

LAPORAN PRAKTIKUM ALGORITMA DAN PEMROGRAMAN 2

MODUL VII

STRUCK & ARRAY



Disusun Oleh :

Fahri Ramadhan / 2311102024

IF-11-06

Dosen Pengampu :

Abednego Dwi Septiadi, S.Kom., M.Kom

PROGRAM STUDI S1 TEKNIK INFORMATIKA

FAKULTAS INFORMATIKA

TELKOM UNIVERSITY PURWOKERTO

2024

I. DASAR TEORI

Struct adalah tipe data yang memungkinkan pengelompokan berbagai variabel dengan tipe data berbeda ke dalam satu kesatuan. Struct sering digunakan untuk merepresentasikan entitas yang kompleks seperti data mahasiswa, kendaraan, atau produk. Dalam struct, setiap anggota (member) memiliki nama unik yang memudahkan akses data. Misalnya, struct dapat memiliki anggota berupa nama (string), umur (integer), dan nilai (float). Dengan struct, programmer dapat membuat tipe data baru yang lebih terstruktur dan sesuai dengan kebutuhan program. Struct sering dipakai dalam bahasa pemrograman seperti C, C++, dan lainnya.

Array adalah struktur data yang digunakan untuk menyimpan sekumpulan elemen bertipe sama dalam satu variabel. Elemen-elemen array diakses menggunakan indeks, yang dimulai dari nol. Array sangat cocok digunakan ketika data memiliki jumlah yang tetap atau dapat dikelompokkan secara linear, seperti daftar nilai atau kumpulan angka. Kombinasi struct dan array memungkinkan pembuatan struktur data yang lebih kompleks, misalnya array dari struct untuk menyimpan kumpulan data mahasiswa atau struct dengan anggota berupa array untuk menyimpan beberapa nilai terkait dalam satu entitas.

II. GUIDED

1. Guided 1

Sourcecode

```
package main
import (
    "fmt"
    "math"
)
type Titik
struct {    x, y
int
}
type Lingkaran struct
{
    pusat    Titik
radius int
}
func jarak(p, q Titik) float64 {
    return math.Sqrt(float64((p.x-q.x)*(p.x-q.x) + (p.y-q.y)*(p.y-q.y)))
}
func diDalam(c Lingkaran, p Titik) bool {
    return jarak(c.pusat, p) <= float64(c.radius)
}
func main() {
    var l1, l2 Lingkaran
    var t Titik
    fmt.Print("Masukkan koordinat pusat dan radius lingkaran 1 (cx1 cy1 r1): ")
    fmt.Scan(&l1.pusat.x, &l1.pusat.y, &l1.radius)

    fmt.Print("Masukkan koordinat pusat dan radius lingkaran 2 (cx2 cy2 r2): ")
    fmt.Scan(&l2.pusat.x, &l2.pusat.y, &l2.radius)

    fmt.Print("Masukkan koordinat titik sembarang (x y): ")
    fmt.Scan(&t.x, &t.y)
    diDalamL1 := diDalam(l1, t)
    diDalamL2 := diDalam(l2, t)

    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 di luar lingkaran 1 dan 2")
    }
}
```

```
}  
}
```

Screenshoot Output

```
PS C:\Users\fahri\OneDrive\Documents\Modul6_al2> go run "c:\Users\fahri\OneDrive\Documents\Modul6_al2\main.go"  
Masukkan koordinat pusat dan radius lingkaran 1 (cx1 cy1 r1): 7 7 7  
Masukkan koordinat pusat dan radius lingkaran 2 (cx2 cy2 r2): 1 1 1  
Masukkan koordinat titik sembarang (x y): 2 2  
Titik di luar lingkaran 1 dan 2  
PS C:\Users\fahri\OneDrive\Documents\Modul6_al2> █
```

Deskripsi Program

Fungsi jarak Fungsi ini menghitung jarak antara dua titik pada bidang kartesius menggunakan rumus jarak Euclidean, Fungsi ini menerima dua parameter bertipe Titik (p dan q) dan mengembalikan nilai bertipe float64 sebagai hasil perhitungan jarak.

.

III. UNGUIDED

Unguided 1 Sourcecode

```
//Fahri Ramadhan 2311102024

package main

import (
    "fmt"
    "math"
)

func main() {
    var n int
    fmt.Print("Masukkan jumlah elemen array: ")
    fmt.Scanln(&n)

    array := make([]int, n)

    // Mengisi array
    fmt.Println("Masukkan elemen array:")
    for i := 0; i < n; i++ {
        fmt.Printf("Elemen ke-%d: ", i)
        fmt.Scanln(&array[i])
    }

    // a. Menampilkan keseluruhan isi array
    fmt.Println("\nKeseluruhan isi array:")
    fmt.Println(array)

    // b. Menampilkan elemen array dengan indeks ganjil
```

```

    fmt.Println("\nElemen dengan indeks ganjil:")
    for i := 1; i < len(array); i += 2 {
        fmt.Printf("Indeks %d: %d\n", i, array[i])
    }

    // c. Menampilkan elemen array dengan indeks genap
    fmt.Println("\nElemen dengan indeks genap:")
    for i := 0; i < len(array); i += 2 {
        fmt.Printf("Indeks %d: %d\n", i, array[i])
    }

    // d. Menampilkan elemen dengan indeks kelipatan
    bilangan x
    var x int
    fmt.Print("\nMasukkan bilangan x untuk kelipatan
    indeks: ")
    fmt.Scanln(&x)
    fmt.Println("Elemen dengan indeks kelipatan", x,
    ":")
    for i := 0; i < len(array); i++ {
        if i%x == 0 {
            fmt.Printf("Indeks    %d:    %d\n",    i,
            array[i])
        }
    }

    // e. Menghapus elemen array pada indeks tertentu
    var index int
    fmt.Print("\nMasukkan indeks elemen yang ingin
    dihapus: ")
    fmt.Scanln(&index)

```

```

        if index >= 0 && index < len(array) {
            array = append(array[:index],
array[index+1:]...)

            fmt.Println("Array setelah elemen dihapus:")
            fmt.Println(array)
        } else {
            fmt.Println("Indeks tidak valid.")
        }

// f. Menampilkan rata-rata bilangan dalam array
sum := 0
for _, value := range array {
    sum += value
}

rataRata := float64(sum) / float64(len(array))

fmt.Printf("\nRata-rata nilai array: %.2f\n",
rataRata)

// g. Menampilkan standar deviasi (simpangan baku)
var deviasiSum float64
for _, value := range array {
    deviasiSum += math.Pow(float64(value) -
rataRata, 2)
}

standarDeviasi := math.Sqrt(deviasiSum /
float64(len(array)))

fmt.Printf("Simpangan baku array: %.2f\n",
standarDeviasi)

// h. Menampilkan frekuensi suatu bilangan
var target int

```

```

    fmt.Print("\nMasukkan bilangan untuk menghitung
frekuensinya: ")

    fmt.Scanln(&target)

    frekuensi := 0

    for _, value := range array {
        if value == target {
            frekuensi++
        }
    }

    fmt.Printf("Frekuensi bilangan %d: %d kali\n",
target, frekuensi)
}

```

Screenshoot Output

```

Masukkan jumlah elemen array: 7
Masukkan elemen array:
Elemen ke-0: 7
Elemen ke-1: 7
Elemen ke-2: 7
Elemen ke-3: 7
Elemen ke-4: 7
Elemen ke-5: 7
Elemen ke-6: 7

Keseluruhan isi array:
[7 7 7 7 7 7 7]

Elemen dengan indeks ganjil:
Indeks 1: 7
Indeks 3: 7
Indeks 5: 7

Elemen dengan indeks genap:
Indeks 0: 7
Indeks 2: 7
Indeks 4: 7
Indeks 6: 7

Masukkan bilangan x untuk kelipatan indeks: 7
Elemen dengan indeks kelipatan 7 :
Indeks 0: 7

Masukkan indeks elemen yang ingin dihapus: 2
Array setelah elemen dihapus:
[7 7 7 7 7 7]

Rata-rata nilai array: 7.00
Simpangan baku array: 0.00

```

Deskripsi Program

Program meminta input data array dan menampilkan semua elemen serta elemen dengan indeks tertentu. Setelah input nilai x, program menampilkan

elemen dengan indeks kelipatan x . Program kemudian menghapus elemen pada indeks yang dipilih dan menampilkan array hasilnya. Rata-rata dan standar deviasi dihitung berdasarkan elemen array setelah penghapusan. Frekuensi kemunculan nilai tertentu dalam array dihitung dan ditampilkan.

Unguided 2

Sourcecode

```
//Fahri Ramadhan
2311102024
package main
import (
    "fmt"
)

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

    // Meminta nama klub yang bertanding
    fmt.Print("Masukkan nama Klub A: ")
    fmt.Scanln(&klubA)
    fmt.Print("Masukkan nama Klub B: ")
    fmt.Scanln(&klubB)
    for i := 1; ; i++ {        fmt.Printf("Pertandingan
%d - Masukkan skor %s:
", i, klubA)        fmt.Scan(&skorA)
    fmt.Printf("Pertandingan %d - Masukkan skor %s:
", i, klubB)
    fmt.Scan(&skorB)

    // Jika skor negatif, keluar dari loop    if
skorA < 0 || skorB < 0 {        fmt.Println("Skor
tidak valid. Pertandingan selesai.")        break
    }

    // Tentukan pemenang dan simpan ke dalam array
    jika ada pemenang    if skorA > skorB {
    pemenang = append(pemenang, klubA)
    fmt.Printf("Hasil %d: %s\n", i, klubA)    } else if
skorB > skorA {        pemenang = append(pemenang,
klubB)        fmt.Printf("Hasil %d: %s\n", i, klubB)
    } else {
    fmt.Printf("Hasil %d: Draw\n", i)    }
    }

    // Tampilkan daftar klub yang memenangkan
pertandingan    fmt.Println("\nDaftar klub yang
memenangkan pertandingan:")    for _, klub := range
pemenang {
```

```
        fmt.Println(klub)
    }
}
```

Screenshoot Output

```
Masukkan nama Klub A: madrid
Masukkan nama Klub B: barca
Pertandingan 1 - Masukkan skor madrid: 99
Pertandingan 1 - Masukkan skor barca: 1
Hasil 1: madrid menang
Pertandingan 2 - Masukkan skor madrid: 99
Pertandingan 2 - Masukkan skor barca: 2
Hasil 2: madrid menang
Pertandingan 3 - Masukkan skor madrid: 100
Pertandingan 3 - Masukkan skor barca: -1
Skor tidak valid. Pertandingan selesai.

Daftar klub yang memenangkan pertandingan:
madrid
madrid
PS C:\Users\fahri\OneDrive\Documents\Modul6_al2>
```

Deskripsi Program

Program menggunakan perulangan tanpa batas (for i := 1; ;) untuk meminta skor dari setiap pertandingan. Setiap iterasi merepresentasikan satu pertandingan: Pengguna memasukkan skor untuk klubA dan klubB (disimpan dalam skorA dan skorB). Jika salah satu skor negatif, program menghentikan loop dengan pesan "Skor tidak valid. Pertandingan selesai."

Unguided 3

Sourcecode

```
//Fahri Ramadhan 2311102024

package main

import (
```

```
        "fmt"

    )

const NMAX int = 127

type tabel [NMAX]rune

// Fungsi untuk mengisi array dengan karakter dari input
user

func isiArray(t *tabel, n *int) {

    var ch rune

    *n = 0

    for {

        fmt.Scanf("%c", &ch)

        if ch == '\n' || ch == 'T' {

            break

        }

        t[*n] = ch

        *n++

        if *n >= NMAX {

            break

        }

    }

}
```

```

    }

    }

}

// Fungsi untuk membalikkan isi array
func balikanArray(t *tabel, n int) {

    for i := 0; i < n/2; i++ {

        t[i], t[n-1-i] = t[n-1-i], t[i]

    }

}

// Fungsi untuk mencetak isi array
func cetakArray(t tabel, n int) {

    for i := 0; i < n; i++ {

        fmt.Printf("%c", t[i])

    }

    fmt.Println()

}

// Fungsi untuk memeriksa apakah array membentuk
palindrom

```

```
func palindrom(t tabel, n int) bool {  
  
    for i := 0; i < n/2; i++ {  
  
        if t[i] != t[n-1-i] {  
  
            return false  
  
        }  
  
    }  
  
    return true  
}  
  
func main() {  
  
    var tab tabel  
  
    var n int  
  
  
    // Mengisi array dengan input dari user  
  
    fmt.Print("Masukkan teks: ")  
  
    isiArray(&tab, &n)  
  
  
    // Mencetak array asli  
  
    fmt.Print("Teks: ")  
  
    cetakArray(tab, n)
```

```

        // Membalikkan isi array

        balikkanArray(&tab, n)

        fmt.Print("Reverse teks: ")

        cetakArray(tab, n)

        // Memeriksa apakah array membentuk palindrom

        if palindrom(tab, n) {

            fmt.Println("Teks ini adalah palindrom.")

        } else {

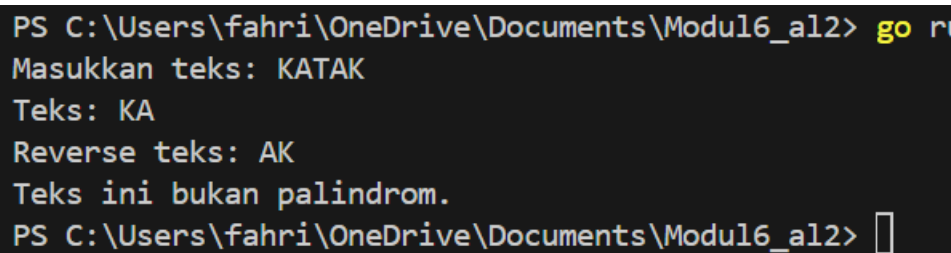
            fmt.Println("Teks ini bukan palindrom.")

        }

    }
}

```

Screenshoot Output



```

PS C:\Users\fahri\OneDrive\Documents\Modul6_a12> go run main.go
Masukkan teks: KATAK
Teks: KA
Reverse teks: AK
Teks ini bukan palindrom.
PS C:\Users\fahri\OneDrive\Documents\Modul6_a12> 

```

Deskripsi Program balikkanArray Fungsi ini membalikkan isi array dengan menukar elemen pertama dengan elemen terakhir, elemen kedua dengan elemen kedua terakhir, dan seterusnya hingga mencapai tengah array. cetakArray Fungsi ini mencetak semua elemen array hingga indeks n ke

layer sebagai sebuah teks. palindrom Fungsi ini memeriksa apakah teks yang ada dalam array merupakan palindrom (teks yang terbaca sama dari depan dan belakang).