

LAPORAN PRAKTIKUM
ALGORITMA DAN PEMROGRAMAN 1
MODUL 16
“SKEMA PEMROSESAN SEKUENSIAL”



DISUSUN OLEH:
SHEILA STEPHANIE ANINDYA
103112400086
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

I. Skema Pemrosesan Sekuensial

Pemrosesan sekuensial adalah metode pemrosesan elemen secara satu per satu dari kumpulan informasi yang diakses berdasarkan urutan tertentu, seolah-olah membentuk deret elemen. Elemen yang diproses dapat berupa tipe dasar (integer, real, character, boolean) atau tipe komposisi (misalnya, objek dengan atribut tertentu).

Deret elemen ini dapat berasal dari berbagai sumber, seperti input device, tabel, matriks, file di media penyimpanan, atau struktur data seperti list. Kumpulan elemen tersebut diatur agar dapat dikenali dengan primitif untuk:

- Mengakses elemen pertama (First_Elmt).
- Mengakses elemen yang sedang diproses (Current_Elmt).
- Menentukan elemen berikutnya setelah elemen saat ini (Next_Elmt).
- Menandai akhir proses (EOP).

Pendekatan sekuensial ini dirancang sebagai pola umum untuk mendukung abstraksi yang lebih luas, melampaui sekadar instruksi pengulangan yang fokus pada bentuk perulangan.

II. Bentuk Struktur Sekuensial

Bentuk struktur sekuensial dapat diilustrasikan seperti bis yang sedang berjalan pada suatu jalan yang lurus dan tidak ada persimpangannya. Bis melaju pada jalan yang lurus sampai tiba di kota tujuan. Struktur sekuensial memiliki runtunan satu atau lebih instruksi. Setiap instruksi hanya dilakukan satu kali saja, kemudian dilanjutkan dengan instruksi berikutnya

Adapun karakteristik dari algoritma bentuk struktur sekuensial adalah sebagai berikut:

1. Setiap instruksi dikerjakan satu per satu .
2. Setiap instruksi dilaksanakan tepat sekali; tidak ada instruksi yang diulang .
3. Urutan instruksi yang dilaksanakan sama dengan urutan instruksi sebagaimana yang tertulis di dalam teks algoritma .
4. Instruksi terakhir merupakan akhir algoritma.

SOAL LATIHAN

Statement perulangan

1. Diberikan sejumlah bilangan riil yang diakhiri dengan marker 9999, cari rerata dari bilangan-bilangan tersebut.

Source Code:

```
package main

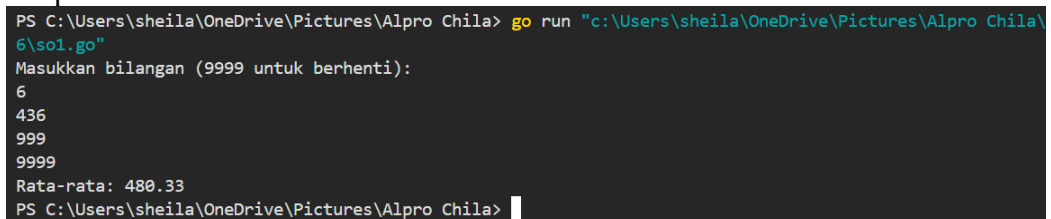
import "fmt"

func main() {
    var num float64
    sum := 0.0
    count := 0

    fmt.Println("Masukkan bilangan (9999 untuk berhenti):")
    for {
        fmt.Scan(&num)
        if num == 9999 {
            break
        }
        sum += num
        count++
    }

    if count > 0 {
        average := sum / float64(count)
        fmt.Printf("Rata-rata: %.2f\n", average)
    } else {
        fmt.Println("Tidak ada bilangan yang dimasukkan")
    }
}
```

Output:



```
PS C:\Users\sheila\OneDrive\Pictures\Alpro Chila> go run "c:\Users\sheila\OneDrive\Pictures\Alpro Chila\6\so1.go"
Masukkan bilangan (9999 untuk berhenti):
6
436
999
9999
Rata-rata: 480.33
PS C:\Users\sheila\OneDrive\Pictures\Alpro Chila>
```

Deskripsi Program:

Program ini menghitung rata-rata dari bilangan-bilangan yang diinput. Program berhenti menerima input ketika pengguna memasukkan angka khusus 9999.

Statement perulangan

2. Diberikan string x dan n buah string. x adalah data pertama yang dibaca, n adalah data bilangan yang dibaca kedua, dan n data berikutnya adalah data string.

Source Code:

```
package main

import "fmt"

func main() {
    var x string
    var n int

    fmt.Print("Masukkan string yang dicari: ")
    fmt.Scan(&x)
    fmt.Print("Masukkan jumlah string: ")
    fmt.Scan(&n)

    strings := make([]string, n)
    count := 0
    firstPos := -1

    fmt.Println("Masukkan", n, "string:")
    for i := 0; i < n; i++ {
        fmt.Scan(&strings[i])
        if strings[i] == x {
            if firstPos == -1 {
                firstPos = i
            }
            count++
        }
    }

    fmt.Println("String ditemukan:", count > 0)
    fmt.Println("Posisi pertama:", firstPos+1)
    fmt.Println("Jumlah kemunculan:", count)
    fmt.Println("Ada minimal dua kemunculan:", count >= 2)
}
```

Output:

```
PS C:\Users\sheila\OneDrive\Pictures\Alpro Chila> go run "c:\Users\sheila\OneDrive\Pictures\Alpro Chila\
Masukkan string yang dicari: x
Masukkan jumlah string: 9
Masukkan 9 string:
a
d
h
r
x
j
e
x
j
String ditemukan: true
Posisi pertama: 5
Jumlah kemunculan: 2
Ada minimal dua kemunculan: true
PS C:\Users\sheila\OneDrive\Pictures\Alpro Chila>
```

Deskripsi Program:

Program ini mencari string tertentu di dalam daftar string yang diinput, serta menentukan jumlah kemunculan string tersebut, posisi pertama kemunculannya, dan berapa kali string tersebut muncul.

- a. Apakah string x ada dalam kumpulan n data string tersebut?
➤ Ada
- b. Pada posisi ke berapa string x tersebut ditemukan?
➤ 5
- c. Ada berapakah string x dalam kumpulan n data string tersebut?
➤ 2
- d. Adakah sedikitnya dua string x dalam n data string tersebut?
➤ Ada

Statement perulangan

3. Empat daerah A, B, C, dan D yang berdekatan ingin mengukur curah hujan. Misal curah hujan dihitung berdasarkan banyaknya tetesan air hujan. Setiap tetesan berukuran 0.0001 ml curah hujan. Tetesan air hujan turun secara acak dari titik (0,0) sampai (1,1). Jika diterima input yang menyatakan banyaknya tetesan air hujan. Tentukan curah hujan untuk keempat daerah tersebut.
Buatlah program yang menerima input berupa banyaknya tetesan air hujan. Kemudian buat koordinat/titik (x, y) secara acak dengan menggunakan fungsi `rand.Float64()`.
Hitung dan tampilkan banyaknya tetesan yang jatuh pada daerah A, B, C dan D. Konversikan satu tetesan berukuran 0.0001 milimeter.

Source Code:

```
package main

import (
    "fmt"
    "math/rand"
)

func main() {
    var drops int
    fmt.Print("jumlah tetesan air: ")
    fmt.Scan(&drops)

    countA, countB, countC, countD := 0, 0, 0, 0

    for i := 0; i < drops; i++ {
        x := rand.Float64()
        y := rand.Float64()

        // region tetesan jatuh
        if x < 0.5 {
            if y < 0.5 {
                countA++
            } else {
                countD++
            }
        } else {
            if y < 0.5 {
                countB++
            } else {
                countC++
            }
        }
    }

    fmt.Printf("Curah hujan daerah A: %.4f mm\n", float64(countA)*0.0001)
    fmt.Printf("Curah hujan daerah B: %.4f mm\n", float64(countB)*0.0001)
    fmt.Printf("Curah hujan daerah C: %.4f mm\n", float64(countC)*0.0001)
    fmt.Printf("Curah hujan daerah D: %.4f mm\n", float64(countD)*0.0001)
}
```

Output:

```
PS C:\Users\sheila\OneDrive\Pictures\Alpro Chila> go run "c:\Users\sheila\OneDrive\Pictures\Alpro Chila\  
jumlah tetesan air: 10000000  
Curah hujan daerah A: 249.9820 mm  
Curah hujan daerah B: 250.0895 mm  
Curah hujan daerah C: 250.1441 mm  
Curah hujan daerah D: 249.7844 mm  
PS C:\Users\sheila\OneDrive\Pictures\Alpro Chila> |
```

Deskripsi Program:

Program ini menghitung distribusi curah hujan di empat wilayah (A, B, C, D) berdasarkan koordinat acak tetesan air hujan. Setelah itu program menampilkan curah hujan dalam satuan milim

DAFTAR PUSTAKA

1. MGMPTIK Semarang. (2011). *Skema Pemrosesan Sekuensial*.
https://mgmptiksemarang.wordpress.com/wp-content/uploads/2011/08/10-skemaseq_nopw.pdf
2. Kementerian Pendidikan dan Kebudayaan. *Modul PB 4: Struktur Algoritma Pemrograman Sekuensial*.
https://lmsspada.kemdikbud.go.id/pluginfile.php/679118/mod_resource/content/4/Modul%20PB%204%20-%20Struktur%20algoritma%20pemrograman%20Sekuensial.pdf

eter (mm) untuk setiap wilayah.