

LAPORAN PRAKTIKUM
ALGORITMA DAN PEMROGRAMAN 1
MODUL 5 & 6
“FOR-LOOP”



DISUSUN OLEH:
ICHYA ULUMIDDIIN
103112400076
S1 IF-12-01
DOSEN:
Yohani Setiya Rafika Nur, M. Kom.

PROGRAM STUDI S1 TEKNIK INFORMATIKA
FAKULTAS INFORMATIKA
TELKOM UNIVERSITY PURWOKERTO
2024/2025

DASAR TEORI

1. Pendahuluan

Perulangan adalah proses mengulang-ulang eksekusi blok kode tanpa henti, selama kondisi yang dijadikan acuan terpenuhi. Biasanya disiapkan variabel untuk iterasi atau variabel penanda kapan perulangan akan diberhentikan.

Di Go keyword perulangan hanya for saja, tetapi meski demikian, kemampuannya merupakan gabungan for, foreach, dan while ibarat bahasa pemrograman lain.

A. Struktur Dasar For Loop

- Inisialisasi: Dieksekusi sekali sebelum perulangan dimulai.
- Kondisi: Dievaluasi sebelum setiap iterasi. Jika bernilai true, blok kode di dalam for loop akan dieksekusi.
- Perubahan: Dieksekusi setelah setiap iterasi, biasanya digunakan untuk memperbarui variabel iterasi

B. Variasi Penggunaan For Loop

- For Loop Tanpa Inisialisasi dan Perubahan: Dapat digunakan seperti while loop di bahasa lain, hanya dengan menyertakan kondisi. Ini berguna untuk perulangan yang bergantung pada kondisi tertentu.
- For Loop Tanpa Kondisi: Ini akan menciptakan perulangan tanpa batas yang harus dihentikan secara manual menggunakan break
- For Loop dengan Range: Digunakan untuk iterasi melalui elemen dalam koleksi seperti array, slice, map, atau channel. Ini mengembalikan indeks dan nilai dari elemen saat ini dalam koleksi.

CONTOH SOAL

1. Latihan 1

Source Code:

```
package main

import "fmt"

func main() {
    var a, b int
    var j int
    fmt.Scan(&a, &b)
    for j = a; j<=b; j = j + 1 {
        fmt.Print(j, " ")
    }
}
```

Output:

```
PS C:\Users\ICHYA ULUMIDDIN\OneDrive\Documents\alpro_1>
go run "c:\Users\ICHYA ULUMIDDIN\OneDrive\Documents\alp
2 5
2 3 4 5
```

Deskripsi Program:

Program ini dibuat dengan tujuan untuk memudahkan pengguna dalam menampilkan deretan bilangan bulat dari angka a hingga b secara otomatis. Dengan memberikan dua bilangan bulat sebagai input, program akan mengeluarkan baris bilangan dari angka awal (a) hingga angka akhir (b) secara berurutan.

2. Latihan 2

Source Code:

```
package main
import "fmt"

func main() {
    var j, alas, tinggi, n int
    var luas float64
    fmt.Scan(&n)
    for j = 1; j <=n; j+=1 {
        fmt.Scan(&alas, &tinggi)
        luas = 0.5 * float64(alas * tinggi)
        fmt.Println(luas)
    }
}
```

Output:

```
PS C:\Users\ICHYA ULUMIDDIN\OneDrive\Documents\alpro_1>
ro_1\modul5&6\coso2\coso2.go"
3
12 32
192
231 234
27027
43 34
731
```

Deskripsi Program:

Program ini bertujuan untuk menghitung dan menampilkan luas segitiga secara otomatis berdasarkan input alas dan tinggi yang diberikan oleh pengguna. Program ini menggunakan pengulangan untuk menghitung luas beberapa segitiga secara berturut-turut.

3. Latihan 3

Source Code:

```
package main

import "fmt"

func main() {
    var j, hasil, v1, v2 int
    fmt.Scan(&v1, &v2)
    for j = 1; j <= v2; j++ {
        hasil = hasil + v1
    }
    fmt.Print(hasil)
}
```

Output:

```
PS C:\Users\ICHYA ULUMIDDIN\OneDrive\Documents\alpro_1>
ro_1\modul5&6\coso3\coso3.go"
2 100
200
```

Deskripsi Program:

Program ini bertujuan untuk menghitung hasil perkalian dua bilangan tanpa menggunakan operator perkalian secara langsung. Sebagai gantinya, konsep dasar penjumlahan berulang digunakan untuk mencapai hasil yang sama.

SOAL LATIHAN

1) Buatlah program untuk menjumlahkan sekumpulan bilangan.

Masukan terdiri dari suatu bilangan bulat positif n .

Keluaran berupa bilangan hasil penjumlahan dari 1 sampai dengan n .

Source Code

```
package main

import "fmt"

func main() {
    var n, sum int
    fmt.Scan(&n)
    for i := 1; i <= n; i++ {
        sum += i
    }
    fmt.Println(sum)
}
```

Output:

```
PS C:\Users\ICHYA ULUMIDDIN\OneDrive\Documents\alpro_1>
ro_1\HOMEWORK_WEEK4(LAB)\latsol1\tempCodeRunnerFile.go"
5
15
```

Deskripsi Program:

Program ini bertujuan untuk menghitung penjumlahan dari serangkaian bilangan bulat positif yang dimulai dari 1 hingga n . Penjumlahan ini dilakukan secara otomatis setelah pengguna memberikan masukan nilai n .

- 2) Buatlah program yang digunakan untuk menghitung volume sejumlah n kerucut, apabila diketahui panjang jari-jari alas kerucut dan tinggi dari kerucut.

Masukan terdiri dari beberapa baris. Baris pertama adalah bilangan bulat n, selanjutnya n baris berikutnya masing-masing merupakan panjang jari-jari alas kerucut dan tinggi dari kerucut.

Keluaran terdiri dari beberapa baris, yang masing-masingnya menyatakan volume dari n kerucut.

Source Code:

```
package main

import "fmt"

func main() {
    var n int
    fmt.Scan(&n)
    const pi = 3.14159
    for i := 1; i <= n; i++ {
        var r, t float64
        fmt.Scan(&r, &t)
        volume := (1.0 / 3.0) * pi * r * r * t
        fmt.Println(volume)
    }
}
```

Output:

```
PS C:\Users\ICHYA ULUMIDDIN\OneDrive\Documents\alpr
ro_1\HOMEWORK_WEEK4(LAB)\latsol2\latsol2.go"
1 3
4
37.69908
```

Deskripsi Program:

Program ini bertujuan untuk menghitung volume sejumlah kerucut berdasarkan input jari-jari alas dan tinggi kerucut. Program ini membantu dalam menyelesaikan masalah perhitungan volume kerucut secara otomatis dengan mengambil beberapa pasangan data input.

- 3) Buatlah program yang digunakan untuk menghitung hasil pemangkatan dari dua buah bilangan. Program dibuat dengan menggunakan operator perkalian dan struktur kontrol perulangan.

Masukan terdiri dari dua bilangan bulat positif.

Keluaran terdiri dari suatu bilangan yang menyatakan hasil bilangan pertama dipangkatkan dengan bilangan kedua.

Source Code:

```
package main

import "fmt"

func main() {
    var bilangan, pangkat, hasil int
    fmt.Scan(&bilangan)
    fmt.Scan(&pangkat)
    hasil = 1
    for i := 1; i <= pangkat; i++ {
        hasil *= bilangan
    }
    fmt.Println(hasil)
}
```

Output:

```
PS C:\Users\ICHYA ULUMIDDIN\OneDrive\Documents\alpro_1>
ro_1\HOMEWORK_WEEK4(LAB)\latso13\latso13.go"
4 2
16
```

Deskripsi Program:

Program ini bertujuan untuk menghitung hasil pemangkatan dua bilangan bulat positif tanpa menggunakan operator pemangkatan bawaan (** atau pow()), tetapi menggunakan operator perkalian dan perulangan untuk menghitung hasilnya.

4) Buatlah program yang digunakan untuk menghitung hasil faktorial dari suatu bilangan.

Masukan terdiri dari suatu bilangan bulat non negatif.

Keluaran terdiri dari hasil faktorial dari bilangan bulat n .

Source Code

```
package main

import "fmt"
func main() {
    var bilangan, hasil int
    fmt.Scan(&bilangan)
    hasil = 1
    for i := 1; i <= bilangan; i++ {
        hasil *= i
    }
    fmt.Println(hasil)
}
```

Output:

```
PS C:\Users\ICHYA ULUMIDDIN\OneDrive\Documents\alpro_1>
ro_1\HOMEWORK_WEEK4(LAB)\latso14\latso14.go"
10
3628800
```

Deskripsi Program:

Program ini dirancang untuk menghitung faktorial dari suatu bilangan bulat non-negatif yang dimasukkan oleh pengguna. Faktorial dari suatu bilangan n !

$n!$ adalah hasil perkalian dari semua bilangan bulat positif dari 1 hingga n

Program ini menggunakan pendekatan berbasis pengulangan (loop) untuk menghitung hasil tersebut.

DAFTAR PUSTAKA

- Agung, N. (2023). *Pemrograman Go Dasar*. Retrieved from Dasar Pemrograman Go Lang: <https://dasarpemrogramangolang.novalagung.com/A-perulangan.html>
- Kurniawan, R. (2022, July 6). *For Loop - Belajar Golang Dari Dasar*. Retrieved from Ruang Developer: <https://blog.ruangdeveloper.com/golang-for-loop/>
- Wagner, L. (2022, October 12). *For Loops in Go*. Retrieved from Boot dev: <https://blog.boot.dev/golang/golang-for-loop/>