

**LAPORAN PRAKTIKUM
ALGORITMA DAN PEMROGRAMAN 1
MODUL 5
“TIPE DATA & VARIABEL”**



DISUSUN OLEH:

RIZKINA AZIZAH

103112400082

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

Loop dalam Bahasa Go

Bahasa Go hanya berisi satu loop Tunggal, yaitu for-loop. For loop adalah struktur kontrol berulang yang memungkinkan kita menulis loop yang mengeksekusi sejumlah kali tertentu

A. Perulangan

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. Perulangan Menggunakan Keyword for

Ada beberapa cara standar menggunakan for. Cara pertama dengan memasukkan variabel counter perulangan beserta kondisinya setelah keyword

b. Penggunaan Keyword for Dengan Argumen Hanya Kondisi

Cara ke-2 adalah dengan menuliskan kondisi setelah keyword for (hanya kondisi). Deklarasi dan iterasi variabel counter tidak dituliskan setelah keyword, hanya kondisi perulangan saja. Konsepnya mirip seperti while milik bahasa pemrograman lain.

c. Penggunaan Keyword for Tanpa Argumen

Cara ke-3 adalah for ditulis tanpa kondisi. Dengan ini akan dihasilkan perulangan tanpa henti (sama dengan for true). Pemberhentian perulangan dilakukan dengan menggunakan keyword break.

d. Penggunaan Keyword for - range

Cara ke-4 adalah perulangan dengan menggunakan kombinasi keyword for dan range. Cara ini biasa digunakan untuk me-looping data gabungan (misalnya string, array, slice, map).

e. Penggunaan Keyword break & continue

Keyword break digunakan untuk menghentikan secara paksa sebuah perulangan, sedangkan continue dipakai untuk memaksa maju ke perulangan berikutnya.

f. Perulangan Bersarang

Tak hanya seleksi kondisi yang bisa bersarang, perulangan juga bisa. Cara pengaplikasiannya kurang lebih sama, tinggal tulis blok statement perulangan di dalam perulangan.

g. Pemanfaatan Label Dalam Perulangan

Di perulangan bersarang, break dan continue akan berlaku pada blok perulangan di mana ia digunakan saja. Ada cara agar kedua keyword ini bisa tertuju pada perulangan terluar atau perulangan tertentu, yaitu dengan memanfaatkan teknik pemberian **label**.

CONTOH SOAL

1. Latihan1

Buatlah program untuk menampilkan baris bilangan dari a sampai dengan b .

Masukan terdiri dari dua bilangan bulat a dan b , yang mana $a \leq b$.

Keluaran berupa baris bilangan dari a sampai dengan b .

Contoh masukan dan keluaran:

NO	Masukan	Keluaran
1	2 5	2 3 4 5
2	6 6	6
3	-5 7	-5 -4 -3 -2 -1 0 1 2 3 4 5 6 7

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\HP\Desktop\tugas> go run "c:\Users\HP\Desktop\tugas\modul 5\tempCodeRunnerFile.go"
2 3 4 5
PS C:\Users\HP\Desktop\tugas> go run "c:\Users\HP\Desktop\tugas\modul 5\tempCodeRunnerFile.go"
6 6
PS C:\Users\HP\Desktop\tugas> go run "c:\Users\HP\Desktop\tugas\modul 5\tempCodeRunnerFile.go"
-5 7
-5 -4 -3 -2 -1 0 1 2 3 4 5 6 7
```

Deskripsi Program:

- Program ini digunakan untuk menampilkan baris bilangan dari a sampai b apabila kita menginputkan a dan b
- Mendeklarasikan tiga variabel yaitu a, b, dan j dengan tipe data integer
- Intruksi Scan digunakan untuk mendeklarasikan input a dan b
- for j = a; j <= b; j = j + 1 deklarasi perulangan
- for j = a; j <= b mendeklarasikan bahwa perulangan dimulai dari a hingga b
- j = j + 1 mendeklarasikan bahwa pada setiap iterasi nilai j akan ditambah satu
- Intruksi Print digunakan untuk mendeklarasikan output j dan tanda petik setelah j (“ “) yang berguna agar setiap mencetak iterasi akan disertai spasi

2. Latihan1

Buatlah program yang digunakan untuk menampilkan sejumlah n luas segitiga, apabila diketahui sisi alas dan tinggi dari masing-masing segitiga.

Masukan terdiri dari $n + 1$ baris. Baris pertama merupakan suatu bilangan bulat n , selanjutnya n baris berikutnya masing-masing merupakan panjang sisi alas dan tinggi dari segitiga

Keluaran terdiri dari n baris, yang masing-masing menyatakan luas dari segitiga.

Contoh masukan dan keluaran:

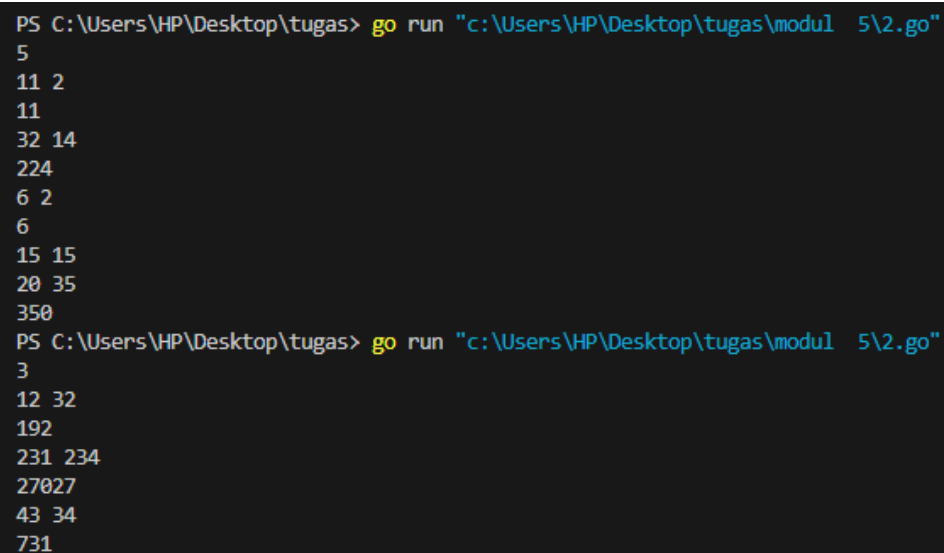
NO	Masukan	Keluaran
1	5 11 2 32 14 6 2 15 15 20 35	11 224 6 112.5 350
2	3 12 32 231 234 43 34	192 27027 731

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\HP\Desktop\tugas> go run "c:\Users\HP\Desktop\tugas\modul 5\2.go"
5
11 2
11
32 14
224
6 2
6
15 15
20 35
350
PS C:\Users\HP\Desktop\tugas> go run "c:\Users\HP\Desktop\tugas\modul 5\2.go"
3
12 32
192
231 234
27027
43 34
731
```

Deskripsi Program:

- Program ini digunakan untuk menampilkan sebuah luas n segitiga
- Mendeklarasikan variabel j, alas, tinggi, dan n dengan tipe data integer, dan variabel luas dengan tipe data real

- Intruksi Scan yang pertama digunakan untuk mendeklarasikan n, n merupakan inputan berapa jumlah segitiga yang ingin dihitung
- for j = 1; j <= n; j += 1 deklarasi perulangan : memulai perulangan dari satu hingga n (1 hingga jumlah segitiga yang ingin dihitung luasnya), dan setiap iterasi akan menghitung luas segitiga
- Intruksi Scan yang kedua digunakan untuk menginput luas dan tinggi
- luas = 0.5 * float64(alas*tinggi) merupakan rumus menghitung luas segitiga yaitu $\frac{1}{2} \times \text{alas} \times \text{tinggi}$
- Intruksi Println digunakan untuk mendeklarasikan output luas

3. Latihan1

Buatlah program yang digunakan untuk menghitung hasil perkalian dua buah bilangan tanpa menggunakan operator kali "*".

Masukan terdiri dari dua bilangan bulat positif.

Keluaran terdiri dari sebuah bilangan yang menyatakan hasil perkalian dari dua bilangan pada masukan.

Contoh masukan dan keluaran:

NO	Masukan	Keluaran
1	2 100	200
2	7 6	42

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\HP\Desktop\tugas> go run "c:\Users\HP\Desktop\tugas\modul 5\3.go"
2 100
200
PS C:\Users\HP\Desktop\tugas> go run "c:\Users\HP\Desktop\tugas\modul 5\3.go"
7 6
42
PS C:\Users\HP\Desktop\tugas> 
```

Deskripsi Program:

- Program ini digunakan untuk menghitung hasil perkalian dua buah bilangan tanpa menggunakan operator kali "*"
- Mendeklarasikan variabel j, hasil, v1, v2 yang menggunakan tipe data integer
- Intruksi Scan digunakan untuk menginput v1 dan v2
- for j = 1; j <= v2; j++ mendeklarasikan perulangan
- for j = 1; j <= v2 mendeklarasikan perulangan dari satu hingga v2
- j++ mendeklarasikan pola perulangan
- hasil = hasil + v1 mendeklarasikan setiap hasil ditambah v1
- Intruksi Print digunakan untuk mendeklarasikan output hasil

SOAL LATIHAN

Statement perulangan

1. 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.

Contoh masukan dan keluaran:

NO	Masukan	Keluaran
1	3	6
2	1	1
3	5	15

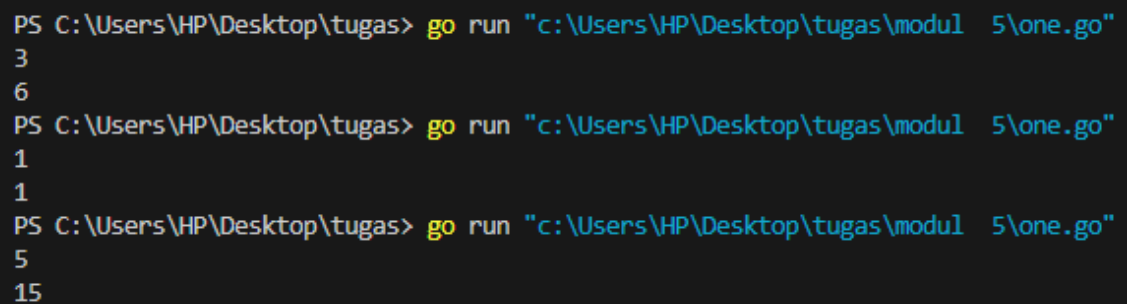
Source Code:

```
package main
import "fmt"

func main() {
    var n, jumlah int
    fmt.Scan(&n)

    for i := 1; i <= n; i++ {
        jumlah += i
    }
    fmt.Print(jumlah)
}
```

Output:



```
PS C:\Users\HP\Desktop\tugas> go run "c:\Users\HP\Desktop\tugas\modul 5\one.go"
3
6
PS C:\Users\HP\Desktop\tugas> go run "c:\Users\HP\Desktop\tugas\modul 5\one.go"
1
1
PS C:\Users\HP\Desktop\tugas> go run "c:\Users\HP\Desktop\tugas\modul 5\one.go"
5
15
```

Deskripsi Program:

- Program ini digunakan untuk menjumlahkan sekumpulan bilangan
- Mendeklarasikan variabel n, jumlah dengan tipe data integer
- Intruksi scan digunakan untuk mendeklarasikan input n
- for i := 1; i <= n mendeklarasikan perulangan dari satu hingga n (i berisi satu hingga n)
- i++ mendeklarasikan pola perulangan
- jumlah += i mendeklarasikan setiap i dijumlahkan
- Intruksi Print mendeklarasikan output jumlah

2. Latihan 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. n kerucut.

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

Contoh masukan dan keluaran

NO	Masukan	Keluaran
1	1 3 4	37.69911184307752
2	3 1 1 2 2 3 3	1.0471975511965976 8.377580409572781 28.274333882308138

Source Code:

```
package main

import "fmt"
import "math"

func main() {
    var i, r, h, n int
    var volume float64

    fmt.Scan(&n)
    for i = 1; i <= n; i += 1 {
        fmt.Scan(&r, &h)
        volume = 1.0/3.0 * math.Pi * math.Pow(float64(r), 2)* float64(h)
        fmt.Println(volume)
    }
}
```

Output:

```
PS C:\Users\HP\Desktop\tugas> go run "c:\Users\HP\Desktop\tugas\modul 5\two.go"
1
3 4
37.699111843077524
PS C:\Users\HP\Desktop\tugas> go run "c:\Users\HP\Desktop\tugas\modul 5\two.go"
3
1 1
1.0471975511965979
2 2
8.377580409572783
3 3
28.27433388230814
```

Deskripsi Program:

- Program ini digunakan untuk menampilkan sebuah volume n kerucut
- Mendeklarasikan varabel j, r, h, dan n dengan tipe data integer, dan variabel volume dengan tipe data real
- Intruksi Scan yang pertama digunakan untuk mendeklarasikan n, n merupakan inputan berapa jumlah kerucut yang ingin dihitung
- for i = 1; i <= n; i += 1 deklarasi perulangan : memulai perulangan dari satu hingga n (1 hingga jumlah segitiga yang ingin dihitung luasnya), dan setiap iterasi akan menghitung volume kerucut
- Intruksi Scan yang kedua digunakan untuk menginput r dan h
- $\text{volume} = 1.0/3.0 * \text{math.Pi} * \text{math.Pow}(\text{float64}(r), 2) * \text{float64}(h)$ merupakan rumus menghitung volume kerucut yaitu $\frac{1}{3} \times \text{phi} \times r^2 \times h$
- Intruksi Println digunakan untuk mendeklarasikan output volume

3. Latihan 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.

Contoh masukan dan keluaran:

NO	Masukan	Keluaran
1	4 2	16
2	2 10	1024
3	10 3	1000

Source Code:

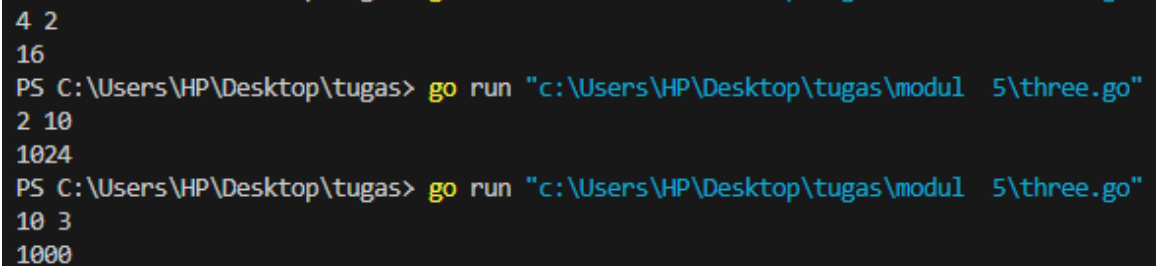
```
package main

import "fmt"

func main() {
    var a, b, pangkat int

    fmt.Scan(&a, &b)
    pangkat = 1
    for i := 1; i <= b; i++ {
        pangkat *= a
    }
    fmt.Print(pangkat)
}
```

Output:



```
4 2
16
PS C:\Users\HP\Desktop\tugas> go run "c:\Users\HP\Desktop\tugas\modul 5\three.go"
2 10
1024
PS C:\Users\HP\Desktop\tugas> go run "c:\Users\HP\Desktop\tugas\modul 5\three.go"
10 3
1000
```

Deskripsi Program:

- Program ini digunakan untuk menghitung hasil pemangkatan dari dua buah bilangan
- Mendeklarasikan variabel a, b, pangkat dengan tipe data integer
- Intruksi Scan digunakan untuk mendeklarasikan input a dan b
- Pangkat mendeklarasikan variabel 1
- for i := 1; i <= b; i++ mendeklarasikan perulangan
- pangkat *= a mendeklarasikan setiap variabel a dikalikan dengan pangkat
- Intruksi Print digunakan untuk mendeklarasikan output pangkat

4. Latihan 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.

Contoh masukan dan keluaran:

NO	Masukan	Keluaran
1	0	1
2	1	1
3	5	120
4	10	3628800

Source Code:

```
package main
import "fmt"

func main() {
    var n, faktorial int
    fmt.Scan(&n)

    faktorial = 1
    for i := 1; i <= n; i++ {
        faktorial *= i
    }
    fmt.Print(faktorial)
}
```

Output:

```
1
1
PS C:\Users\HP\Desktop\tugas> go run "c:\Users\HP\Desktop\tugas\modul 5\four.go"
5
120
PS C:\Users\HP\Desktop\tugas> go run "c:\Users\HP\Desktop\tugas\modul 5\four.go"
10
3628800
PS C:\Users\HP\Desktop\tugas> go run "c:\Users\HP\Desktop\tugas\modul 5\four.go"
0
1
PS C:\Users\HP\Desktop\tugas> 
```

Deskripsi Program:

- Program ini digunakan untuk menghitung hasil faktorial dari suatu bilangan
- Mendeklarasikan variabel n, faktorial dengan tipe data integer
- Intruksi Scan untuk mendeklarasikan input n
- faktorial mendeklarasikan variabel satu
- for i := 1; i <= n; i++ deklarasi perulangan
- faktorial *= 1 mendeklarasikan setiap variabel 1 hingga n dikalikan
- Intruksi Print digunakan untuk mendeklarasikan output faktorial

DAFTAR PUSTAKA

Loop dalam Bahasa Go (<https://www.geeksforgeeks.org/loops-in-go-language/> ,2024)

novalagung(2024), A.14.Perulangan (<https://dasarpemrogramangolang.novalagung.com/A-perulangan.html>,2024)