

**LAPORAN PRAKTIKUM  
ALGORITMA PEMROGRAMAN 2**

**MODUL VII  
STRUCK DAN ARRAY**



Oleh:

ALIFATUS SHABRINA AMALIA

NIM:

2311102225

**S1 TEKNIK INFORMATIKA  
TELKOM UNIVERSITY PURWOKERTO  
2024**

## I. DASAR TEORI

Array adalah struktur data yang menyimpan sekumpulan data atau elemen yang memiliki tipe data yang sama dalam urutan tertentu. Setiap elemen dalam array memiliki indeks (nomor posisi) yang dapat diakses secara langsung, dengan elemen pertama berada pada indeks 0. Array bisa statis (dengan ukuran tetap) atau dinamis (ukuran bisa berubah).

Sedangkan stack (tumpukan) adalah struktur data berbasis LIFO (Last In, First Out), yang berarti elemen terakhir yang ditambahkan adalah elemen pertama yang akan dihapus. Stack memungkinkan operasi **push** (menambahkan elemen) dan **pop** (menghapus elemen). Ini digunakan di banyak aplikasi, seperti pemrosesan rekursif, penelusuran pohon, dan manajemen undo-redo pada editor teks.

## II. GUIDED

### 1. Guided 1

#### Source code

```
package main

import (
    "fmt"
    "sort"
)

// Struktur untuk menampung data mahasiswa
type Mahasiswa struct {
    Nama      string
    Matematika int
    Fisika     int
    Kimia      int
    RataRata  float64
}

// Fungsi untuk menghitung rata-rata nilai tiap mahasiswa
func hitungRataRata(m *Mahasiswa) {
    total := m.Matematika + m.Fisika + m.Kimia
    m.RataRata = float64(total) / 3.0
}

// Fungsi utama untuk mengelola dan mengurutkan data mahasiswa
// berdasarkan nilai rata-rata
func main() {
```

```

// Array untuk menampung data mahasiswa

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},

}

// Menghitung rata-rata nilai tiap mahasiswa

for i := range mahasiswa {

    hitungRataRata(&mahasiswa[i])

}

// Mengurutkan mahasiswa berdasarkan nilai rata-rata
(descending)

sort.Slice(mahasiswa, func(i, j int) bool {

    return mahasiswa[i].RataRata > mahasiswa[j].RataRata

})

// Menampilkan hasil

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)

}

}

```

## Screenshoot program

```
PS D:\Praktikum\Alpro2\Modul7\Guided> go run "d:\Praktikum\Alpro2\Modul7\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. Dodi - Rata-rata: 65.00 (Matematika: 60, Fisika: 65, Kimia: 70)
PS D:\Praktikum\Alpro2\Modul7\Guided>
```

## Deskripsi program

Program di atas mengelola data nilai beberapa mahasiswa dalam mata pelajaran Matematika, Fisika, dan Kimia. Untuk setiap mahasiswa, program menghitung rata-rata nilai, kemudian mengurutkan mahasiswa berdasarkan nilai rata-rata mereka dalam urutan menurun. Setelah diurutkan, program menampilkan daftar peringkat mahasiswa dengan nama, nilai rata-rata, serta nilai individu mereka untuk setiap mata pelajaran.

## 2. 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)
}

}
```

## Screenshoot program

```
PS D:\Praktikum\Alpro2\Modul7\Guided> go run "d:\Praktikum\Alpro2\Modul7\Guided\tempCodeRunnerFile.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:\Praktikum\Alpro2\Modul7\Guided>
```

## Deskripsi program

Program di atas membuat dan mengelola data mahasiswa dalam bentuk map dengan NIM sebagai kunci dan nama sebagai nilai, memungkinkan penambahan, pengaksesan, serta penghapusan data, dan menampilkan isi map sebelum dan setelah data dihapus.

### III. UNGUIDED

#### 1. Unguided 1

##### Source code

```
package main

import (
    "fmt"
    "math"
)

type Titik225 struct {
    x, y int
}

type Lingkaran225 struct {
    center Titik225
    radius int
}

func jarak(p, q Titik225) float64 {
    return math.Sqrt(float64((p.x-q.x)*(p.x-q.x) + (p.y-q.y)*(p.y-q.y)))
}

func didalam(c Lingkaran225, p Titik225) bool {
    return jarak(c.center, p) <= float64(c.radius)
}
```



```
func main() {

    var cx1, cy1, r1 int

    var cx2, cy2, r2 int

    var tx, ty int


    fmt.Println("Masukkan titik pusat dan radius lingkaran 1 (cx1, cy1, r1):")

    fmt.Scan(&cx1, &cy1, &r1)

    fmt.Println("Masukkan titik pusat dan radius lingkaran 2 (cx2, cy2, r2):")

    fmt.Scan(&cx2, &cy2, &r2)

    fmt.Println("Masukkan koordinat titik sembarang (tx, ty):")

    fmt.Scan(&tx, &ty)


    lingkaran1 := Lingkaran225{Titik225{cx1, cy1}, r1}
    lingkaran2 := Lingkaran225{Titik225{cx2, cy2}, r2}
    titik := Titik225{tx, ty}


    diDalamLingkaran1 := didalam(lingkaran1, titik)
    diDalamLingkaran2 := didalam(lingkaran2, titik)


    if diDalamLingkaran1 && diDalamLingkaran2 {

        fmt.Println("Titik di dalam lingkaran 1 dan 2")

    } else if diDalamLingkaran1 {

        fmt.Println("Titik di dalam lingkaran 1")

    } else if diDalamLingkaran2 {

        fmt.Println("Titik di dalam lingkaran 2")

    } else {
```

```

        fmt.Println("Titik di luar lingkaran 1 dan 2")
    }
}

```

## Screenshoot program

```

PS D:\Praktikum\Alpro2\Modul7\Unguided> go run "d:\Praktikum\Alpro2\Modul7\Unguided\unguided1.go"
Masukkan titik pusat dan radius lingkaran 1 (cx1, cy1, r1):
1 1 5
Masukkan titik pusat dan radius lingkaran 2 (cx2, cy2, r2):
10 10 5
Masukkan koordinat titik sembarang (tx, ty):
3 3
Titik di dalam lingkaran 1
PS D:\Praktikum\Alpro2\Modul7\Unguided>

```

## Deskripsi program

Program ini menentukan posisi suatu titik relatif terhadap dua lingkaran, apakah titik tersebut berada di dalam, di luar, atau hanya di dalam salah satu lingkaran. Berikut adalah deskripsi singkat fungsionalitasnya:

1. Struct Titik225 dan Lingkaran225:
  - Titik225: Menyimpan koordinat (x, y) untuk merepresentasikan titik.
  - Lingkaran225: Menyimpan titik pusat lingkaran (dalam tipe Titik225) dan radius lingkaran.
2. Fungsi jarak:
  - Menghitung jarak Euclidean antara dua titik p dan q menggunakan rumus  $\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$ .
3. Fungsi didalam:
  - Menentukan apakah suatu titik p berada di dalam lingkaran c dengan memeriksa apakah jarak antara titik pusat lingkaran dan titik p kurang dari atau sama dengan radius lingkaran.
4. Fungsi main:

- Meminta input dari pengguna untuk titik pusat dan radius dari kedua lingkaran, serta koordinat titik sembarang.
- Membuat objek lingkaran dan titik berdasarkan input tersebut.
- Memeriksa posisi titik relatif terhadap kedua lingkaran:
  - Jika titik berada di dalam kedua lingkaran, menampilkan pesan "Titik di dalam lingkaran 1 dan 2".
  - Jika titik hanya berada di dalam lingkaran 1, menampilkan "Titik di dalam lingkaran 1".
  - Jika titik hanya berada di dalam lingkaran 2, menampilkan "Titik di dalam lingkaran 2".
  - Jika titik di luar kedua lingkaran, menampilkan "Titik di luar lingkaran 1 dan 2".

Contoh Output:

- Berdasarkan input posisi titik dan lingkaran, program akan mengidentifikasi dan menampilkan posisi titik tersebut relatif terhadap lingkaran 1 dan 2.

## 2. Unguided 2

### Source code

```
package main

import (
    "fmt"
    "math"
)

func tampilkanArray225(arr []int) {
    for i, val := range arr {
        fmt.Printf("Indeks %d: %d\n", i, val)
    }
}
```

```

    }

}

func tampilkanIndeksGanjil225(arr []int) {
    for i := 1; i < len(arr); i += 2 {
        fmt.Printf("Indeks %d (ganjil): %d\n", i, arr[i])
    }
}

func tampilkanIndeksGenap225(arr []int) {
    for i := 0; i < len(arr); i += 2 {
        fmt.Printf("Indeks %d (genap): %d\n", i, arr[i])
    }
}

func tampilkanIndeksKelipatan225(arr []int, x int) {
    for i := 0; i < len(arr); i += x {
        fmt.Printf("Indeks %d (kelipatan %d): %d\n", i, x, arr[i])
    }
}

func hapusElemen225(arr []int, indeks int) []int {
    return append(arr[:indeks], arr[indeks+1:]...)
}

func rataRata225(arr []int) float64 {
    total := 0

```

```

    for _, val := range arr {
        total += val
    }

    return float64(total) / float64(len(arr))
}

func standarDeviasi225(arr []int) float64 {
    mean := rataRata225(arr)

    var sum float64

    for _, val := range arr {
        sum += math.Pow(float64(val)-mean, 2)
    }

    return math.Sqrt(sum / float64(len(arr)))
}

func hitungFrekuensi225(arr []int, target int) int {
    frekuensi := 0

    for _, val := range arr {
        if val == target {
            frekuensi++
        }
    }

    return frekuensi
}

func main() {
    var n int

```

```
fmt.Print("Masukkan jumlah elemen dalam array: ")

fmt.Scan(&n)

arr := make([]int, n)

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

fmt.Println("Seluruh isi array:")
tampilkanArray225(arr)

fmt.Println("Elemen dengan indeks ganjil:")
tampilkanIndeksGanjil225(arr)

fmt.Println("Elemen dengan indeks genap:")
tampilkanIndeksGenap225(arr)

var x int

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

fmt.Println("Elemen dengan indeks kelipatan", x, ":")
tampilkanIndeksKelipatan225(arr, x)

var indeksHapus int
```

```
    fmt.Print("Masukkan indeks yang ingin dihapus: ")

    fmt.Scan(&indeksHapus)

    if indeksHapus >= 0 && indeksHapus < len(arr) {

        arr = hapusElemen225(arr, indeksHapus)

        fmt.Println("Array setelah elemen dihapus:")

        tampilkanArray225(arr)

    } else {

        fmt.Println("Indeks tidak valid!")

    }

    fmt.Printf("Rata-rata: %.2f\n", rataRata225(arr))

    fmt.Printf("Standar deviasi: %.2f\n", standarDeviasi225(arr))

    var target int

    fmt.Print("Masukkan bilangan untuk menghitung frekuensinya: ")

    fmt.Scan(&target)

    fmt.Printf("Frekuensi bilangan %d: %d kali\n", target,
hitungFrekuensi225(arr, target))

}
```

## Screenshoot program

```
PS D:\Praktikum\Alpro2\Modul7\Unguided> go run "d:\Praktikum\Alpro2\Modul7\Unguided\unguided2.go"
Masukkan jumlah elemen dalam array: 5
Masukkan elemen-elemen array:
Elemen ke-0: 1
Elemen ke-1: 2
Elemen ke-2: 3
Elemen ke-3: 4
Elemen ke-4: 5
Seluruh isi array:
Indeks 0: 1
Indeks 1: 2
Indeks 2: 3
Indeks 3: 4
Indeks 4: 5
Elemen dengan indeks ganjil:
Indeks 1 (ganjil): 2
Indeks 3 (ganjil): 4
Elemen dengan indeks genap:
Indeks 0 (genap): 1
Indeks 2 (genap): 3
Indeks 4 (genap): 5
Masukkan bilangan x untuk kelipatan indeks: 2
Elemen dengan indeks kelipatan 2 :
Indeks 0 (kelipatan 2): 1
Indeks 2 (kelipatan 2): 3
Indeks 4 (kelipatan 2): 5
Masukkan indeks yang ingin dihapus: 1
Array setelah elemen dihapus:
Indeks 0: 1
Indeks 1: 3
Indeks 2: 4
Indeks 3: 5
Rata-rata: 3.25
Standar deviasi: 1.48
Masukkan bilangan untuk menghitung frekuensinya: 3
Frekuensi bilangan 3: 1 kali
PS D:\Praktikum\Alpro2\Modul7\Unguided> █
```

## Deskripsi program

Program ini ditulis dalam bahasa Go untuk melakukan operasi pada array bilangan bulat. Berikut adalah fungsi-fungsi utamanya:

1. `tampilkanArray225`: Menampilkan seluruh elemen array beserta indeksnya.
2. `tampilkanIndeksGanjil225`: Menampilkan elemen-elemen array yang berada pada indeks ganjil.
3. `tampilkanIndeksGenap225`: Menampilkan elemen-elemen array yang berada pada indeks genap.



4. `tampilkanIndeksKelipatan225`: Menampilkan elemen-elemen array pada indeks yang merupakan kelipatan dari bilangan yang dimasukkan pengguna (x).
5. `hapusElemen225`: Menghapus elemen pada indeks tertentu dalam array dan mengembalikan array baru tanpa elemen tersebut.
6. `rataRata225`: Menghitung rata-rata dari seluruh elemen dalam array.
7. `standarDeviasi225`: Menghitung standar deviasi dari elemen-elemen dalam array menggunakan rata-rata sebagai dasar perhitungannya.
8. `hitungFrekuensi225`: Menghitung berapa kali suatu bilangan (target) muncul di dalam array.
9. `main`: Mengelola interaksi pengguna, seperti:
  - Mengisi array.
  - Menampilkan elemen array sesuai dengan kriteria tertentu (ganjil, genap, kelipatan x).
  - Menghapus elemen pada indeks tertentu.
  - Menghitung dan menampilkan rata-rata, standar deviasi, serta frekuensi suatu bilangan dalam array.

Contoh Output:

- Program akan meminta jumlah elemen dan isi array dari pengguna.
- Menampilkan elemen sesuai kriteria (ganjil, genap, kelipatan x).
- Menghapus elemen sesuai indeks, dan menghitung rata-rata, standar deviasi, serta frekuensi elemen yang diinginkan dalam array.

### 3. Unguided 3

#### Source code

```
package main

import (
    "fmt"
```

```

)

// Fungsi utama

func main() {

    var klubA, klubB string

    var skorA, skorB int

    var hasil []string


    fmt.Print("Masukkan nama Klub A: ")

    fmt.Scanln(&klubA)

    fmt.Print("Masukkan nama Klub B: ")

    fmt.Scanln(&klubB)


    for pertandingan := 1; ; pertandingan++ {

        fmt.Printf("Pertandingan %d: ", pertandingan)

        fmt.Scan(&skorA, &skorB)


        if skorA < 0 || skorB < 0 {

            fmt.Println("Skor negatif terdeteksi, program dihentikan.")

            break

        }


        if skorA > skorB {

            hasil = append(hasil, klubA)

            fmt.Printf("Hasil %d: %s menang\n", pertandingan, klubA)

        } else if skorB > skorA {

```

```
        hasil = append(hasil, klubB)

        fmt.Printf("Hasil %d: %s menang\n", pertandingan,
klubB)

    } else {

        hasil = append(hasil, "Draw")

        fmt.Printf("Hasil %d: Draw\n", pertandingan)

    }

}

fmt.Println("\nDaftar klub yang memenangkan pertandingan:")

for i, pemenang := range hasil {

    if pemenang != "Draw" {

        fmt.Printf("Pertandingan %d: %s\n", i+1, pemenang)

    }

}

}
```

## Screenshoot program

```
PS D:\Praktikum\Alpro2\Modul7\Unguided> go run "d:\Praktikum\Alpro2\Modul7\Unguided\unguided3.go"
Masukkan nama Klub A: Lavani
Masukkan nama Klub B: Samator
Pertandingan 1: 3 0
Hasil 1: Lavani menang
Pertandingan 2: 3 2
Hasil 2: Lavani menang
Pertandingan 3: 2 3
Hasil 3: Samator menang
Pertandingan 4: 3 1
Hasil 4: Lavani menang
Pertandingan 5: 0 3
Hasil 5: Samator menang
Pertandingan 6: 1 3
Hasil 6: Samator menang
Pertandingan 7: 3 0
Hasil 7: Lavani menang
Pertandingan 8: -3 2
Skor negatif terdeteksi, program dihentikan.

Daftar klub yang memenangkan pertandingan:
Pertandingan 1: Lavani
Pertandingan 2: Lavani
Pertandingan 3: Samator
Pertandingan 4: Lavani
Pertandingan 5: Samator
Pertandingan 6: Samator
Pertandingan 7: Lavani
PS D:\Praktikum\Alpro2\Modul7\Unguided> |
```

## Deskripsi program

Program ini merekap hasil pertandingan antara dua klub bola voli dan menampilkan klub yang memenangkan tiap pertandingan. Berikut deskripsi singkat dari setiap bagian kode:

1. Deklarasi Variabel:
  - o klubA dan klubB menyimpan nama kedua klub.
  - o skorA dan skorB menyimpan skor masing-masing klub dalam satu pertandingan.
  - o hasil adalah array dinamis yang menyimpan hasil pertandingan untuk tiap putaran.
2. Input Nama Klub:
  - o Meminta input nama kedua klub (klubA dan klubB).
3. Pengulangan Pertandingan:
  - o Menggunakan loop tanpa batas untuk merekam skor setiap pertandingan.

- Jika skor negatif dimasukkan, loop berhenti, dan program mengakhiri pencatatan.
- Program memeriksa skor, menambahkan nama pemenang atau "Draw" ke dalam hasil, dan menampilkan hasil tiap pertandingan.

#### 4. Menampilkan Daftar Pemenang:

- Setelah loop selesai, program menampilkan daftar pertandingan yang dimenangkan oleh klub tertentu saja (tidak menampilkan pertandingan yang berakhir "Draw").

Contoh Output:

- Jika klubA menang pada pertandingan 1, klubB pada pertandingan 2, dan pertandingan 3 adalah draw, maka program akan menampilkan daftar klub yang menang sesuai urutan tersebut.

## 4. Unguided 4

### Source code

```
package main

import (
    "fmt"
)

const NMAX int = 127

type tabel [NMAX]rune

func isiArray(t *tabel, n *int) {
    var input rune
    *n = 0
```

```

    fmt.Print("Masukkan teks (akhiri dengan titik): ")

    for *n < NMAX {

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

        if input == '.' {

            break

        }

        t[*n] = input

        *n++

    }

}

func cetakArray(t tabel, n int) {

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

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

    }

    fmt.Println()

}

func balikanArray(t *tabel, n int) {

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

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

    }

}

func palindrom(t tabel, n int) bool {

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

```

```
        if t[i] != t[n-i-1] {

            return false

        }

    }

    return true
}

func main() {

    var tab tabel

    var m int

    isiArray(&tab, &m)

    fmt.Print("Teks: ")

    cetakArray(tab, m)

    balikanArray(&tab, m)

    fmt.Print("Reverse teks: ")

    cetakArray(tab, m)

    if palindrom(tab, m) {

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

    } else {

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

    }

}
```

## Screenshoot program

```
PS D:\Praktikum\Alpro2\Modul7\Unguided> go run "d:\Praktikum\Alpro2\Modul7\Unguided\unguided4.go"
Masukkan teks (akhiri dengan titik): makam.
Teks: makam
Reverse teks: makam
Teks tersebut adalah palindrom.
PS D:\Praktikum\Alpro2\Modul7\Unguided> █
```

```
PS D:\Praktikum\Alpro2\Modul7\Unguided> go run "d:\Praktikum\Alpro2\Modul7\Unguided\unguided4.go"
Masukkan teks (akhiri dengan titik): siang.
Teks: siang
Reverse teks: gnais
Teks tersebut bukan palindrom.
PS D:\Praktikum\Alpro2\Modul7\Unguided> █
```

## Deskripsi program

Program ini merupakan program dalam bahasa Go untuk memeriksa apakah suatu teks adalah palindrom. Berikut penjelasan singkat fungsi-fungsi yang ada:

1. isiArray: Mengisi array tabel dengan karakter yang diinputkan oleh pengguna hingga pengguna mengetikkan titik (.) atau jumlah karakter mencapai batas maksimum (NMAX).
2. cetakArray: Mencetak seluruh karakter dalam array tabel hingga indeks n.
3. balikanArray: Membalikkan urutan karakter dalam array tabel dengan cara menukar elemen awal dan akhir secara berpasangan hingga setengah panjang array.
4. palindrom: Memeriksa apakah array tabel adalah palindrom dengan membandingkan elemen dari kedua sisi array. Jika semua pasangan cocok, maka teks adalah palindrom (mengembalikan true); jika tidak, false.
5. main:
  - Mengisi array tab dengan teks yang diinputkan pengguna.
  - Mencetak teks asli.
  - Membalik dan mencetak teks yang telah dibalik.
  - Memeriksa apakah teks asli adalah palindrom dan menampilkan hasilnya.



Contoh hasil:

- Input: MALAM. → Output: "Teks tersebut adalah palindrom."
- Input: SIANG. → Output: "Teks tersebut bukan palindrom."