

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



DISUSUN OLEH:
ANASTASIA ADINDA NARENDRA INDRIANTO
103112400085
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. Konsep Input dan Output

Input merupakan intruksi dasar untuk membaca data yang diberikan dari pengguna. Data yang diberikan oleh pengguna akan disimpan ke dalam suatu wadah yang disebut *variable*. Penulisan intruksi input beragam menyesuaikan Bahasa pemrograman yang digunakan. Output merupakan perintah untuk menampilkan data ke layar monitor. Data yang sudah diproses atau diolah oleh program komputer perlu ditampilkan ke layar sehingga pengguna bisa memperoleh informasi dari hasil pengolahan data yang dilakukan program.

2. Konsep Data, Variabel, dan Intruksi Dasar

Variabel adalah nama dari suatu lokasi di memori, yang data dengan tipe tertentu dapat disimpan. Nama variabel dimulai dengan huruf dan dapat diikuti dengan sejumlah huruf, angka atau garis bawah. Tipe data yang umum adalah integer, real, Boolean, karakter dan string. Nilai data yang tersimpan dalam variabel dapat diperoleh dengan menyebutkan langsung nama variabelnya. Informasi Alamat atau Lokasi variabel dapat diperoleh dengan menambahkan prefix dan di depan nama variabel tersebut.

3. Konsep Bahasa Pemrograman Go

Bahasa Go menganut kesesuaian tipe data yang ketat. Tipe data yang berbeda tidak boleh dicampur dalam satu ekspresi, bahkan tipe data masih yang sejenis. Menyesuaikan tipe data ada beberapa cara yaitu *casting* tipe (data) mengubah tipe dari data yang diberikan ke tipe data yang diinginkan, memanfaatkan fungsi *Sprint* dan *Sscan* dari paket *fmt*, dan memanfaatkan fungsi-fungsi dalam paket *strconv*, seperti *Atoi*, *Itoa* dan *ParseBool*. Variabel harus dideklarasikan terlebih dahulu sebelum digunakan. Variabel juga harus diinisialisasi dulu agar nilai yang tersimpan diketahui dengan jelas dan eksekusi algoritma menjadi terprediksi. Dalam Bahasa Go, variabel yang tidak diinisialisasi lebih dahulu otomatis diisi dengan nilai default ekuivalen dengan bit 0.

4. Konsep Tipe Data

Melalui ilmu komputer tentunya kita sering mendengar istilah data. Data terbagi menjadi bermacam-macam tipe data yang terklasifikasi dan memiliki fungsi sendiri. Tentunya, berbagai macam tipe data tersebut sangat bermanfaat bagi kinerja komputer melalui kode-kode dalam bahasa pemrograman. Data types atau tipe data adalah sebuah pengklasifikasian data berdasarkan jenis data tersebut. Tipe data dibutuhkan agar kompiler dapat mengetahui bagaimana sebuah data akan digunakan. Untuk mengembangkan sebuah program ada beberapa tipe data yang dibutuhkan. Tipe data memiliki 4 data type yang sering digunakan yaitu,

- 1) Bilangan Bulat (Integer) Tipe bilangan bulat (Integer) adalah tipe data numerik yang biasa digunakan apabila bertemu dengan bilangan bulat, seperti 1, 27, 100, dll. Bilangan ini juga mengenal nilai positif dan negatif. Tipe data numerik yang termasuk ke dalam bilangan bulat adalah *byte*, *short*, *int*, dan *long*.
- 2) Bilangan Pecahan (Floating Point) Tipe bilangan pecahan atau *floating point* adalah bilangan yang menangani bilangan desimal atau perhitungan secara detail. Karena kemampuannya, float point berbanding terbalik dengan integer. Terdapat dua tipe pada bilangan pecahan ini yaitu *float* dan *double*.
- 3) Karakter (Char)
Tipe data karakter tunggal yang biasa didefinisikan dengan tanda petik (') di awal dan di akhir karakternya. Tipe ini mengikuti aturan "unicode" sehingga bilangan harus

diawali kode “/u”. Tetapi juga biasa menggunakan bilangan heksadesimal dari 0000 sampai FFFF.

4) Boolean

Tipe data *boolean* merupakan tipe yang memiliki dua nilai yaitu benar (*true*) atau salah (*false*). Nilai yang digunakan pada tipe ini sangat penting dalam mengambil keputusan suatu kejadian tertentu.

5. **Deklarasi dan Inisialisasi**

Deklarasi variabel dalam bahasa pemrograman komputer adalah pernyataan yang digunakan untuk menentukan nama variabel dan tipe datanya. Deklarasi memberi tahu kompiler tentang keberadaan entitas dalam program dan lokasinya. Saat Anda mendeklarasikan variabel, Anda juga harus menginisialisasinya. Inisialisasi adalah proses pemberian nilai pada Variabel. Setiap bahasa pemrograman memiliki metode tersendiri untuk menginisialisasi variabel. Jika nilai tidak diberikan pada Variabel, maka proses tersebut hanya disebut Deklarasi.

6. **Konstanta dalam Bahasa Pemrograman**

Konstanta adalah entitas yang merujuk pada nilai data yang tetap dan tidak dapat diubah. Selama eksekusi/perhitungan pemrograman, nilai suatu konstanta tidak dapat diubah, tetap konstan. Jenis Konstanta ada konstanta Bilangan Bulat. Konstanta bilangan bulat adalah barisan bilangan bulat yang nilainya tetap. Mereka tidak boleh mengandung koma desimal atau angka pecahan. Konstanta bilangan bulat dapat berupa bilangan positif atau negatif. Mereka termasuk bilangan bulat sistem desimal, bilangan bulat sistem oktal, bilangan bulat sistem heksadesimal.

7. **Integer Division dan Modulo**

Pembagian pada tipe data integer (**integer division** atau **div**) sedikit berbeda dengan pembagian yang sudah sering kita pelajari dari sekolah dasar. Hasil pembagian akan bertipe data integer, artinya kita akan mengabaikan bilangan yang muncul setelah tanda koma atau floating point. Hasil dari operasi div ini dikenal dengan istilah **quotient**. Modulo atau modulus "mod" adalah operasi untuk mencari nilai integer dari sisa pembagian pada integer division, biasanya dikenal juga dengan istilah remainder. Misalnya untuk operasi 10 dibagi dengan 3, maka sisa pembagian adalah 1. Operator modulo menggunakan "%" di dalam bahasa pemrograman Go.

8. **Casting atau Konversi Tipe Data**

Pada Bahasa pemrograman Go, tipe data bersifat statis, artinya tipe data yang sudah didefinisikan tidak dapat diganti selama program berjalan. Casting merupakan salah satu Teknik konversi antar tipe data di dalam Bahasa pemrograman.

9. **Paradigma Perulangan**

Perulangan merupakan salah satu struktur kontrol yang memungkinkan suatu instruksi yang sama dilakukan berulang kali dalam waktu atau jumlah yang lama. Tanpa instruksi perulangan, maka suatu instruksi akan ditulis dalam jumlah yang sangat banyak. Salah satu instruksi perulangan yang paling mudah adalah **for-loop**, yang mana dengan instruksi ini dapat digunakan untuk mengulangi instruksi sebanyak **n** kali (iterasi). Batasan besar nilai dari **n** menyesuaikan dengan batasan dari tipe data integer yang digunakan.

CONTOH SOAL

1. Contoh Latihan Soal 1

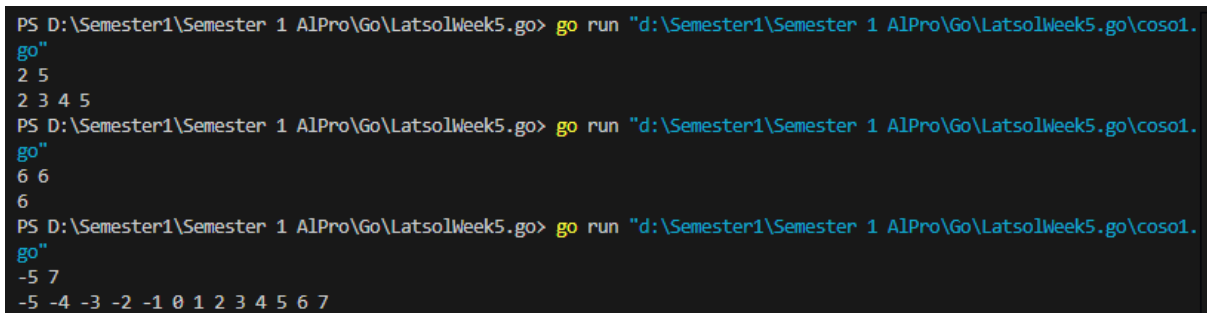
Source Code:

```
package main

import "fmt"

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

Output:



```
PS D:\Semester1\Semester 1 AlPro\Go\LatsolWeek5.go> go run "d:\Semester1\Semester 1 AlPro\Go\LatsolWeek5.go\coso1.go"
2 5
2 3 4 5
PS D:\Semester1\Semester 1 AlPro\Go\LatsolWeek5.go> go run "d:\Semester1\Semester 1 AlPro\Go\LatsolWeek5.go\coso1.go"
6 6
6
PS D:\Semester1\Semester 1 AlPro\Go\LatsolWeek5.go> go run "d:\Semester1\Semester 1 AlPro\Go\LatsolWeek5.go\coso1.go"
-5 7
-5 -4 -3 -2 -1 0 1 2 3 4 5 6 7
```

Deskripsi Program:

Program coso1.go dibuat untuk menampilkan baris bilangan dari a sampai dengan b. Masukan terdiri dari dua bilangan bulat a dan b, yang mana a kurang dari sama dengan b. Outputnya berupa baris bilangan dari a sampai dengan b. User dapat menginputkan data yang diinginkan didalam bagian terminal setelah program dijalankan dengan runner program. Berikut input dan output;

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

2. Contoh Latihan Soal 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 D:\Semester1\Semester 1 AlPro\Go\LatsolWeek5.go> go run "d:\Semester1\Semester 1 AlPro\Go\LatsolWeek5.go
\coso2.go"
5
11 2
11
32 14
224
6 2
6
15 15
112.5
20 35
350
PS D:\Semester1\Semester 1 AlPro\Go\LatsolWeek5.go> go run "d:\Semester1\Semester 1 AlPro\Go\LatsolWeek5.go
\coso2.go"
3
12 32
192
231 234
27027
43 34
731
```

Deskripsi Program:

Program coso2.go dibuat dengan tujuan menampilkan sejumlah n luas segitiga, apabila diketahui sisi alas dan tinggi dari masing-masing segitiga. Input 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. Output terdiri dari n baris, yang masing-masing menyatakan luas dari segitiga. User dapat menginputkan data yang diinginkan didalam bagian terminal setelah program dijalankan dengan runner program. Berikut input dan ouput;

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

3. Contoh Latihan Soal 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:

```
Go\LatsolWeek5.go> go run "d:\Semester1\Semester 1 AlPro\Go\LatsolWeek5.go\coso3.go"
2 100
200
PS D:\Semester1\Semester 1 AlPro\Go\LatsolWeek5.go> go run "d:\Semester1\Semester 1 AlPro\Go\LatsolWeek5.go\coso3.go"
7 6
42
PS D:\Semester1\Semester 1 AlPro\Go\LatsolWeek5.go> |
```

Deskripsi Program:

Program coso3.go dibuat dengan tujuan hasil perkalian dua buah bilangan tanpa menggunakan operator “ * “. Input terdiri dari dua bilangan bulat positif. Output terdiri dari sebuah bilangan yang menyatakan hasil perkalian dari dua bilangan pada masukan. User

dapat menginputkan data yang diinginkan didalam bagian terminal setelah program dijalankan dengan runner program. Berikut input dan ouput;

No	Masukan	Keluaran
1	2 100	200
2	7 6	42

SOAL LATIHAN

1. Latihan Soal 1

Source Code:

```
package main

import "fmt"

func main() {
    var bilangan int
    var hasil int = 0
    fmt.Scan(&bilangan)

    for i := 1; i <= bilangan; i++ {
        hasil += i
    }
    fmt.Println(hasil)
}
```

Output:

```
PS D:\Semester1\Semester 1 AlPro\Go\LatsolWeek5.go> go run "d:\Semester1\Semester 1 AlPro\Go\LatsolWeek5.go\Latsol1.go"
3
6
PS D:\Semester1\Semester 1 AlPro\Go\LatsolWeek5.go> go run "d:\Semester1\Semester 1 AlPro\Go\LatsolWeek5.go\Latsol1.go"
1
1
PS D:\Semester1\Semester 1 AlPro\Go\LatsolWeek5.go> go run "d:\Semester1\Semester 1 AlPro\Go\LatsolWeek5.go\Latsol1.go"
5
15
```

Deskripsi Program:

Program Latsol1.go dibuat dengan tujuan menjumlahkan sekumpulan bilangan. Input terdiri dari suatu bilangan bulat n positif. Input berupa hasil penjumlahan dari 1 sampai dengan n. User dapat menginputkan data yang diinginkan didalam bagian terminal setelah program dijalankan dengan runner program. Berikut input dan ouput;

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

2. Latihan Soal 2

Source Code:

```
package main

import (
    "fmt"
    "math"
)

func main() {
    var n int
    var V, r, t float64
    fmt.Scanln(&n)
    for i := 1; i <= n; i++ {
        fmt.Scanln(&r, &t)

        V = (1.0 / 3.0) * (math.Pi * r * r * t)
        fmt.Println(V)
    }
}
```

Output:

```
PS D:\Semester1\Semester 1 AlPro\Go\LatsolWeek5.go> go run "d:\Semester1\Semester 1 AlPro\Go\LatsolWeek5.go\Latsol2.go"
1
3 4
37.69911184307752
PS D:\Semester1\Semester 1 AlPro\Go\LatsolWeek5.go> go run "d:\Semester1\Semester 1 AlPro\Go\LatsolWeek5.go\Latsol2.go"
3
1 1
1.0471975511965976
2 2
8.377580409572781
3 3
28.274333882308134
```

Deskripsi Program:

Program Latsol2.go dibuat dengan tujuan menghitung volume sejumlah n kerucut, apabila diketahui Panjang jari-jari alas kerucut dan tinggi dari kerucut. Input terdiri dari beberapa baris. Baris pertama adalah bilanganbukat n, selanjutnya n baris berikutnya masing-masing

merupakan Panjang jari-jari alas kerucut dan tinggi dari kerucut. Output terdiri dari beberapa baris, yang masing-masing menyatakan volume dari n kerucut. User dapat menginputkan data yang diinginkan didalam bagian terminal setelah program dijalankan dengan runner program. Berikut input dan ouput;

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

3. Latihan Soal 3

Source Code:

```
package main

import "fmt"

func main() {
    var bilangan1, bilangan2, hasil int
    fmt.Scanln(&bilangan1, &bilangan2)
    for i := 0; i < bilangan2; i++ {
        if i == 0 {
            hasil = bilangan1
        } else {
            hasil = hasil * bilangan1
        }
    }
    fmt.Print(hasil)
}
```

Output:

```
PS D:\Semester1\Semester 1 AlPro\Go\LatsolWeek5.go> go run "d:\Semester1\Semester 1 AlPro\Go\LatsolWeek5.go\Latsol3.go"
4 2
16
PS D:\Semester1\Semester 1 AlPro\Go\LatsolWeek5.go> go run "d:\Semester1\Semester 1 AlPro\Go\LatsolWeek5.go\Latsol3.go"
2 10
1024
PS D:\Semester1\Semester 1 AlPro\Go\LatsolWeek5.go> go run "d:\Semester1\Semester 1 AlPro\Go\LatsolWeek5.go\Latsol3.go"
10 3
1000
```

Deskripsi Program:

Program Latsol3.go dibuat dengan tujuan menghitung hasil pemangkatan dari dua buah bilangan. Program dibuat dengan menggunakan operator perkalian dan struktur control

perulangan. Input terdiri dari dua bilangan bulat positif. Output terdiri dari suatu bilangan yang menyatakan hasil bilangan pertama dipangkatkan dengan bilangan kedua. User dapat menginputkan data yang diinginkan didalam bagian terminal setelah program dijalankan dengan runner program. Berikut input dan ouput;

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

4. Latihan Soal 4

Source Code:

```
package main

import "fmt"

func main() {
    var bilangan int
    var hasil int = 1
    fmt.Scan(&bilangan)

    for i := 1; i <= bilangan; i++ {
        hasil = hasil * i
    }
    fmt.Println(hasil)
}
```

Output:

```
PS D:\Semester1\Semester 1 APro\Go\LatsolWeek5.go> go run "d:\Semester1\Semester 1 APro\Go\LatsolWeek5.go\Latsol4.go"
0
1
PS D:\Semester1\Semester 1 APro\Go\LatsolWeek5.go> go run "d:\Semester1\Semester 1 APro\Go\LatsolWeek5.go\Latsol4.go"
1
1
PS D:\Semester1\Semester 1 APro\Go\LatsolWeek5.go> go run "d:\Semester1\Semester 1 APro\Go\LatsolWeek5.go\Latsol4.go"
5
120
PS D:\Semester1\Semester 1 APro\Go\LatsolWeek5.go> go run "d:\Semester1\Semester 1 APro\Go\LatsolWeek5.go\Latsol4.go"
10
3628800
```

Deskripsi Program:

Program Latsol4.go dibuat dengan tujuan menghitung hasil faktorial dari suatu bilangan. Input terdiri dari suatu bilangan bulat non negatif. Input terdiri dari hasil faktorial dari

bilangan bulat n. User dapat memasukkan data yang diinginkan didalam bagian terminal setelah program dijalankan dengan runner program. Berikut input dan ouput;

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

DAFTAR PUSTAKA

<file:///D:/Semester%201%20AIPro/Modul/MODUL%202.pdf>

<https://www.toppr.com/guides/computer-science/introduction-to-c/data-types-variables-and-constants/declaration-of-variables/>

<https://www.toppr.com/guides/computer-science/introduction-to-c/data-types-variables-and-constants/constants-in-programming-language/>

<https://www.dicoding.com/blog/macam-macam-tipe-data/>

<file:///D:/Semester%201%20AIPro/Modul/MODUL%203.pdf>

<file:///D:/Semester1/Semester%201%20AIPro/Modul/MODUL%205%20dan%206.pdf>