

LAPORAN PRAKTIKUM
ALGORITMA DAN PEMROGRAMAN 1
MODUL 13
“TIPE DATA & VARIABEL”



DISUSUN OLEH:
MUHAMMAD ZAKY MUBAROK
103112400073 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

Dasar Teori Pemrosesan Sekuensial

1. Pembacaan Data Tanpa Marker: Semua data yang diberikan diproses.

- Notasi Algoritma:

pseudo

input(n)

i = 1

while i <= n do

input(dat)

{ proses dat }

i = i + 1

endwhile

- Notasi dalam Go:

go

fmt.Scanln(&n)

i = 0

for i < n {

fmt.Scan(&dat)

// proses dat

i = i + 1

}

2. Pembacaan Data dengan Marker: Data terakhir adalah marker untuk menghentikan perulangan.

- Notasi Algoritma:

pseudo

input(dat)

while dat != MARKER do

{ proses dat }

```
    input(dat)
endwhile

    • Notasi dalam Go:
```

```
go
fmt.Scanln(&dat)
for dat != MARKER {
    // proses dat
    fmt.Scanln(&dat)
}
```

3. Kemungkinan Rangkaian Data Kosong: Menangani kasus di mana data pertama adalah marker.

```
    • Notasi Algoritma:

pseudo
input(dat)
if dat == MARKER then
    { kode untuk data kosong }
else
    while dat != MARKER do
        { proses dat }
        input(dat)
    endwhile
endif
```

```
    • Notasi dalam Go:

go
fmt.Scanln(&dat)
if dat == MARKER {
    // kode untuk data kosong
} else {
    for dat != MARKER {
```

```

    // proses dat
    fmt.Scanln(&dat)
}
}

```

4. Elemen Pertama Perlu Diproses Tersendiri: Data pertama diproses sebelum perulangan.

- **Notasi Algoritma:**

pseudo

input(dat)

if dat == MARKER then

{ kode untuk data kosong }

else

{ proses data pertama }

input(dat)

while dat != MARKER do

{ proses dat }

input(dat)

endwhile

endif

- **Notasi dalam Go:**

go

fmt.Scanln(&dat)

if dat == MARKER {

// kode untuk data kosong

} else {

// proses data pertama

fmt.Scan(&dat)

for dat != MARKER {

// proses dat

```
fmt.Scanln(&dat)
```

```
}
```

```
}
```

LATIHAN SOAL

1. Latihan1

Source Code:

```
package main

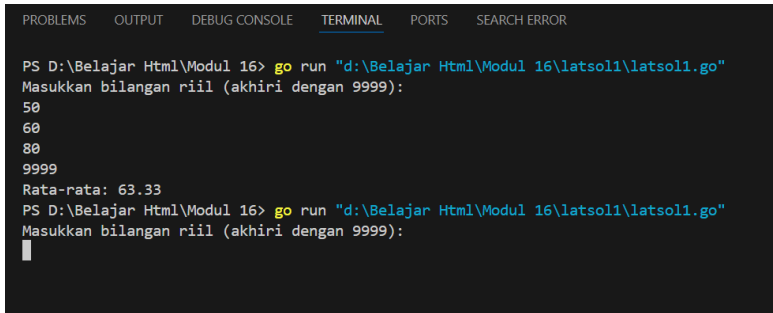
import (
    "fmt"
)

func main() {
    var jumlah, hitungan float64
    var masukan float64

    fmt.Println("Masukkan bilangan riil (akhiri dengan 9999):")
    for {
        fmt.Scan(&masukan)
        if masukan == 9999 {
            break
        }
        jumlah += masukan
        hitungan++
    }

    if hitungan > 0 {
        fmt.Printf("Rata-rata: %.2f\n", jumlah/hitungan)
    } else {
        fmt.Println("Tidak ada bilangan yang dimasukkan.")
    }
}
```

Output:



The screenshot shows a Go IDE interface with a terminal window. The terminal displays the following output:

```
PS D:\Belajar Html\Modul 16> go run "d:\Belajar Html\Modul 16\latsol1\latsol1.go"
Masukkan bilangan riil (akhiri dengan 9999):
50
60
80
9999
Rata-rata: 63.33
PS D:\Belajar Html\Modul 16> go run "d:\Belajar Html\Modul 16\latsol1\latsol1.go"
Masukkan bilangan riil (akhiri dengan 9999):
```

1. Deklarasi variabel:

var jumlah, hitungan, masukan float64

Program mendeklarasikan tiga variabel bertipe float64: jumlah untuk menyimpan total nilai input yang dijumlahkan, hitungan untuk menghitung jumlah bilangan yang diinputkan, dan masukan untuk menyimpan bilangan yang diinputkan.

2. Instruksi input:

fmt.Println("Masukkan bilangan riil (akhiri dengan 9999):")

Program menampilkan pesan untuk menginstruksikan input bilangan riil dan menyebutkan bahwa input berakhir dengan memasukkan "9999".

3. Perulangan untuk membaca input:

```
for {  
    fmt.Scan(&masukan)  
    if masukan == 9999 {  
        break  
    }  
    jumlah += masukan  
    hitungan++  
}
```

Program menjalankan loop tak terbatas (for {}), di mana setiap iterasi membaca input bilangan. Jika bilangan yang dimasukkan adalah "9999", loop dihentikan dengan break. Jika tidak, nilai input ditambahkan ke jumlah dan hitungan ditambah 1.

4. Menghitung rata-rata dan mencetak hasil:

```
if hitungan > 0 {  
    fmt.Printf("Rata-rata: %.2f\n", jumlah/hitungan)  
} else {  
    fmt.Println("Tidak ada bilangan yang dimasukkan.")  
}
```

Setelah loop selesai, program memeriksa apakah ada bilangan yang diinputkan (hitungan > 0). Jika ada, program menghitung rata-rata dengan membagi jumlah dengan hitungan dan mencetak hasilnya dalam format dua angka desimal. Jika tidak

ada bilangan yang diinputkan, program mencetak pesan bahwa tidak ada bilangan yang dimasukkan.

Secara keseluruhan, program ini efektif untuk menghitung dan mencetak rata-rata dari sejumlah bilangan riil yang diinputkan sampai "9999" dimasukkan sebagai penanda akhir

LATIHAN SOAL

2. Latihan2

Source Code:

```
package main

import "fmt"

func main() {
    var kata, s1, s2, s3, s4, s5 string
    var jumlah int

    fmt.Print("Masukkan string kata: ")
    fmt.Scanln(&kata)

    fmt.Print("Masukkan jumlah string jumlah (Max 5 ya): ")
    fmt.Scanln(&jumlah)

    fmt.Println("Masukkan string:")
    if jumlah >= 1 {
        fmt.Scanln(&s1)
    }
    if jumlah >= 2 {
        fmt.Scanln(&s2)
    }
    if jumlah >= 3 {
        fmt.Scanln(&s3)
    }
    if jumlah >= 4 {
        fmt.Scanln(&s4)
    }
    if jumlah >= 5 {
        fmt.Scanln(&s5)
    }

    ditemukan := false
    posisi := -1
    hitung := 0

    if jumlah >= 1 && s1 == kata {
        hitung++
        if !ditemukan {
            ditemukan = true
        }
    }
}
```

```

        posisi = 1
    }
}
if jumlah >= 2 && s2 == kata {
    hitung++
    if !ditemukan {
        ditemukan = true
        posisi = 2
    }
}
if jumlah >= 3 && s3 == kata {
    hitung++
    if !ditemukan {
        ditemukan = true
        posisi = 3
    }
}
if jumlah >= 4 && s4 == kata {
    hitung++
    if !ditemukan {
        ditemukan = true
        posisi = 4
    }
}
if jumlah >= 5 && s5 == kata {
    hitung++
    if !ditemukan {
        ditemukan = true
        posisi = 5
    }
}
fmt.Println()

if ditemukan {
    fmt.Println("a. String kata ada dalam kumpulan jumlah data string tersebut.")
} else {
    fmt.Println("a. String kata tidak ada dalam kumpulan jumlah data string
tersebut.")
}

if ditemukan {
    fmt.Printf("b. String kata ditemukan di posisi ke-%d.\n", posisi)
} else {

```

```

        fmt.Println("b. String kata tidak ditemukan.")
    }

    fmt.Printf("c. String kata muncul sebanyak %d kali dalam kumpulan jumlah data
string tersebut.\n", hitung)

    if hitung >= 2 {
        fmt.Println("d. Terdapat sedikitnya dua string kata dalam jumlah data string
tersebut.")
    } else {
        fmt.Println("d. Tidak terdapat sedikitnya dua string kata dalam jumlah data
string tersebut.")
    }
}

```

Output:

```

Go Run ... Modul 16
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS SEARCH ERROR

PS D:\Belajar Html\Modul 16> go run "d:\Belajar Html\Modul 16\latsol2\latsol2.go"
Masukkan string x: 5
Masukkan jumlah string n (Max 5 ya) : 5
Masukkan string :
1
2
3
4
5

a. String x ada dalam kumpulan n data string tersebut.
b. String x ditemukan di posisi ke-5.
c. String x muncul sebanyak 1 kali dalam kumpulan n data string tersebut.
d. Tidak terdapat sedikitnya dua string x dalam n data string tersebut.
PS D:\Belajar Html\Modul 16> go run "d:\Belajar Html\Modul 16\latsol2\latsol2.go"
Masukkan string x: 7
Masukkan jumlah string n (Max 5 ya) : 4
Masukkan string :
3
7
6
8

a. String x ada dalam kumpulan n data string tersebut.
b. String x ditemukan di posisi ke-2.
c. String x muncul sebanyak 1 kali dalam kumpulan n data string tersebut.
d. Tidak terdapat sedikitnya dua string x dalam n data string tersebut.
PS D:\Belajar Html\Modul 16> go run "d:\Belajar Html\Modul 16\latsol2\latsol2.go"

```

Deskripsi Program:

Deklarasi variabel:

var kata, s1, s2, s3, s4, s5 string

var jumlah int

Program mendeklarasikan variabel kata untuk menyimpan string yang dicari, s1 hingga s5 untuk menyimpan string yang diinputkan, dan jumlah untuk menyimpan jumlah string yang akan diinputkan.

Membaca input:

fmt.Print("Masukkan string kata: ")

fmt.Scanln(&kata)

fmt.Print("Masukkan jumlah string jumlah (Max 5 ya): ")

fmt.Scanln(&jumlah)

fmt.Println("Masukkan string:")

if jumlah >= 1 {

fmt.Scanln(&s1)

}

if jumlah >= 2 {

fmt.Scanln(&s2)

}

if jumlah >= 3 {

fmt.Scanln(&s3)

}

if jumlah >= 4 {

fmt.Scanln(&s4)

}

```
if jumlah >= 5 {  
    fmt.Scanln(&s5)  
}
```

Program membaca input string kata dan jumlah string jumlah yang akan diinputkan. Kemudian, program membaca jumlah string berdasarkan nilai jumlah.

Inisialisasi variabel hasil:

```
ditemukan := false  
posisi := -1  
hitung := 0
```

Program mendeklarasikan variabel ditemukan untuk menandai apakah kata ditemukan, posisi untuk menyimpan posisi pertama kali kata ditemukan, dan hitung untuk menghitung berapa kali kata ditemukan.

Memeriksa setiap string yang diinputkan:

```
if jumlah >= 1 && s1 == kata {  
    hitung++  
    if !ditemukan {  
        ditemukan = true  
        posisi = 1  
    }  
}  
  
if jumlah >= 2 && s2 == kata {  
    hitung++  
    if !ditemukan {  
        ditemukan = true  
        posisi = 2  
    }  
}  
  
if jumlah >= 3 && s3 == kata {
```

```

    hitung++
    if !ditemukan {
        ditemukan = true
        posisi = 3
    }
}
if jumlah >= 4 && s4 == kata {
    hitung++
    if !ditemukan {
        ditemukan = true
        posisi = 4
    }
}
if jumlah >= 5 && s5 == kata {
    hitung++
    if !ditemukan {
        ditemukan = true
        posisi = 5
    }
}

```

Program memeriksa setiap string yang diinputkan (s1 hingga s5) apakah sama dengan kata. Jika sama, hitung ditambah 1 dan posisi diset dengan posisi pertama kali kata ditemukan.

Mencetak hasil:

```
fmt.Println()
```

```

if ditemukan {
    fmt.Println("a. String kata ada dalam kumpulan jumlah data string tersebut.")
} else {
    fmt.Println("a. String kata tidak ada dalam kumpulan jumlah data string tersebut.")
}

```

```
}
```

```
if ditemukan {
```

```
    fmt.Printf("b. String kata ditemukan di posisi ke-%d.\n", posisi)
```

```
} else {
```

```
    fmt.Println("b. String kata tidak ditemukan.")
```

```
}
```

```
fmt.Printf("c. String kata muncul sebanyak %d kali dalam kumpulan jumlah data  
string tersebut.\n", hitung)
```

```
if hitung >= 2 {
```

```
    fmt.Println("d. Terdapat sedikitnya dua string kata dalam jumlah data string  
tersebut.")
```

```
} else {
```

```
    fmt.Println("d. Tidak terdapat sedikitnya dua string kata dalam jumlah data string  
tersebut.")
```

```
}
```

Program mencetak hasil dengan menjawab pertanyaan:

- a. Apakah kata ada dalam kumpulan jumlah data string?
- b. Pada posisi ke berapa kata ditemukan pertama kali?
- c. Berapa kali kata muncul dalam kumpulan jumlah data string?
- d. Apakah terdapat sedikitnya dua kata dalam kumpulan jumlah data string?

DAFTAR PUSTAKA

MODUL PRAKTIKUM 16 - SKEMA PEMROSESAN SEKUENSIAL