

LAPORAN PRAKTIKUM
ALGORITMA DAN PEMROGRAMAN
MODUL 5 & 6
FOR & LOOP



DISUSUN OLEH:
MOHAMMAD REYHAN ARETHA FATIN
103112400078
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. Paradigma Perulangan

Perulangan merupakan salah satu kontrol yang memungkinkan suatu instruksi yang sama dilakukan berulang kali dalam waktu atau jumlah yang lama. Tanpa instruksi perulangan, maka suatu intruksi akan ditulis dalam jumlah yang sangat banyak.

2. Karakteristik For-Loop (perulangan berdasarkan iterasi)

Salah satu intruksi perulangan yang paling mudah adalah for-loop, yang mana dengan instruksi ini dapat digunakan untuk mengulangi intruksi sebanyak n kali (iterasi). Batasan besar nilai n menyesuaikan dengan batasan besar nilai dari n menyesuaikan dengan Batasan dari tipe data integer yang di gunakan.

3. Contoh sederhana penggunaan for & loop dalam Bahasa go.

Sebagai contoh penulisan for-loop dengan menggunakan bahasa Go adalah sebagai berikut, yang mana kita akan gunakan contoh menampilkan teks "CAK1BAB3 Algoritma Pemrograman 1" sebanyak 1000x.

```
1 // filename: Alpo1.go
2 package main
3 import "fmt"
4
5 func main() {
6     var iterasi, n int
7     n = 1000
8
9     for iterasi = 1; iterasi <= n; iterasi = iterasi + 1 {
10         fmt.Println(iterasi, "CAK1BAB3 Algoritma Pemrograman 1")
11     }
12 }
```

For Loop juga dapat di gunakan tanpa bagian inisialisasi atau kondisi seperti berikut

```
for {
    fmt.Println("Loop tak terbatas")
}
```

Instruksi for-loop memiliki beberapa komponen, yaitu:

1. inisialisasi merupakan assignment variabel iterasi yang bertipe integer. Pada contoh variabel iterasi = 0 atau 1, artinya iterasi dimulai dari 0 atau 1.
2. kondisi merupakan suatu operasi bernilai boolean yang menyatakan kapan perulangan harus dilakukan. Pada contoh di atas kondisi adalah variabel iterasi <= n (kurang dari atau sama dengan)
3. update merupakan ekspresi yang menyatakan perubahan nilai dari variabel iterasi. Pada contoh di atas update adalah variabel iterasi = variabel iterasi + 1.

CONTOH SOAL

1. Contoh 1

Source Code:

```
coso1 > go coso1.go > ...
1  package main
2
3  import "fmt"
4
5  func main() {
6      var a, b int
7      var j int
8      fmt.Scan(&a, &b)
9      for j = a; j <= b; j = j + 1 {
10         fmt.Print(j, " ")
11     }
12 }
```

Output:

```
PS D:\prak4> go run "d:\prak4\coso1\coso1.go"
2 5
2 3 4 5
PS D:\prak4> go run "d:\prak4\coso1\coso1.go"
6 6
6
PS D:\prak4> go run "d:\prak4\coso1\coso1.go"
-5 7
-5 -4 -3 -2 -1 0 1 2 3 4 5 6 7
PS D:\prak4> █
```

Deskripsi Program:

Buatlah program untuk menampilkan baris bilangan dari a sampai dengan b . Masukan terdiri dari dua bilangan bulat a dan b , yang mana $a \leq b$. dan Keluaran berupa baris bilangan dari a sampai dengan b .

2. Contoh 2

Source Code:

```
1 package main
2
3 import "fmt"
4
5 func main() {
6     var j, alas, tinggi, n int
7     var luas float64
8     fmt.Scan(&n)
9     for j = 1; j <= n; j += 1 {
10         fmt.Scan(&alas, &tinggi)
11         luas = 0.5 * float64(alas*tinggi)
12         fmt.Println(luas)
13     }
14 }
```

Output:

```
PS D:\prak4> go run "d:\prak4\coso2\coso2.go"
5
11 2
11
32 14
224
6 2
6
15 15
112.5
20 35
350
PS D:\prak4> go run "d:\prak4\coso2\coso2.go"
3
12 32
192
231 234
27027
43 34
731
PS D:\prak4> 
```

Deksripsi Program:

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

3. Contoh 3

Source Code:

```
coso3 > go coso3.go > ...
1  package main
2
3  import "fmt"
4
5  func main() {
6      var j, hasil, v1, v2 int
7      fmt.Scan(&v1, &v2)
8      for j = 1; j <= v2; j++ {
9          hasil = hasil + v1
10     }
11     fmt.Print(hasil)
12 }
```

Output:

```
PS D:\prak4> go run "d:\prak4\coso3\coso3.go"
2 100
200
PS D:\prak4> go run "d:\prak4\coso3\coso3.go"
7 6
42
PS D:\prak4> █
```

Deksripsi Program:

Program yang digunakan untuk menghitung hasil perkalian dua buah bilangan tanpa menggunakan operator kali, yaitu menggunakan fungsi for. Masukan terdiri dari dua bilangan bulat positif. dan Keluaran terdiri dari sebuah bilangan yang menyatakan hasil perkalian dari dua bilangan pada masukan.

SOAL LATIHAN

1. Buatlah program untuk menjumlahkan sekumpulan bilangan.

Source Code:

```
latso1 > go latso1.go > main
1  package main
2
3  import "fmt"
4
5  func main() {
6      var hasil, n, j int
7      fmt.Scan(&n)
8      for j = 1; j <= n; j++ {
9          hasil = hasil + j
10     }
11     fmt.Println(hasil)
12 }
13
```

Output:

```
PS D:\prak4> go run "d:\prak4\latso1\latso1.go"
3
6
PS D:\prak4> go run "d:\prak4\latso1\latso1.go"
1
1
PS D:\prak4> go run "d:\prak4\latso1\latso1.go"
5
15
PS D:\prak4> 
```

Deskripsi Program:

Program ini menerima sekumpulan bilangan dan menghitung jumlah totalnya. Input berupa serangkaian bilangan bulat, dan outputnya adalah total penjumlahan dari bilangan-bilangan tersebut.

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

Source Coding:

```
latso2 > go latso2.go > main
1  package main
2
3  import "fmt"
4
5  func main() {
6      var n, j int
7      var phi, volume float64
8      fmt.Scan(&n)
9      phi = 3.14159265358979323849
10     for j = 0; j < n; j++ {
11         var r, t float64
12         fmt.Scan(&r, &t)
13         volume = 1.0 / 3.0 * phi * r * r * t
14         fmt.Printf("%.14f\n", volume)
15     }
16 }
17
```

Output:

```
PS D:\prak4> go run "d:\prak4\latso2\latso2.go"
1
3 4
37.69911184307752
PS D:\prak4> go run "d:\prak4\latso2\latso2.go"
3
1 1
1.04719755119660
2 2
8.37758040957278
3 3
28.27433388230814
PS D:\prak4> █
```

Deksripsi Program:

Program ini menghitung volume dari n kerucut, di mana panjang jari-jari dan tinggi dari setiap kerucut diberikan sebagai input. Input pertama adalah n, yang menunjukkan jumlah kerucut yang dihitung, diikuti oleh n pasang nilai jari-jari dan tinggi. Program ini kemudian mengeluarkan hasil volume masing-masing kerucut menggunakan rumus Volume kerucut.

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

Source Coding:

```
latso3 > go latso3.go > ...
2
3  import "fmt"
4
5  func main() {
6      var j, hasil, bil, pangkat int
7      fmt.Scan(&bil, &pangkat)
8      hasil = 1
9      for j = 0; j < pangkat; j++ {
10         hasil = hasil * bil
11     }
12     fmt.Print(hasil)
13 }
14
```

Output:

```
PS D:\prak4> go run "d:\prak4\latso3\latso3.go"
4 2
16
PS D:\prak4> go run "d:\prak4\latso3\latso3.go"
2 10
1024
PS D:\prak4> go run "d:\prak4\latso3\latso3.go"
10 3
1000
PS D:\prak4> 
```

Deksripsi Program:

Program ini menghitung hasil pemangkatan dari dua bilangan menggunakan perulangan, bukan fungsi pemangkatan bawaan. Input terdiri dari dua bilangan bulat, yaitu bilangan pokok dan pangkatnya, dan outputnya adalah hasil pemangkatan tersebut.

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

Source Coding:

```
latso4 > go latso4.go > main
1  package main
2
3  import "fmt"
4
5  func main() {
6      var n, j, hasil int
7      fmt.Scan(&n)
8      hasil = 1
9      for j = 1; j <= n; j++ {
10         hasil = hasil * j
11     }
12     fmt.Print(hasil)
13 }
14
```

Output:

```
PS D:\prak4> go run "d:\prak4\latso4\latso4.go"
0
1
PS D:\prak4> go run "d:\prak4\latso4\latso4.go"
1
1
PS D:\prak4> go run "d:\prak4\latso4\latso4.go"
5
120
PS D:\prak4> go run "d:\prak4\latso4\latso4.go"
10
3628800
PS D:\prak4> 
```

Deksripsi Program:

Program ini menghitung faktorial dari suatu bilangan bulat. Input adalah satu bilangan bulat n, dan outputnya adalah faktorial dari n,

DAFTAR PUSTAKA

Prasti Eko Yunanto, S.T., M.Kom.

**MODUL PRAKTIKUM 5 & 6 FOR & LOOP ALGORITMA DAN PEMOGRAMAN 1
S1 INFORMATIKA**