

**LAPORAN PRAKTIKUM
ALGORITMA DAN PEMROGRAMAN 2**

**MODUL VII
STRUCT & ARRAY**



Disusun Oleh :

M. Haidar Akhbiyani / 2311102276

S1-IF-11-06

Dosen Pengampu :

Abednego Dwi Septiadi

PROGRAM STUDI S1 TEKNIK INFORMATIKA

FAKULTAS INFORMATIKA

TELKOM UNIVERSITY PURWOKERTO

2024

I. DASAR TEORI

Dalam pemrograman, penggunaan struct dan array sangat penting untuk pengorganisasian dan pengelolaan data. Struct merupakan tipe data komposit yang memungkinkan pengelompokan berbagai variabel dengan tipe data berbeda ke dalam satu kesatuan yang lebih kompleks. Misalnya, sebuah struct dapat merepresentasikan objek geometris seperti titik (dengan koordinat x dan y) atau lingkaran (dengan pusat berupa titik dan radius). Dengan struct, data yang saling terkait dapat dimanipulasi secara logis dan terstruktur, memungkinkan pengembangan program yang modular dan dapat diperluas.

Sementara itu, array digunakan untuk menyimpan sejumlah elemen dengan tipe data yang sama dalam sebuah struktur data linier. Setiap elemen dalam array dapat diakses melalui indeks tertentu. Dalam konteks modul ini, array digunakan untuk menyimpan dan memproses elemen seperti hasil pertandingan, deret bilangan, atau karakter dalam string. Kombinasi antara struct dan array memberikan fleksibilitas tinggi untuk mengelola data, terutama dalam implementasi algoritma yang melibatkan perhitungan matematis, validasi data, atau analisis pola, seperti penghitungan jarak Euclidean antara titik-titik, pemeriksaan palindrom, dan pengelolaan hasil pertandingan. Dengan penerapan struct dan array secara bersamaan, program dapat dikelola secara efisien, mencakup validasi, modularitas, serta penggunaan kembali komponen program.

II. GUIDED

1. Guided 1

Sourcecode

```
package main

import (
    "fmt"
    "math"
)

type Titik struct {
    x int
    y int
}

type Lingkaran struct {
    pusat Titik
    radius int
}

func hitungJarak(a, b Titik) float64 {
    return math.Sqrt(float64((a.x-b.x)*(a.x-b.x) + (a.y-b.y)*(a.y-b.y)))
}

func titikDiDalamLingkaran(t Titik, l Lingkaran) bool {
    jarak := hitungJarak(t, l.pusat)
    return jarak <= float64(l.radius)
}

func main() {
    var cx1, cy1, r1 int
    fmt.Print("Masukkan koordinat pusat dan radius lingkaran 1 (cx1 cy1 r1): ")
    fmt.Scanln(&cx1, &cy1, &r1)
    lingkaran1 := Lingkaran{pusat: Titik{x: cx1, y: cy1}, radius: r1}

    var cx2, cy2, r2 int
    fmt.Print("Masukkan koordinat pusat dan radius lingkaran 2 (cx2 cy2 r2): ")
    fmt.Scanln(&cx2, &cy2, &r2)
    lingkaran2 := Lingkaran{pusat: Titik{x: cx2, y: cy2}, radius: r2}

    var x, y int
    fmt.Print("Masukkan koordinat titik sembarang (x y): ")
```

```

fmt.Scanln(&x, &y)
titik := Titik{x: x, y: y}

diDalamL1 := titikDiDalamLingkaran(titik, lingkaran1)
diDalamL2 := titikDiDalamLingkaran(titik, lingkaran2)

if diDalamL1 && diDalamL2 {
    fmt.Println("Titik di dalam lingkaran 1 dan 2")
} else if diDalamL1 {
    fmt.Println("Titik di dalam lingkaran 1")
} else if diDalamL2 {
    fmt.Println("Titik di dalam lingkaran 2")
} else {
    fmt.Println("Titik berada di luar lingkaran")
}
}

```

Screenshoot Output

```

PS C:\Users\Lenovo> go run "e:\2311102276_M.Haidar Akhbiyani Modul VII\GUIDED 1 MODUL VII.go"
Masukkan koordinat pusat dan radius lingkaran 1 (cx1 cy1 r1): 125
Masukkan koordinat pusat dan radius lingkaran 2 (cx2 cy2 r2): 152
Masukkan koordinat titik sembarang (x y): 24
Titik berada di luar lingkaran
PS C:\Users\Lenovo>

```

Deskripsi Program

Program diatas mempunyai kegunaan untuk menentukan apakah titik berada di dalam atau diluar lingkaran. Pertama program menerima input koordinasi pusat dan radius dua lingkaran. Fungsi hitungJarak digunakan untuk menghitung jarak Euclidean antara dua titik, dan fungsi titikDiDalamLingkaran memeriksa apakah jarak titik ke pusat lingkaran lebih kecil atau sama dengan radius lingkaran

I. UNGUIDED

1. Unguided 1

Sourcecode

```
package main

import (
    "fmt"
    "math"
)

func main() {
    var jumlahElemen int
    fmt.Print("Masukkan jumlah elemen array (N): ")
    fmt.Scan(&jumlahElemen)

    // Membuat dan mengisi array
    bilangan := make([]int, jumlahElemen)
    for i := 0; i < jumlahElemen; i++ {
        fmt.Printf("Masukkan bilangan ke-%d: ", i)
        fmt.Scan(&bilangan[i])
    }

    // Menu utama program
    for {
        fmt.Println("\nPilihan Menu:")
        fmt.Println("1. Tampilkan semua isi array")
        fmt.Println("2. Tampilkan elemen dengan indeks ganjil")
        fmt.Println("3. Tampilkan elemen dengan indeks genap")
        fmt.Println("4. Tampilkan elemen dengan indeks kelipatan tertentu")
        fmt.Println("5. Hapus elemen pada indeks tertentu")
        fmt.Println("6. Hitung rata-rata")
        fmt.Println("7. Hitung standar deviasi")
        fmt.Println("8. Hitung frekuensi kemunculan suatu bilangan")
        fmt.Println("9. Keluar dari program")

        var pilihan int
        fmt.Print("Masukkan pilihan menu (1-9): ")
        fmt.Scan(&pilihan)

        switch pilihan {
```

```

        case 1:
            tampilkanArray(bilangan)
        case 2:
            tampilkanIndeksGanjil(bilangan)
        case 3:
            tampilkanIndeksGenap(bilangan)
        case 4:
            var kelipatan int
            fmt.Print("Masukkan nilai kelipatan: ")
            fmt.Scan(&kelipatan)
            tampilkanIndeksKelipatan(bilangan, kelipatan)
        case 5:
            var indeks int
            fmt.Print("Masukkan indeks yang akan dihapus: ")
            fmt.Scan(&indeks)
            bilangan = hapusElemen(bilangan, indeks)
        case 6:
            rerata := hitungRataRata(bilangan)
            fmt.Printf("Rata-rata: %.2f\n", rerata)
        case 7:
            simpanganBaku := hitungStandarDeviasi(bilangan)
            fmt.Printf("Standar deviasi: %.2f\n",
simpanganBaku)
        case 8:
            var nilai int
            fmt.Print("Masukkan bilangan yang ingin dihitung
frekuensinya: ")
            fmt.Scan(&nilai)
            frekuensi := hitungFrekuensi(bilangan, nilai)
            fmt.Printf("Frekuensi kemunculan bilangan %d: %d
kali\n", nilai, frekuensi)
        case 9:
            fmt.Println("Terima kasih, program selesai.")
            return
        default:
            fmt.Println("Pilihan tidak valid! Silakan pilih
menu 1-9.")
    }
}

func tampilkanArray(arr []int) {
    fmt.Print("Isi array: ")
    for i, nilai := range arr {
        fmt.Printf("[%d]=%d ", i, nilai)
    }
}

```

```

    }
    fmt.Println()
}

func tampilkanIndeksGanjil(arr []int) {
    fmt.Print("Elemen dengan indeks ganjil: ")
    for i := 1; i < len(arr); i += 2 {
        fmt.Printf("[%d]=%d ", i, arr[i])
    }
    fmt.Println()
}

func tampilkanIndeksGenap(arr []int) {
    fmt.Print("Elemen dengan indeks genap: ")
    for i := 0; i < len(arr); i += 2 {
        fmt.Printf("[%d]=%d ", i, arr[i])
    }
    fmt.Println()
}

func tampilkanIndeksKelipatan(arr []int, kelipatan int) {
    fmt.Printf("Elemen dengan indeks kelipatan %d: ",
kelipatan)
    for i := 0; i < len(arr); i++ {
        if i%kelipatan == 0 {
            fmt.Printf("[%d]=%d ", i, arr[i])
        }
    }
    fmt.Println()
}

func hapusElemen(arr []int, indeks int) []int {
    if indeks < 0 || indeks >= len(arr) {
        fmt.Println("Indeks tidak valid!")
        return arr
    }

    hasil := append(arr[:indeks], arr[indeks+1:]...)
    fmt.Println("Elemen berhasil dihapus!")
    return hasil
}

func hitungRataRata(arr []int) float64 {
    if len(arr) == 0 {
        return 0
    }

```

```

    }
    jumlah := 0
    for _, nilai := range arr {
        jumlah += nilai
    }
    return float64(jumlah) / float64(len(arr))
}

func hitungStandarDeviasi(arr []int) float64 {
    if len(arr) == 0 {
        return 0
    }
    rerata := hitungRataRata(arr)

    var jumlahKuadratSelisih float64
    for _, nilai := range arr {
        selisih := float64(nilai) - rerata
        jumlahKuadratSelisih += selisih * selisih
    }

    variansi := jumlahKuadratSelisih / float64(len(arr))
    return math.Sqrt(variansi)
}

func hitungFrekuensi(arr []int, nilai int) int {
    frekuensi := 0
    for _, num := range arr {
        if num == nilai {
            frekuensi++
        }
    }
    return frekuensi
}

```


Screenshoot Output

```
PS C:\Users\Lenovo> go run "e:\2311102276_M.Haidar Akhbiyani Modul VII\UNGUIDED 1 MODUL VII.go"
Masukkan jumlah elemen array (N): 5
Masukkan bilangan ke-0: 10
Masukkan bilangan ke-1: 11
Masukkan bilangan ke-2: 12
Masukkan bilangan ke-3: 13
Masukkan bilangan ke-4: 14
```

```
Pilihan Menu:
1. Tampilkan semua isi array
2. Tampilkan elemen dengan indeks ganjil
3. Tampilkan elemen dengan indeks genap
4. Tampilkan elemen dengan indeks kelipatan tertentu
5. Hapus elemen pada indeks tertentu
6. Hitung rata-rata
7. Hitung standar deviasi
8. Hitung frekuensi kemunculan suatu bilangan
9. Keluar dari program
Masukkan pilihan menu (1-9): 2
Elemen dengan indeks ganjil: [1]=11 [3]=13
```

```
Pilihan Menu:
1. Tampilkan semua isi array
2. Tampilkan elemen dengan indeks ganjil
3. Tampilkan elemen dengan indeks genap
4. Tampilkan elemen dengan indeks kelipatan tertentu
5. Hapus elemen pada indeks tertentu
6. Hitung rata-rata
7. Hitung standar deviasi
8. Hitung frekuensi kemunculan suatu bilangan
9. Keluar dari program
Masukkan pilihan menu (1-9): 1
Isi array: [0]=10 [1]=11 [2]=12 [3]=13 [4]=14
```

```
Pilihan Menu:
1. Tampilkan semua isi array
2. Tampilkan elemen dengan indeks ganjil
3. Tampilkan elemen dengan indeks genap
4. Tampilkan elemen dengan indeks kelipatan tertentu
5. Hapus elemen pada indeks tertentu
6. Hitung rata-rata
7. Hitung standar deviasi
8. Hitung frekuensi kemunculan suatu bilangan
9. Keluar dari program
Masukkan pilihan menu (1-9): 3
Elemen dengan indeks genap: [0]=10 [2]=12 [4]=14
```

```
Pilihan Menu:
1. Tampilkan semua isi array
2. Tampilkan elemen dengan indeks ganjil
3. Tampilkan elemen dengan indeks genap
4. Tampilkan elemen dengan indeks kelipatan tertentu
5. Hapus elemen pada indeks tertentu
6. Hitung rata-rata
7. Hitung standar deviasi
8. Hitung frekuensi kemunculan suatu bilangan
9. Keluar dari program
Masukkan pilihan menu (1-9): 4
Masukkan nilai kelipatan: 5
Elemen dengan indeks kelipatan 5: [0]=10
```

```
Pilihan Menu:
1. Tampilkan semua isi array
2. Tampilkan elemen dengan indeks ganjil
3. Tampilkan elemen dengan indeks genap
4. Tampilkan elemen dengan indeks kelipatan tertentu
5. Hapus elemen pada indeks tertentu
6. Hitung rata-rata
7. Hitung standar deviasi
8. Hitung frekuensi kemunculan suatu bilangan
9. Keluar dari program
Masukkan pilihan menu (1-9): 6
Rata-rata: 12.00
```

```
Pilihan Menu:
1. Tampilkan semua isi array
2. Tampilkan elemen dengan indeks ganjil
3. Tampilkan elemen dengan indeks genap
4. Tampilkan elemen dengan indeks kelipatan tertentu
5. Hapus elemen pada indeks tertentu
6. Hitung rata-rata
7. Hitung standar deviasi
8. Hitung frekuensi kemunculan suatu bilangan
9. Keluar dari program
Masukkan pilihan menu (1-9): 7
Standar deviasi: 1.41
```

```
Pilihan Menu:
1. Tampilkan semua isi array
2. Tampilkan elemen dengan indeks ganjil
3. Tampilkan elemen dengan indeks genap
4. Tampilkan elemen dengan indeks kelipatan tertentu
5. Hapus elemen pada indeks tertentu
6. Hitung rata-rata
7. Hitung standar deviasi
8. Hitung frekuensi kemunculan suatu bilangan
9. Keluar dari program
Masukkan pilihan menu (1-9): 8
Masukkan bilangan yang ingin dihitung frekuensinya: 9
Frekuensi kemunculan bilangan 9: 0 kali
```

```
Pilihan Menu:
1. Tampilkan semua isi array
2. Tampilkan elemen dengan indeks ganjil
3. Tampilkan elemen dengan indeks genap
4. Tampilkan elemen dengan indeks kelipatan tertentu
5. Hapus elemen pada indeks tertentu
6. Hitung rata-rata
7. Hitung standar deviasi
8. Hitung frekuensi kemunculan suatu bilangan
9. Keluar dari program
Masukkan pilihan menu (1-9): 9
Terima kasih, program selesai.
PS C:\Users\Lenovo> █
```

Deskripsi Program

Program ini mengelola array bilangan bulat dengan berbagai fitur seperti menampilkan isi array, elemen berdasarkan indeks tertentu (ganjil, genap, atau kelipatan), menghitung rata-rata, standar deviasi, dan frekuensi kemunculan bilangan. Pengguna juga dapat menghapus elemen array berdasarkan indeks. Program berjalan interaktif melalui menu hingga pengguna memilih untuk keluar. Fungsionalitasnya diimplementasikan secara modular untuk kemudahan penggunaan dan pemeliharaan.

2. Unguided 2

Sourcecode

```
package main
import "fmt"

func main() {
    var klubA, klubB string
    var skorA, skorB int
    var hasil []string

    fmt.Print("Masukkan Klub A : ")
    fmt.Scan(&klubA)

    fmt.Print("Masukkan Klub B : ")
    fmt.Scan(&klubB)

    i := 1
    for {
        fmt.Printf("Pertandingan %v :", i)
        fmt.Scan(&skorA, &skorB)

        if skorA < 0 || skorB < 0 {
            break
        } else if skorA > skorB {
            hasil = append(hasil, klubA)
        } else if skorB > skorA {
            hasil = append(hasil, klubB)
        } else {
            hasil = append(hasil, "Draw")
        }
        i++
    }

    for i := 0; i < len(hasil); i++ {
        fmt.Println("Hasil ", i+1, " : ", hasil[i])
    }
}
```

Screenshoot Output

```
PS C:\Users\Lenovo> go run "e:\2311102276_M.Haidar Akhbiyani Modul VII\UNGUIDED 2 MODUL VII.go"
Masukkan Klub A : Vorzhat
Masukkan Klub B : Garez
Pertandingan 1 :3 2
Pertandingan 2 :2 2
Pertandingan 3 :4 3
Pertandingan 4 :2 4
Pertandingan 5 :0 0
Pertandingan 6 :1 2
Pertandingan 7 :2 1
Pertandingan 8 :3 6
Pertandingan 9 :4 2
Pertandingan 10 :1 5
Pertandingan 11 :-2 3
Hasil 1 : Vorzhat
Hasil 2 : Draw
Hasil 3 : Vorzhat
Hasil 4 : Garez
Hasil 5 : Draw
Hasil 6 : Garez
Hasil 7 : Vorzhat
Hasil 8 : Garez
Hasil 9 : Vorzhat
Hasil 10 : Garez
PS C:\Users\Lenovo> █
```

Deskripsi Program

Program ini mencatat hasil pertandingan antara dua klub sepak bola, Klub A dan Klub B. Pengguna memasukkan nama kedua klub terlebih dahulu, lalu memasukkan skor untuk setiap pertandingan dalam format berulang. Jika skor Klub A lebih besar, hasil pertandingan dicatat sebagai kemenangan Klub A; jika sebaliknya, sebagai kemenangan Klub B. Jika skor sama, pertandingan dianggap seri (Draw). Pengisian skor berlanjut hingga pengguna memasukkan skor negatif pada salah satu klub, yang menandai akhir input.

3. Unguided 3

Sourcecode

```
package main

import (
    "fmt"
    "strings"
)

func balikArray(arr []rune) []rune {
    n := len(arr)
    for i := 0; i < n/2; i++ {
        arr[i], arr[n-1-i] = arr[n-1-i], arr[i]
    }
    return arr
}

func cekPalindrom(arr []rune) bool {
    n := len(arr)
    for i := 0; i < n/2; i++ {
        if arr[i] != arr[n-1-i] {
            return false
        }
    }
    return true
}

func main() {
    var input1, input2 string

    fmt.Print("Teks: ")
    fmt.Scanln(&input1)

    txtLower1 := strings.ToLower(input1)
    array1 := []rune(txtLower1)
    balik1 := balikArray(array1)

    fmt.Printf("Reverse teks: %s\n",
strings.ToLower(string(balik1)))
    fmt.Println()

    fmt.Print("Teks: ")
    fmt.Scanln(&input2)
```

```

    txtLower2 := strings.ToLower(input2)
    array2 := []rune(txtLower2)
    balik2 := balikArray(array2)

    fmt.Printf("Reverse teks: %s\n",
strings.ToLower(string(balik2)))
    fmt.Println()

    isPalindrom1 := cekPalindrom(array1)
    fmt.Printf("Teks :%s\n", input1)
    fmt.Printf("polindrom ?%t\n", isPalindrom1)
    fmt.Println()

    isPalindrom2 := cekPalindrom(array2)
    fmt.Printf("Teks :%s\n", input2)
    fmt.Printf("polindrom ?%t\n", isPalindrom2)
}

```

Screenshoot Output

```

PS C:\Users\Lenovo> go run "e:\2311102276_M.Haidar Akhbiyani Modul VII\UNGUIDED 3 MODUL VII.go"
Teks: daun
Reverse teks: nuad

Teks: southcrew
Reverse teks: werchtuos

Teks :daun
polindrom ?false

Teks :southcrew
polindrom ?false
PS C:\Users\Lenovo>

```

Deskripsi Program

Program ini memproses dua teks input dari pengguna untuk membalikkan urutan hurufnya dan memeriksa apakah teks tersebut merupakan palindrom. Sebuah teks dikatakan palindrom jika tetap sama ketika dibaca dari depan maupun belakang, tanpa memperhatikan huruf besar atau kecil. Setelah pengguna memasukkan teks, program mengubahnya menjadi huruf kecil untuk menghindari masalah sensitivitas kapitalisasi. Fungsi `balikArray` digunakan untuk membalikkan urutan karakter teks, dan hasilnya ditampilkan sebagai teks terbalik. Selanjutnya, fungsi `cekPalindrom` memeriksa apakah teks asli sama dengan versi terbaliknya untuk menentukan apakah teks tersebut palindrom. Program kemudian menampilkan hasil akhir berupa teks terbalik dan status apakah teks tersebut palindrom atau tidak untuk kedua input yang diberikan.

