

**LAPORAN PRAKTIKUM**  
**ALGORITMA DAN PEMROGRAMAN**  
**MODUL 16**  
**SKEMA PEMROSESAN SEKUENSIAL**



**DISUSUN OLEH:**  
**HISYAM NURDIATMOKO**  
**103112400049**  
**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**

### **16.1 Pengantar Skema Pemrosesan Sekuensial**

Dengan dipersenjatai bentuk perulangan dan bentuk percabangan, banyak problem komputasi yang dapat diselesaikan. Berikut ini beberapa skema (pola) yang umum ditemukan untuk pemrosesan data (secara sekuensial).

### **16.2 Pembacaan Data Tanpa Marker pada Akhir Rangkaian Data**

Pola ini memperlihatkan bahwa semua data yang diberikan pada masukan adalah data yang harus diproses.

### **16.3 Pembacaan Data dengan Marker pada Akhir Rangkaian Data**

Pada pola dengan marker, terdapat data yang dipersiapkan khusus untuk menghentikan perulangan. Artinya semua data yang diberikan pada masukan adalah data yang valid, kecuali data yang terakhir, karena digunakan untuk menghentikan perulangan.

### **16.4 Kemungkinan Rangkaian Data Kosong, Hanya Ada Marker**

Pola dengan marker pada contoh di atas memungkinkan terjadi bahwa data pertama yang diberikan pada masukan adalah marker, artinya tidak ada satu data pun yang valid. Kemungkinan ini disebut juga rangkaian data kosong atau kasus kosong.

### **16.5 Elemen Pertama Perlu Diproses Tersendiri/Kasus Khusus**

Pada pola ini data pertama diproses terlebih dahulu secara khusus sebelum perulangan dilakukan. Apabila melihat contoh pencarian nilai maksimum di atas, terlihat bahwa nilai variabel `max` selalu diinisialisasi oleh sebuah nilai `BILANGAN KECIL` berapapun. Kekurangan dari pendekatan ini adalah kita harus mengetahui secara pasti nilai-nilai yang mungkin ada pada variabel `dat`, yang mana nilai pada variabel `dat` tersebut **TIDAK BOLEH** lebih kecil dibandingkan nilai dari `BILANGAN KECIL` yang digunakan saat inisialisasi. Sebagai contoh mencari nilai temperatur atau suhu maksimum. Pada kasus ini, berapa nilai dari `BILANGAN KECIL` yang akan digunakan? Tentu kita akan kesulitan menentukannya, karena temperatur bisa bernilai negatif. Oleh karena itu, dengan menggunakan pola Kasus Khusus ini, penentuan `BILANGAN KECIL` tersebut data dapat terselesaikan.

## LATIHAN SOAL 1

SOURCE CODE:

```
1  package main
2
3  import "fmt"
4
5  func main() {
6      var num, jumlah, count float64
7      for {
8          fmt.Scan(&num)
9          if num == 9999 {
10             break
11         }
12         jumlah += num
13         count++
14     }
15     if count > 0 {
16         fmt.Printf("Rerata: %.2f\n", jumlah/count)
17     } else {
18         fmt.Println("Tidak ada bilangan yang dimasukkan.")
19     }
20 }
```

OUTPUT:

```
PS F:\0 KULIAH\ALPROG\MODUL_16> go run "f:\0 KULIAH\ALPROG\MODUL_16\la
2727
3131
9999
Rerata: 2929.00
```

Deskripsi program latihan soal 1:

Tujuan program latihan soal 1 adalah Program Go untuk menghitung rata-rata dari angka-angka yang dimasukkan pengguna. Pengguna memasukkan angka secara berulang, dan ketika angka 9999 dimasukkan, program berhenti menerima input.

Contoh:

Input	Output
2727	Rerata : 2929.00
3131	
9999	

## LATIHAN SOAL 2

SOURCE CODE:

```
latsol2 > go latsol2.go > main
1  package main
2
3  import "fmt"
4
5  func main() {
6      var x string
7      var n int
8      fmt.Scan(&x)
9      fmt.Scan(&n)
10     strings := make([]string, n)
11     for i := 0; i < n; i++ {
12         fmt.Scan(&strings[i])
13     }
14     ditemukan := false
15     for _, s := range strings {
16         if s == x {
17             ditemukan = true
18             break
19         }
20     }
21     if ditemukan {
22         fmt.Println("String ditemukan.")
23     } else {
24         fmt.Println("String tidak ditemukan.")
25     }
26     posisi := -1
27     for i, s := range strings {
28         if s == x {
29             posisi = i + 1
30             break
31         }
32     }
33     if posisi != -1 {
34         fmt.Printf("String ditemukan pada posisi ke-%d.\n", posisi)
35     } else {
36         fmt.Println("String tidak ditemukan pada posisi manapun.")
37     }
38     count := 0
39     for _, s := range strings {
40         if s == x {
41             count++
42         }
43     }
44     fmt.Printf("Jumlah string x: %d\n", count)
45     if count >= 2 {
46         fmt.Println("Ada sedikitnya dua string x dalam kumpulan data.")
47     } else {
48         fmt.Println("Tidak ada sedikitnya dua string x dalam kumpulandata.")
49     }
50 }
```

## OUTPUT:

```
PS F:\0 KULIAH\ALPROG\MODUL_16> go run "f:\0 KULIAH\ALPROG\MODUL_16\latsol2\latsol2.go"
melon
5
semangka
durian
melon
mangga
jeruk
String ditemukan.
String ditemukan pada posisi ke-3.
Jumlah string x: 1
Tidak ada sedikitnya dua string x dalam kumpulandata.
PS F:\0 KULIAH\ALPROG\MODUL_16> █
```

Deskripsi program latihan soal 2:

Tujuan program latihan soal 2 adalah Program ini adalah program Go yang digunakan untuk memeriksa keberadaan, posisi, dan jumlah kemunculan sebuah string tertentu (x) dalam kumpulan string yang dimasukkan oleh pengguna.

Contoh:

Input	Output
Melon 5 Semangka Durian Melon Manga Jeruk	String ditemukan. String ditemukan pada posisi ke-3. Jumlah string x: 1 Tidak ada sedikitnya dua string x dalam kumpulan data.

## **DAFTAR PUSTAKA**

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

MODUL PRAKTIKUM 16 - SKEMA PEMROSESAN SEKUENSIAL  
ALGORITMA DAN PEMROGRAMAN 1 S1 INFORMATIKA