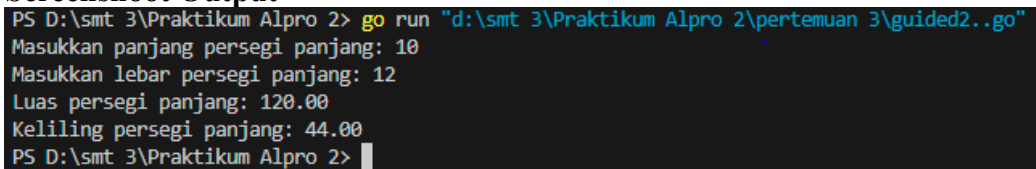
func main() {
    var panjang, lebar float64

    // Input nilai panjang dan lebar
    fmt.Print("Masukkan panjang persegi panjang: ")
    fmt.Scanln(&panjang)
    fmt.Print("Masukkan lebar persegi panjang: ")
    fmt.Scanln(&lebar)

    // Menghitung luas dan keliling
    luas := hitungLuas(panjang, lebar)
    keliling := hitungKeliling(panjang, lebar)
```

```
// Output hasil perhitungan
fmt.Printf("Luas persegi panjang: %.2f\n",
luas)
fmt.Printf("Keliling persegi panjang:
%.2f\n", keliling)
}
```

Screenshoot Output



```
PS D:\smt 3\Praktikum Alpro 2> go run "d:\smt 3\Praktikum Alpro 2\pertemuan 3\guided2..go"
Masukkan panjang persegi panjang: 10
Masukkan lebar persegi panjang: 12
Luas persegi panjang: 120.00
Keliling persegi panjang: 44.00
PS D:\smt 3\Praktikum Alpro 2> |
```

Deskripsi Program

Program ini bertujuan untuk menghitung **luas** dan **keliling** persegi panjang berdasarkan input panjang dan lebar yang diberikan oleh pengguna. Berikut adalah deskripsi singkat dari program ini:

1. **Fungsi hitungLuas:** Fungsi ini menghitung luas persegi panjang dengan rumus $\text{luas} = \text{panjang} \times \text{lebar}$.
2. **Fungsi hitungKeliling:** Fungsi ini menghitung keliling persegi panjang dengan rumus $\text{keliling} = 2 \times (\text{panjang} + \text{lebar})$.
3. **Proses utama:** Program meminta pengguna untuk memasukkan nilai **panjang** dan **lebar**. Setelah itu, program menghitung luas dan keliling menggunakan fungsi yang telah didefinisikan, dan menampilkan hasilnya dalam format dua angka di belakang koma.
4. **Output:** Hasil perhitungan luas dan keliling ditampilkan di layar.

III. UNGUIDED

1. Unguided 1

Soal Studi Case

Minggu ini, mahasiswa Fakultas Informatika mendapatkan tugas dari mata kuliah matematika diskrit untuk mempelajari kombinasi dan permutasi. Jonas salah seorang mahasiswa, iseng untuk mengimplementasikannya ke dalam suatu program. Oleh karena itu bersediakah kalian membantu Jonas? (tidak tentunya ya :p) Masukan terdiri dari empat buah bilangan asli a , b , c , dan d yang dipisahkan oleh spasi, dengan syarat $a \geq c$ dan $b \geq d$. Keluaran terdiri dari dua baris. Baris pertama adalah hasil permutasi dan kombinasi

a terhadap c , sedangkan baris kedua adalah hasil permutasi dan kombinasi b terhadap d .

Catatan: permutasi (P) dan kombinasi (C) dari n terhadap r ($n \geq r$) dapat dihitung dengan menggunakan persamaan berikut!

$$P(n, r) = \frac{n!}{(n-r)!}, \text{ sedangkan } C(n, r) = \frac{n!}{r!(n-r)!}$$

Sourcecode

```
package main

import (
    "fmt"
)

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

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

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

func main() {
    // Input empat bilangan: a, b, c, d
    var a, b, c, d int
    fmt.Print("Masukkan 4 bilangan: ")
    fmt.Scan(&a, &b, &c, &d)

    // Baris pertama: permutasi dan kombinasi a
    terhadap c
    p1 := permutation(a, c)
    c1 := combination(a, c)
```

```

        fmt.Printf("%d %d\n", p1, c1)

        // Baris kedua: permutasi dan kombinasi b
        terhadap d
        p2 := permutation(b, d)
        c2 := combination(b, d)
        fmt.Printf("%d %d\n", p2, c2)
    }

```

Screenshot Output

```

PS D:\smt 3\Praktikum Alpro 2> go run "d:\smt 3\Praktikum Alpro 2\pertemuan 3\unguided1.go"
Masukkan 4 bilangan: 2 3 4 5
2 0
6 0
PS D:\smt 3\Praktikum Alpro 2>

```

Deskripsi Program

Program ini menghitung **permutasi** dan **kombinasi** dari dua pasang bilangan yang dimasukkan pengguna. Setelah menerima input empat bilangan ('a', 'b', 'c', 'd'), program menghitung dan menampilkan permutasi serta kombinasi dari 'a' terhadap 'c', dan dari 'b' terhadap 'd'.

2. Unguided 2

Soal Studi Case

Diberikan tiga buah fungsi matematika yaitu $f(x) = x^2$, $g(x) = x - 2$ dan $h(x) = x + 1$. Fungsi komposisi $(f \circ g \circ h)(x)$ artinya adalah $f(g(h(x)))$. Tuliskan $f(x)$, $g(x)$ dan $h(x)$ dalam bentuk function. Masukan terdiri dari sebuah bilangan bulat a, b dan c yang dipisahkan oleh spasi. Keluaran terdiri dari tiga baris. Baris pertama adalah $(f \circ g \circ h)(a)$, baris kedua $(g \circ h \circ f)(b)$, dan baris ketiga adalah $(h \circ f \circ g)(c)$!

No	Masukan	Keluaran	Penjelasan
1	7 2 10	36 3 65	$(f \circ g \circ h)(7) = 36$ $(g \circ h \circ f)(2) = 3$ $(h \circ f \circ g)(10) = 65$
2	5 5 5	16 24 10	$(f \circ g \circ h)(5) = 16$ $(g \circ h \circ f)(5) = 24$ $(h \circ f \circ g)(5) = 10$
3	3 8 4	4 63 5	$(f \circ g \circ h)(5) = 4$ $(g \circ h \circ f)(5) = 63$ $(h \circ f \circ g)(5) = 5$

Sourcecode

```
package main

import (
    "fmt"
)

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

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

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

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

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

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

func main() {
    // Input tiga bilangan: a, b, c
    var a, b, c int
    fmt.Print("Masukkan 3 bilangan (pisahkan dengan spasi): ")
    fmt.Scan(&a, &b, &c)

    // Output hasil dari komposisi fungsi
    fmt.Println(fogoh(a)) // Hasil (fogoh)(a)
    fmt.Println(gohof(b)) // Hasil (gohof)(b)
}
```

```

        fmt.Println(hofog(c)) // Hasil (hofog) (c)
    }

```

Screenshot Output

```

PS D:\smt 3\Praktikum Alpro 2> go run "d:\smt 3\Praktikum Alpro 2\pertemuan 3\unguided2.go"
Masukkan 3 bilangan (pisahkan dengan spasi): 2 2 7
1
3
26
PS D:\smt 3\Praktikum Alpro 2>

```

Deskripsi Program

Program ini menghitung hasil komposisi tiga fungsi matematika: $f(x) = x^2$, $g(x) = x - 2$, dan $h(x) = x + 1$. Setelah pengguna memasukkan tiga bilangan ('a', 'b', 'c'), program menghitung:

- 'fogoh(a)': Hasil dari $f(g(h(a)))$,
- 'gohof(b)': Hasil dari $g(h(f(b)))$,
- 'hofog(c)': Hasil dari $h(f(g(c)))$.

Hasil dari komposisi fungsi ini kemudian ditampilkan.

3. Unguided 3

Soal Studi Case

[Lingkaran] Suatu lingkaran didefinisikan dengan koordinat titik pusat (cx, cy) dengan radius r. Apabila diberikan dua buah lingkaran, maka tentukan posisi sebuah titik sembarang (x, y) berdasarkan dua lingkaran tersebut. Masukan terdiri dari beberapa tiga baris. Baris pertama dan kedua adalah koordinat titik pusat dan radius dari lingkaran 1 dan lingkaran 2, sedangkan baris ketiga adalah koordinat titik sembarang. Asumsi sumbu x dan y dari semua titik dan juga radius direpresentasikan dengan bilangan bulat. Keluaran berupa string yang menyatakan posisi titik "Titik di dalam lingkaran 1 dan 2", "Titik di dalam lingkaran 1", "Titik di dalam lingkaran 2", atau "Titik di luar lingkaran 1 dan 2".

No	Masukan	Keluaran
1	1 1 5 8 8 4 2 2	Titik di dalam lingkaran 1
2	1 2 3	Titik di dalam lingkaran 2

	4 5 6 7 8	
3	5 10 15 -15 4 20 0 0	Titik di dalam lingkaran 1 dan 2
4	1 1 5 8 8 4 15 20	Titik di luar lingkaran 1 dan 2

Sourcecode

```
package main

import (
    "fmt"
    "math"
)

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

// Fungsi untuk menentukan apakah titik (x, y)
berada di dalam lingkaran dengan pusat (cx, cy)
dan radius r
func dalam(cx, cy, r, x, y float64) bool {
    return jarak(cx, cy, x, y) <= r
}

// Fungsi untuk mengecek posisi titik terhadap
dua lingkaran
func checkPosition(cx1, cy1, r1, cx2, cy2, r2,
x, y float64) string {
    inCircle1 := dalam(cx1, cy1, r1, x, y)
    inCircle2 := dalam(cx2, cy2, r2, x, y)

    if inCircle1 && inCircle2 {
        return "Titik berada di dalam lingkaran
1 dan 2"
    } else if inCircle1 {
        return "Titik berada di dalam lingkaran
1"
    } else if inCircle2 {
```

```

        return "Titik berada di dalam lingkaran
2"
    } else {
        return "Titik berada di luar lingkaran 1
dan 2"
    }
}

func main() {
    var cx1, cy1, r1, cx2, cy2, r2, x, y float64

    // Masukkan data sesuai dengan format yang
    diberikan
    fmt.Println("Masukkan koordinat pusat dan
radius lingkaran 1:")
    fmt.Scan(&cx1, &cy1, &r1)

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

    fmt.Println("Masukkan koordinat titik
sembarang:")
    fmt.Scan(&x, &y)

    // Cek posisi titik sembarang
    result := checkPosition(cx1, cy1, r1, cx2,
cy2, r2, x, y)
    fmt.Println(result)
}

```

Screenshot Output

```

PS D:\smt 3\Praktikum Alpro 2> go run "d:\smt 3\Praktikum Alpro 2\pertemuan 3\unguided3.go"
Masukkan koordinat pusat dan radius lingkaran 1:
2 2 7
Masukkan koordinat pusat dan radius lingkaran 2:
5 1 0
Masukkan koordinat titik sembarang:
1 2 6
Titik berada di dalam lingkaran 1
PS D:\smt 3\Praktikum Alpro 2>

```

Deskripsi Program

Program ini menentukan posisi sebuah titik terhadap dua lingkaran berdasarkan input pengguna. Berikut deskripsinya:

1. **Fungsi jarak:** Menghitung jarak antara dua titik menggunakan rumus jarak Euclidean.

2. **Fungsi dalam:** Mengecek apakah titik (x, y) berada di dalam sebuah lingkaran dengan pusat (cx, cy) dan radius r dengan membandingkan jaraknya dengan radius.
3. **Fungsi checkPosition:** Mengevaluasi apakah titik berada di dalam lingkaran pertama, lingkaran kedua, atau keduanya. Hasil yang mungkin:
 - Titik berada di dalam lingkaran 1 dan 2.
 - Titik berada di dalam lingkaran 1 saja.
 - Titik berada di dalam lingkaran 2 saja.
 - Titik berada di luar kedua lingkaran.
4. **Proses utama:** Program meminta pengguna memasukkan koordinat pusat dan radius untuk dua lingkaran, serta koordinat sebuah titik. Program kemudian mengecek posisi titik tersebut terhadap kedua lingkaran dan menampilkan hasilnya.

IV. KESIMPULAN

Setelah mempelajari fungsi dalam bahasa Go, dapat disimpulkan bahwa fungsi berperan penting dalam membuat kode lebih modular dan mendukung prinsip DRY (Don't Repeat Yourself), sehingga mengurangi pengulangan kode yang sama. Fungsi di Go dapat mengembalikan nilai dengan cara mendeklarasikan tipe nilai yang akan dikembalikan dan menggunakan kata kunci `return` untuk mengirimkannya kembali. Selain itu, fungsi juga dapat memiliki parameter, yang memungkinkan kita untuk memasukkan data ke dalam fungsi dan memprosesnya. Pemanggilan fungsi dilakukan dengan memberikan argumen yang sesuai dengan jumlah dan tipe parameter yang telah didefinisikan. Pada contoh yang dipelajari, penggunaan fungsi built-in seperti `strings.Join()` menunjukkan bagaimana fungsi membantu memanipulasi data dengan lebih efisien. Secara keseluruhan, penerapan fungsi dalam Go sangat penting untuk menjaga kode tetap terorganisir, bersih, dan efisien.

V. REFERENSI

[1] Modul 3 Praktikum Algoritma 2

[2] Novalagung. "Golang Tipe Data." Dasar Pemrograman Golang. Accessed October 6, 2024.
<https://dasarpemrogramangolang.novalagung.com/A-tipe-data.html>