

LAPORAN PRAKTIKUM
ALGORITMA DAN PEMROGRAMAN 2
MODUL VII
STRUCT DAN ARRAY



Oleh:

FAJAR FARIZQI AZMI

2311102192

IF-11-02

S1 TEKNIK INFORMATIKA
TELKOM UNIVERSITPURWOKERTO
2024

I. DASAR TEORI

Tipe Bentukan (Structs and Aliases)

1.1. Tipe Alias

Tipe alias memungkinkan programmer untuk mendefinisikan nama baru untuk tipe data yang sudah ada, sehingga lebih mudah dan lebih intuitif untuk digunakan. Misalnya, dalam bahasa pemrograman Go, kita dapat menggunakan kata kunci `type` untuk membuat alias. Contoh:

- | |
|--|
| <ol style="list-style-type: none">1. <code>type</code> bilangan <code>int</code>2. <code>type</code> pecahan <code>float64</code> |
|--|

- II. Dengan cara ini, kita dapat menggunakan bilangan dan pecahan sebagai tipe data baru yang lebih mudah diingat dan digunakan dalam kode.

1.2. Struct

Struct adalah tipe bentukan yang memungkinkan programmer untuk mengelompokkan beberapa data yang memiliki relasi menjadi satu kesatuan. Setiap data dalam struct disimpan dalam field yang berbeda. Contoh definisi struct dalam Go adalah sebagai berikut:

- | |
|---|
| <ol style="list-style-type: none">1. <code>type</code> waktu struct {2. <code>jam</code> <code>int</code>3. <code>menit</code> <code>int</code>4. <code>detik</code> <code>int</code>5. } |
|---|

- III. Dalam contoh ini, waktu adalah struct yang memiliki tiga field: jam, menit, dan detik, semuanya bertipe `int`.

1.3. Kesamaan Tipe

Dalam Go, dua variabel dengan struktur yang sama (nama field dan tipe field yang sama) dianggap memiliki tipe yang sama, meskipun nama struct-nya berbeda. Ini

memudahkan penggunaan struct dalam program.

2. Array

2.1. Definisi Array

Array adalah kumpulan elemen dengan tipe yang sama, di mana ukuran (jumlah elemen) ditentukan saat deklarasi dan bersifat tetap selama eksekusi program.

Contoh

deklarasi array dalam Go:

```
1. var arr [73]CircType
```

IV. Di sini, arr adalah array dengan 73 elemen yang bertipe CircType.

2.2. Mengakses Elemen Array

Indeks array dimulai dari 0, sehingga elemen pertama dapat diakses dengan arr[0], dan elemen terakhir dengan arr[len(arr)-1]. Fungsi len dapat digunakan untuk mendapatkan jumlah elemen dalam array.

2.3. Slice

Slice adalah array dinamik yang dapat berubah ukuran. Slice dapat dideklarasikan tanpa menentukan jumlah elemen, dan dapat diperluas menggunakan fungsi built-in append.

Contoh deklarasi slice:

```
1. var s []int
2. s = append(s, 1, 2, 3)
```

V. Fungsi len dan cap dapat digunakan untuk mengetahui ukuran dan kapasitas slice.

2.4. Map

Map adalah tipe data yang menyimpan pasangan kunci-nilai, di mana kunci tidak harus

berupa integer. Contoh deklarasi map dalam Go:

```
1. var dct map[string]int
2. dct =
```

```
make(map[string]int)
```

```
3. dct["john"] = 1
```

VI. Map memungkinkan akses cepat ke nilai berdasarkan kunci yang

II. GUIDED

Guided 1

Source code :

```
package main

import (
    "fmt"
    "sort"
)

type Mahasiswa struct {
    Nama      string
    Matematika int
    Fisika    int
    Kimia     int
    RataRata  float64
}

func hitungRataRata(m *Mahasiswa) {
    total := m.Matematika + m.Fisika + m.Kimia
    m.RataRata = float64(total) / 3.0
}

func main() {
    mahasiswa := []Mahasiswa{
        {"Ali", 85, 90, 80, 0},
        {"Budi", 70, 75, 80, 0},
        {"Cici", 90, 85, 95, 0},
        {"Doni", 60, 65, 70, 0},
        {"Eka", 100, 95, 90, 0},
    }
}
```

```

    for i := range mahasiswa {
        hitungRataRata(&mahasiswa[i])
    }

    sort.Slice(mahasiswa, func(i, j int) bool {
        return mahasiswa[i].RataRata > mahasiswa[j].RataRata
    })

    fmt.Println("Peringkat mahasiswa berdasarkan rata-rata nilai:")
    for i, m := range mahasiswa {
        fmt.Printf("%d, %s - Rata-rata: %.2f (Matematika: %d, Fisika: %d,
        Kimia: %d)\n",
            i+1, m.Nama, m.RataRata, m.Matematika, m.Fisika, m.Kimia)
    }
}

```

Screenshot output :

```

PS D:\semester3\Alpro 2 materi dan praktikum\Fajar Farizqi Azmi_2311102192_modul 7> go run "d:\semester3\Alpro 2 materi dan praktikum\Fajar F
arizqi Azmi_2311102192_modul 7\Guided\guided1.go"
Peringkat mahasiswa berdasarkan rata-rata nilai:
1, Eka - Rata-rata: 95.00 (Matematika: 100, Fisika: 95, Kimia: 90)
2, Cici - Rata-rata: 90.00 (Matematika: 90, Fisika: 85, Kimia: 95)
3, Ali - Rata-rata: 85.00 (Matematika: 85, Fisika: 90, Kimia: 80)
4, Budi - Rata-rata: 75.00 (Matematika: 70, Fisika: 75, Kimia: 80)
5, Doni - Rata-rata: 65.00 (Matematika: 60, Fisika: 65, Kimia: 70)
PS D:\semester3\Alpro 2 materi dan praktikum\Fajar Farizqi Azmi_2311102192_modul 7>

```

Deskripsi program :

Program ini mendefinisikan sebuah program dalam bahasa Go untuk menghitung rata-rata nilai dari beberapa mahasiswa, kemudian mengurutkan mahasiswa berdasarkan nilai rata-rata tersebut secara menurun. Hasilnya adalah daftar mahasiswa yang terurut berdasarkan rata-rata nilai dari yang tertinggi ke terendah.

Guided 2

Source code :

```
package main

import "fmt"

func main() {
    // Membuat map dengan NIM sebagai kunci dan Nama sebagai nilai
    mahasiswa := map[string]string{
        "20231001": "Andi",
        "20231002": "Budi",
        "20231003": "Cici",
    }

    // Menambahkan data baru ke map
    mahasiswa["20231004"] = "Dedi"

    // Menampilkan seluruh isi map dalam format kolom dan baris
    fmt.Println("Daftar Mahasiswa:")
    fmt.Println("NIM\t\tNama")
    fmt.Println("-----")
    for nim, nama := range mahasiswa {
        fmt.Printf("%s\t%s\n", nim, nama)
    }

    // Mengakses data berdasarkan NIM
    nim := "20231002"
    fmt.Println("\nNama Mahasiswa dengan NIM", nim, "adalah",
mahasiswa[nim])

    // Menghapus data berdasarkan NIM
    delete(mahasiswa, "20231003")

    // Menampilkan isi map setelah data dihapus dalam format kolom dan baris
    fmt.Println("\nDaftar Mahasiswa setelah dihapus:")
}
```

```

    fmt.Println("NIM\t\tNama")
    fmt.Println("-----")
    for nim, nama := range mahasiswa {
        fmt.Printf("%s\t%s\n", nim, nama)
    }
}

```

Screenshot output :

```

arizqi Azmi_2311102192_modul 7\Guided\guided2.go
Daftar Mahasiswa:
NIM      Nama
-----
20231001  Andi
20231002  Budi
20231003  Cici
20231004  Dedi

Nama Mahasiswa dengan NIM 20231002 adalah Budi

Daftar Mahasiswa setelah dihapus:
NIM      Nama
-----
20231001  Andi
20231002  Budi
20231004  Dedi
PS D:\semester3\Alpro 2 materi dan praktikum\Fajar Farizqi Azmi_2311102192_modul 7>

```

Deskripsi program :

Program ini memungkinkan pengguna untuk:

1. Menambahkan data mahasiswa (NIM dan Nama).
2. Menampilkan seluruh data mahasiswa dalam format tabel.
3. Mengakses data mahasiswa berdasarkan NIM.
4. Menghapus data mahasiswa dari map.
5. Menampilkan data mahasiswa yang tersisa setelah penghapusan.

Map di sini berfungsi sebagai struktur penyimpanan data yang efisien, memungkinkan pencarian, penambahan, dan penghapusan data dengan cepat.

III. UNGUIDED

Unguided 1

Source code :

```
// Fajar Farizqi Azmi  
// 2311102192  
  
package main  
  
import (  
    "fmt"  
    "math"  
)  
  
// Struct untuk merepresentasikan titik  
type Titik struct {  
    x, y int  
}
```

```

// Struct untuk merepresentasikan lingkaran
type Lingkaran struct {
    titikPusat Titik
    jariJari int
}

// Fungsi untuk mengecek apakah suatu titik berada di dalam lingkaran
func apakahTitikDiDalamLingkaran(t Titik, l Lingkaran) bool {
    jarak := math.Sqrt(math.Pow(float64(t.x-l.titikPusat.x), 2) +
math.Pow(float64(t.y-l.titikPusat.y), 2))
    return jarak <= float64(l.jariJari)
}

func main() {
    // Masukan baris pertama untuk lingkaran 1
    var x1, y1, r1 int
    fmt.Printf("Masukkan untuk lingkaran 1: ")
    fmt.Scan(&x1, &y1, &r1)
    lingkaran1 := Lingkaran{Titik{x1, y1}, r1}

    // Masukan baris kedua untuk lingkaran 2
    var x2, y2, r2 int
    fmt.Printf("Masukkan untuk lingkaran 2: ")
    fmt.Scan(&x2, &y2, &r2)
    lingkaran2 := Lingkaran{Titik{x2, y2}, r2}

    // Masukan baris ketiga untuk titik sembarang
    var x, y int
    fmt.Printf("Masukkan untuk titik sembarang: ")
    fmt.Scan(&x, &y)
    titik := Titik{x, y}

```

```

// Mengecek apakah titik berada di dalam lingkaran 1, lingkaran 2,
atau keduanya
diLingkaran1 := apakahTitikDiDalamLingkaran(titik, lingkaran1)
diLingkaran2 := apakahTitikDiDalamLingkaran(titik, lingkaran2)

// Menentukan keluaran berdasarkan hasil pengecekan
if diLingkaran1 && diLingkaran2 {
    fmt.Println("Titik di dalam lingkaran 1 dan 2")
} else if diLingkaran1 {
    fmt.Println("Titik di dalam lingkaran 1")
} else if diLingkaran2 {
    fmt.Println("Titik di dalam lingkaran 2")
} else {
    fmt.Println("Titik di luar lingkaran 1 dan 2")
}
}

```

Screenshot output :

```

PS D:\semester3\Alpro 2 materi dan praktikum\Fajar Farizqi Azmi_2311102192_modul 7> go run "d:\semester3\Alpro 2 materi dan praktikum\Fajar Farizqi Azmi_2311102192_modul 7\Unguided\unguided1.go"
Masukkan untuk lingkaran 1: 1 1 5
Masukkan untuk lingkaran 2: 8 8 4
Masukkan untuk titik sembarang: 2 2
Titik di dalam lingkaran 1
PS D:\semester3\Alpro 2 materi dan praktikum\Fajar Farizqi Azmi_2311102192_modul 7>

```

Deskripsi program :

Program ini digunakan untuk menentukan apakah sebuah titik berada di dalam salah satu, kedua, atau di luar dua lingkaran yang diberikan berdasarkan input koordinat dan jari-jari lingkaran. Dengan menggunakan konsep jarak Euclidean, program ini melakukan pengecekan posisi relatif titik terhadap lingkaran secara efisien dan memberikan hasil sesuai dengan kondisi geometris yang diuji.

Unguided 2

Sourch code :

```
// Fajar Farizqi Azmi
// 2311102192

package main

import (
    "fmt"
    "math"
)

func displayArray(arr []int) {
    fmt.Println("Isi array:", arr)
}

func displayElementsByIndex(arr []int, start int, step int, desc string) {
    fmt.Printf("Elemen dengan indeks %s: ", desc)
    for i := start; i < len(arr); i += step {
        fmt.Print(arr[i], " ")
    }
    fmt.Println()
}

func deleteElementAtIndex(arr *[]int, index int) {
    if index >= 0 && index < len(*arr) {
        *arr = append((*arr)[:index], (*arr)[index+1:]...)
        displayArray(*arr)
    } else {
        fmt.Println("Indeks tidak valid.")
    }
}

func calculateAverage(arr []int) float64 {
    if len(arr) == 0 {
```

```

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

func calculateStandardDeviation(arr []int, avg float64) float64 {
    if len(arr) == 0 {
        return 0
    }
    var varianceSum float64
    for _, v := range arr {
        varianceSum += math.Pow(float64(v)-avg, 2)
    }
    return math.Sqrt(varianceSum / float64(len(arr)))
}

func countFrequency(arr []int, num int) int {
    count := 0
    for _, v := range arr {
        if v == num {
            count++
        }
    }
    return count
}

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

```

```
fmt.Scan(&N)
```

```
arr := make([]int, N)
```

```
for i := 0; i < N; i++ {
```

```
    fmt.Printf("Masukkan elemen ke-%d: ", i)
```

```
    fmt.Scan(&arr[i])
```

```
}
```

```
displayArray(arr)
```

```
displayElementsByIndex(arr, 1, 2, "ganjil")
```

```
displayElementsByIndex(arr, 0, 2, "genap")
```

```
var x int
```

```
fmt.Print("Masukkan bilangan x untuk kelipatan indeks: ")
```

```
fmt.Scan(&x)
```

```
displayElementsByIndex(arr, x, x, fmt.Sprintf("kelipatan %d", x))
```

```
var index int
```

```
fmt.Print("Masukkan indeks untuk menghapus elemen: ")
```

```
fmt.Scan(&index)
```

```
deleteElementAtIndex(&arr, index)
```

```
avg := calculateAverage(arr)
```

```
fmt.Println("Rata-rata array:", avg)
```

```
stdDev := calculateStandardDeviation(arr, avg)
```

```
fmt.Println("Simpangan baku (standar deviasi):", stdDev)
```

```
var num int
```

```
fmt.Print("Masukkan bilangan untuk menghitung frekuensi: ")
```

```
fmt.Scan(&num)
```

```
fmt.Printf("Frekuensi bilangan %d: %d\n", num, countFrequency(arr,  
num))
```

}

Screenshot output :

```
PS D:\semester3\Alpro 2 materi dan praktikum\Fajar Farizqi Azmi_2311102192_modul 7> go run "d:\semester3\Alpro 2 materi dan praktikum\Fajar Farizqi Azmi_2311102192_modul 7\Unguided\unguided2.go"
Masukkan jumlah elemen array: 5
Masukkan elemen ke-0: 23
Masukkan elemen ke-1: 4
Masukkan elemen ke-2: 6
Masukkan elemen ke-3: 8
Masukkan elemen ke-4: 9
Isi array: [23 4 6 8 9]
Elemen dengan indeks ganjil: 4 8
Elemen dengan indeks genap: 23 6 9
Masukkan bilangan x untuk kelipatan indeks: 
```

Deskripsi program :

Program ini menghitung statistik dasar dan melakukan operasi pada elemen-elemen

array. Berikut adalah fungsionalitasnya:

1. Input Array:

Pengguna diminta memasukkan jumlah elemen dan nilai dari elemen-elemen array.

2. Menampilkan Isi Array:

Program menampilkan seluruh elemen, elemen dengan indeks ganjil, dan elemen dengan indeks genap.

3. Kelipatan Indeks:

Program meminta input angka x dari pengguna dan menampilkan elemen pada indeks yang merupakan kelipatan dari x.

4. Hapus Elemen:

Program meminta indeks untuk dihapus, kemudian menampilkan array yang diperbarui.

5. Statistik:

- Rata-rata: Menghitung rata-rata elemen array.
- Standar Deviasi: Menghitung standar deviasi berdasarkan rata-rata.

6. Frekuensi:

Program menghitung dan menampilkan frekuensi kemunculan suatu bilangan dalam array.

Program ini mencetak hasil setiap operasi secara langsung.

Unguided 3

Sourch code :

```
// Fajar Farizqi Azmi
// 2311102192

package main

import "fmt"

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

    fmt.Print("Klub A: ")
    fmt.Scanln(&klubA)
    fmt.Print("Klub B: ")
    fmt.Scanln(&klubB)

    pertandingan := 1
    for {
        fmt.Printf("Pertandingan %d: ", pertandingan)
        fmt.Scanln(&skorA, &skorB)

        if skorA < 0 || skorB < 0 {
            break
        }

        if skorA > skorB {
            pemenang = append(pemenang, klubA)
        } else if skorA < skorB {
            pemenang = append(pemenang, klubB)
        } else {
            pemenang = append(pemenang, "Draw")
        }
    }
}
```



```

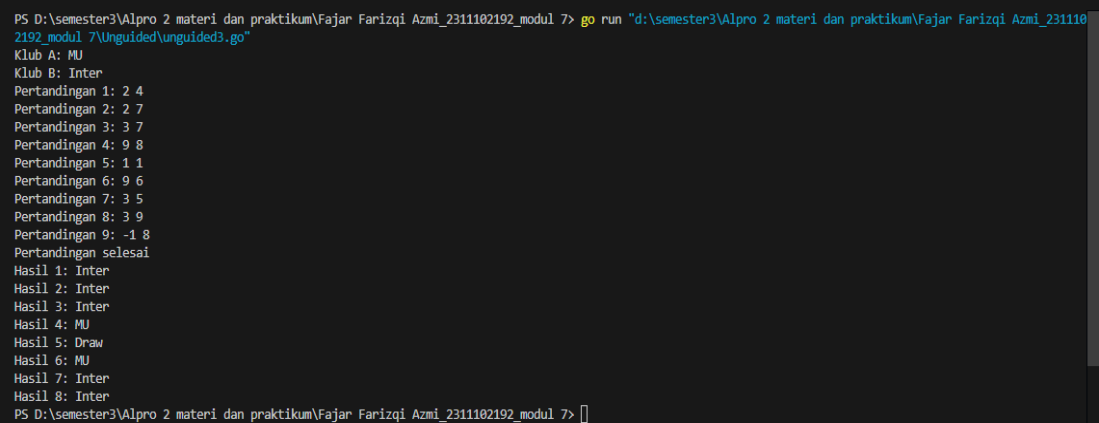
    }

    pertandingan++
}

fmt.Println("Pertandingan selesai")
for i, p := range pemenang {
    fmt.Printf("Hasil %d: %s\n", i+1, p)
}
}

```

Screenshot output :



```

PS D:\semester3\Alpro 2 materi dan praktikum\Fajar Farizqi Azmi_2311102192_modul 7> go run "d:\semester3\Alpro 2 materi dan praktikum\Fajar Farizqi Azmi_2311102192_modul 7\Unguided3.go"
Klub A: MU
Klub B: Inter
Pertandingan 1: 2 4
Pertandingan 2: 2 7
Pertandingan 3: 3 7
Pertandingan 4: 9 8
Pertandingan 5: 1 1
Pertandingan 6: 9 6
Pertandingan 7: 3 5
Pertandingan 8: 3 9
Pertandingan 9: -1 8
Pertandingan selesai
Hasil 1: Inter
Hasil 2: Inter
Hasil 3: Inter
Hasil 4: MU
Hasil 5: Draw
Hasil 6: MU
Hasil 7: Inter
Hasil 8: Inter
PS D:\semester3\Alpro 2 materi dan praktikum\Fajar Farizqi Azmi_2311102192_modul 7>

```

Deskripsi program :

Program ini memungkinkan pengguna untuk mencatat dan melihat hasil pertandingan antar dua klub sepak bola. Input terus diterima hingga terdapat skor negatif, dan hasil pertandingan dicetak secara berurutan setelah pertandingan selesai.

Unguided 4

Source code :

```

// FAJAR FARIZQI AZMI

// 2311102192

package main

```

```

import "fmt"

const NMAX int = 127

type tabel struct {
    tab [NMAX]rune
    m    int
}

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

    fmt.Print("Masukkan karakter (akhiri dengan titik): ")
    var input rune
    for *n = 0; *n < NMAX; *n++ {
        fmt.Scanf("%c", &input)
        if input == '.' {
            break
        }
        t.tab[*n] = input
    }
    t.m = *n
}

func cetakArray(t tabel, n int) {

    for i := 0; i < n; i++ {
        fmt.Printf("%c", t.tab[i])
    }
    fmt.Println()
}

func balikanArray(t *tabel, n int) {

```

```

    for i := 0; i < n/2; i++ {
        t.tab[i], t.tab[n-i-1] = t.tab[n-i-1], t.tab[i]
    }
}

```

```

func palindrom(t tabel, n int) bool {

```

```

    for i := 0; i < n/2; i++ {
        if t.tab[i] != t.tab[n-i-1] {
            return false
        }
    }
    return true
}

```

```

func main() {
    var tab tabel
    var m int

```

```

    isiArray(&tab, &m)

```

```

    fmt.Print("Teks: ")
    cetakArray(tab, m)

```

```

    fmt.Print("Reverse teks: ")
    balikanArray(&tab, m)
    cetakArray(tab, m)

```

```

    isPalindrom := palindrom(tab, m)
    if isPalindrom {
        fmt.Println("Palindrom: true")
    } else {

```

```
        fmt.Println("Palindrom: false")
    }
}
```

Screenshot output :

```
PS D:\semester3\Alpro 2 materi dan praktikum\Fajar Farizqi Azmi_2311102192_modul 7> go run "d:\semester3\Alpro 2 materi dan praktikum\Fajar Farizqi Azmi_2311102192_modul 7\Unguided\unguided4.go"
Masukkan karakter (akhiri dengan titik): senang.
Teks: senang
Reverse teks: gnanes
Palindrom: false
PS D:\semester3\Alpro 2 materi dan praktikum\Fajar Farizqi Azmi_2311102192_modul 7> |
```

```
PROBLEMS 8 OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS D:\semester3\Alpro 2 materi dan praktikum\Fajar Farizqi Azmi_2311102192_modul 7> go run "d:\semester3\Alpro 2 materi dan praktikum\Fajar Farizqi Azmi_2311102192_modul 7\Unguided\unguided4.go"
Masukkan karakter (akhiri dengan titik): katak.
Teks: katak
Reverse teks: katak
Palindrom: true
PS D:\semester3\Alpro 2 materi dan praktikum\Fajar Farizqi Azmi_2311102192_modul 7> |
```

Deskripsi program :

Program untuk memasukkan teks berupa karakter, membalikkan urutan karakter, dan memeriksa apakah teks tersebut adalah palindrom. Program ini menggunakan array dan operasi pertukaran elemen untuk membalikkan urutan teks, serta memanfaatkan perbandingan elemen array untuk memeriksa palindrom.