

**LAPORAN PRAKTIKUM**  
**ALGORITMA DAN PEMROGRAMAN 2**

**MODUL 7**  
**STRUCT & ARRAY**



**Disusun Oleh:**

**Muhammad Djulianoor / 2311102253**

**Kelas**

**IF-11-06**

**Dosen Pengampu:**

**Abednego Dwi Septiadi**

**PROGRAM STUDI S1 TEKNIK INFORMATIKA**  
**FAKULTAS INFORMATIKA**  
**TELKOM UNIVERSITY PURWOKERTO**

**2024**

## **I. DASAR TEORI**

### Dasar Teori

#### **A. Struct**

*Struct* merupakan kumpulan dari beberapa variabel yang beragam tipe datanya, kemudian disimpan menjadi satu variabel. *Struct* ialah tipe data yang berfungsi untuk menggabungkan objek-objek yang ada di bawah satu nama atau variabel. Programmer membuat *struct* dengan tujuan agar dapat dikelola dengan lebih terorganisir dan efisien. Dengan begitu, programmer atau para developer dapat mengembangkan sebuah program dengan cara yang mudah untuk dipahami.

#### **B. Array**

*Array* ialah kumpulan data yang memiliki tipe data yang sama. *Array* tidak dapat menyimpan kumpulan data yang memiliki tipe data yang beragam.

## II. GUIDED

### 1. Soal Studi Case

Suatu lingkaran didefinisikan dengan koordinat titik pusat (cx, cy) dengan radius r. Apabila diberikan dua buah lingkaran, maka tentukan posisi sebuah titik sembarang (x, y) berdasarkan dua lingkaran tersebut. Gunakan tipe bentukan titik untuk menyimpan koordinat, dan tipe bentukan lingkaran untuk menyimpan titik pusat lingkaran dan radiusnya.

Masukan terdiri dari beberapa tiga baris. Baris pertama dan kedua adalah koordinat titik pusat dan radius dari lingkaran 1 dan lingkaran 2, sedangkan baris ketiga adalah koordinat titik sembarang. Asumsi sumbu x dan y dari semua titik dan juga radius direpresentasikan dengan bilangan bulat.

Keluaran berupa string yang menyatakan posisi titik “Titik di dalam lingkaran 1 dan 2”, “Titik di dalam lingkaran 1”, “Titik di dalam lingkaran 2”, atau “Titik di luar lingkaran 1 dan 2”.

Fungsi untuk menghitung jarak titik (a, b) dan (c, d) di mana rumus jarak adalah:

$$jarak = \sqrt{(a - c)^2 + (b - d)^2}$$

dan juga fungsi untuk menentukan posisi sebuah titik sembarang berada di dalam suatu lingkaran atau tidak.

```
function jarak(p, q : titik) -> real
{Mengembalikan jarak antara titik p(x,y) dan titik q(x,y)}

function didalam(c:lingkaran, p:titik) -> boolean
{Mengembalikan true apabila titik p(x,y) berada di dalam lingkaran c yang
memiliki titik pusat (cx,cy) dan radius r}
```

## Source code

```
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 1 (cx1 cy1 r1): ")
    fmt.Scanln(&cx2, &cy2, &r2)
```

```

lingkaran2 := Lingkaran{pusat: Titik{x: cx1, y: cy1}, radius: r1}

var x, y int
fmt.Println("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 kedua lingkaran")
}
}

```

## Screenshoot Output

```

PS C:\Users\ASUS K3402Z\Documents\Collage Works\Mata Kuliah (Semester 3)\Praktikum Algoritma dan Pemrograman 2\Tugas Individu (Coding)\Modul 7> go run "c:\Users\ASUS K3402Z\Documents\Collage Works\Mata Kuliah (Semester 3)\Praktikum Algoritma dan Pemrograman 2\Tugas Individu (Coding)\Modul 7\Modul 7_Guided\Modul 7_Guided 1\main.go"
Masukkan koordinat pusat dan radius lingkaran 1 (cx1 cy1 r1): 4 3 5
Masukkan koordinat pusat dan radius lingkaran 1 (cx1 cy1 r1): 6 3 1
Masukkan koordinat titik sembarang (x y): 5 3
Titik di dalam lingkaran 1 dan 2
PS C:\Users\ASUS K3402Z\Documents\Collage Works\Mata Kuliah (Semester 3)\Praktikum Algoritma dan Pemrograman 2\Tugas Individu (Coding)\Modul 7>

```

## **Deskripsi Program**

### III. UNGUIDED

#### 1. Soal Studi Case

Sebuah array digunakan untuk menampung sekumpulan bilangan bulat. Buatlah program yang digunakan untuk mengisi array tersebut sebanyak N elemen nilai. Asumsikan array memiliki kapasitas penyimpanan data sejumlah elemen tertentu. Program dapat menampilkan beberapa informasi berikut:

- A. Menampilkan keseluruhan isi dari array.
- B. Menampilkan elemen-elemen array dengan indeks ganjil saja.
- C. Menampilkan elemen-elemen array dengan indeks genap saja (asumsi indek ke-0 adalah genap).
- D. Menampilkan elemen-elemen array dengan indeks kelipatan bilangan x. x bisa diperoleh dari masukan pengguna
- E. Menghapus elemen array pada indeks tertentu, asumsi indeks yang hapus selalu valid. Tampilkan keseluruhan isi dari arraynya, pastikan data yang dihapus tidak tampil.
- F. Menampilkan rata-rata dari bilangan yang ada di dalam array.
- G. Menampilkan standar deviasi atau simpangan baku dari bilangan yang ada di dalam array tersebut.
- H. Menampilkan frekuensi dari suatu bilangan tertentu di dalam array yang telah diisi tersebut.

#### Source code

```
package main

import (
    "fmt"
    "math"
)

func main() {
    var n, x, deleteIndex, targetNumber int
    fmt.Print("Masukkan jumlah elemen array (N): ")
    fmt.Scan(&n)

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

```

        fmt.Scan(&array[i])
    }

    for {
        fmt.Println("\nMenu:")
        fmt.Println("1. Tampilkan keseluruhan 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 X")
        fmt.Println("5. Hapus elemen pada indeks tertentu")
        fmt.Println("6. Tampilkan rata-rata elemen array")
        fmt.Println("7. Tampilkan standar deviasi elemen array")
        fmt.Println("8. Tampilkan frekuensi dari bilangan tertentu")
        fmt.Println("9. Keluar")

        var choice int
        fmt.Print("Pilih menu: ")
        fmt.Scan(&choice)

        switch choice {
        case 1:
            fmt.Println("Isi array:", array)
        case 2:
            fmt.Print("Elemen dengan indeks ganjil: ")
            for i := 1; i < len(array); i += 2 {
                fmt.Print(array[i], " ")
            }
            fmt.Println()
        case 3:
            fmt.Print("Elemen dengan indeks genap: ")
            for i := 0; i < len(array); i += 2 {
                fmt.Print(array[i], " ")
            }
            fmt.Println()
        case 4:
            fmt.Print("Masukkan nilai X: ")
            fmt.Scan(&x)
            fmt.Printf("Elemen dengan indeks kelipatan %d: ", x)
            for i := 0; i < len(array); i++ {
                if i%x == 0 {
                    fmt.Print(array[i], " ")
                }
            }
        }
    }
}

```



```

    }
}
fmt.Println()
case 5:
    fmt.Print("Masukkan indeks elemen yang akan dihapus: ")
    fmt.Scan(&deleteIndex)
    if deleteIndex >= 0 && deleteIndex < len(array) {
        array = append(array[:deleteIndex],
array[deleteIndex+1:]...)
        fmt.Println("Array setelah penghapusan:", array)
    } else {
        fmt.Println("Indeks tidak valid!")
    }
}
case 6:
    sum := 0
    for _, value := range array {
        sum += value
    }
    average := float64(sum) / float64(len(array))
    fmt.Printf("Rata-rata elemen array: %.2f\n", average)
case 7:
    sum := 0
    for _, value := range array {
        sum += value
    }
    average := float64(sum) / float64(len(array))
    variance := 0.0
    for _, value := range array {
        variance += math.Pow(float64(value)-average, 2)
    }
    stdDev := math.Sqrt(variance / float64(len(array)))
    fmt.Printf("Standar deviasi elemen array: %.2f\n", stdDev)
case 8:
    fmt.Print("Masukkan bilangan yang ingin dicari
frekuensinya: ")
    fmt.Scan(&targetNumber)
    frequency := 0
    for _, value := range array {
        if value == targetNumber {
            frequency++
        }
    }

```

```

    }
    fmt.Printf("Frekuensi bilangan %d: %d kali\n",
targetNumber, frequency)
    case 9:
        fmt.Println("Keluar dari program...")
        return
    default:
        fmt.Println("Pilihan tidak valid, silakan coba lagi!")
    }
}
}

```

## Screenshoot Output

```

PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS
> go run "c:\Users\ASUS_K3402Z\Documents\Collage Works\Mata Kuliah
(Semester 3)\Praktikum Algoritma dan Pemrograman 2\Tugas Individu (Coding)\Modul 7\Modul 7_Unguided\Modul 7_Unguided 1\main2.go"
Masukkan jumlah elemen array (N): 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

Menu:
1. Tampilkan keseluruhan isi array
2. Tampilkan elemen dengan indeks ganjil
3. Tampilkan elemen dengan indeks genap
4. Tampilkan elemen dengan indeks kelipatan X
5. Hapus elemen pada indeks tertentu
6. Tampilkan rata-rata elemen array
7. Tampilkan standar deviasi elemen array
8. Tampilkan frekuensi dari bilangan tertentu
9. Keluar
Pilih menu: 2
Elemen dengan indeks ganjil: 2 4

```

## Deskripsi Program

Program di atas merupakan program untuk menyimpan array, program ini menggunakan percabangan *switch case* untuk membuat menu pilihan untuk pengguna yang menjalankan program. Setelah array disimpan oleh program, pengguna dapat memilih pilihan pada menu pilihan yang sudah disediakan oleh program. Terdapat beberapa pilihan, yaitu 1). Tampilkan keseluruhan isi array, 2). Tampilkan elemen dengan indeks ganjil, 3). Tampilkan elemen dengan indeks genap, 4). Tampilkan elemen dengan indeks kelipatan x, 5). Hapus elemen pada indeks tertentu, 6). Tampilkan rata-rata elemen array, 7). Tampilkan standar deviasi elemen array, 8). Tampilkan frekuensi dari bilangan tertentu, 9). Keluar.

## 2. Soal Studi Case

Sebuah program digunakan untuk menyimpan dan menampilkan nama-nama klub yang memenangkan pertandingan bola pada suatu grup pertandingan. Buatlah program yang digunakan untuk merekap skor pertandingan bola 2 buah klub bola yang berlaga.

Pertama-tama program meminta masukan nama-nama club yang bertanding, kemudian program meminta masukan skor hasil pertandingan kedua klub tersebut. Yang disimpan dalam array adalah nama-nama klub yang menang saja.

Proses input skor berhenti ketika skor salah satu atau kedua klub tidak valid (negatif). Di akhir program, tampilkan daftar klub yang memenangkan pertandingan.

```
Klub A : MU
Klub B : Inter
Pertandingan 1 : 2   0           // MU = 2 sedangkan Inter = 0
Pertandingan 2 : 1   2
Pertandingan 3 : 2   2
Pertandingan 4 : 0   1
Pertandingan 5 : 3   2
Pertandingan 6 : 1   0
Pertandingan 7 : 5   2
Pertandingan 8 : 2   3
Pertandingan 9 : -1  2
Hasil 1 : MU
Hasil 2 : Inter
Hasil 3 : Draw
Hasil 4 : Inter
Hasil 5 : MU
Hasil 6 : MU
Hasil 7 : MU
Hasil 8 : Inter
Pertandingan selesai
```

## Source Code

```
package main

import (
    "fmt"
)

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

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

    fmt.Println("\nMasukkan skor pertandingan antara", klubA, "dan",
klubB)
    fmt.Println("Masukkan skor negatif untuk menghentikan input.")
    pertandingan := 1
    for {
        fmt.Printf("Pertandingan %d - Skor %s: ", pertandingan, klubA)
        fmt.Scan(&skorA)
        fmt.Printf("Pertandingan %d - Skor %s: ", pertandingan, klubB)
        fmt.Scan(&skorB)
        if skorA < 0 || skorB < 0 {
            fmt.Println("Pertandingan selesai.")
            break
        }
        if skorA > skorB {
            hasil = append(hasil, klubA)
            fmt.Printf("Hasil %d: %s menang\n", pertandingan, klubA)
        } else if skorA < skorB {
            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)
        }
        pertandingan++
    }
}
```

```

    }

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

```

## Screenshoot Output

```

PS C:\Users\ASUS K3402Z\Documents\Collage Works\Mata Kuliah (Semester 3)\Praktikum Algoritma dan Pemrograman 2\Tugas Individu (Coding)\Modul 7>
> go run "c:\Users\ASUS K3402Z\Documents\Collage Works\Mata Kuliah (Semester 3)\Praktikum Algoritma dan Pemrograman 2\Tugas Individu (Coding)\Modul 7\Modul 7_Unguided\Modul 7_Unguided 2\main3.go"
Klub A: Chelsea
Klub B: Arsenal

Masukkan skor pertandingan antara Chelsea dan Arsenal
Masukkan skor negatif untuk menghentikan input.
Pertandingan 1 - Skor Chelsea: 3
Pertandingan 1 - Skor Arsenal: 0
Hasil 1: Chelsea menang
Pertandingan 2 - Skor Chelsea: 2
Pertandingan 2 - Skor Arsenal: -1
Pertandingan selesai.

Hasil Pertandingan:
Hasil 1: Chelsea
PS C:\Users\ASUS K3402Z\Documents\Collage Works\Mata Kuliah (Semester 3)\Praktikum Algoritma dan Pemrograman 2\Tugas Individu (Coding)\Modul 7>

```

## Deskripsi Program

Pertandingan bola di atas menggunakan program untuk melihat klub mana yang memenangkan pertandingan. Program di atas menggunakan percabangan yang ada pada dalam perulangan. Perulangan berfungsi untuk menerima *input* dari pengguna, *input* tersebut adalah skor dari masing-masing klub. Kemudian terdapat percabangan untuk menentukan klub mana yang menang, kalah, atau malah kedua klub memiliki hasil yang sama alias seri.

### 3. Soal Studi Case

Sebuah array digunakan untuk menampung sekumpulan karakter, anda diminta untuk memuat sebuah subprogram untuk melakukan membalikkan urutan isi array dan memeriksa apakah membentuk palindrom.

Lengkapi potongan algoritma berikut ini!

```
package main
import "fmt"
const NMAX int = 127
type tabel [NMAX]rune
    tab : tabel
    m : integer

func isiArray(t *tabel, n *int)
/*I.S. Data tersedia dalam piranti masukan
  F.S. Array t berisi sejumlah n karakter yang dimasukkan user,
  Proses input selama karakter bukanlah TITIK dan n <= NMAX */
```

```
func cetakArray(t tabel, n int)
/*I.S. Terdefinisi array t yang berisi sejumlah n karakter
  F.S. n karakter dalam array muncul di layar */

func balikanArray(t *tabel, n int)
/*I.S. Terdefinisi array t yang berisi sejumlah n karakter
  F.S. Urutan isi array t terbalik */

func main(){
    var tab tabel
    var m int
    // si array tab dengan memanggil prosedur isiArray

    // Balikian isi array tab dengan memanggil balikanArray

    // Cetak is array tab
}
```

Perhatikan sesi interaksi pada contoh berikut ini

Teks	: <u>S</u> <u>E</u> <u>N</u> <u>A</u> <u>N</u> <u>G</u> .
Reverse teks	: G N A N E S
Teks	: <u>K</u> <u>A</u> <u>I</u> <u>A</u> <u>K</u> .
Reverse teks	: K A T A K

Modifikasi program tersebut dengan menambahkan fungsi palindrom. Tambahkan instruksi untuk memanggil fungsi tersebut dan menampilkan hasilnya pada program utama.

<pre>func palindrom(t tabel, n int) bool /* Mengembalikan true apabila susunan karakter di dalam t membentuk palindrom, dan false apabila sebaliknya. Petunjuk: Manfaatkan prosedur balikanArray */</pre>
---

## Source Code

```
package main

import (
    "fmt"
)

const NMAX int = 127

type tabel [NMAX]rune

func isiArray(t *tabel, n *int) {
    var char rune
    *n = 0
    fmt.Println("Masukkan karakter ke dalam array (akhiri dengan karakter '.'): ")

    for {
        fmt.Scanf("%c\n", &char)
        if char == '.' || *n >= NMAX {
            break
        }
        t[*n] = char
        *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-1-i] = t[n-1-i], t[i]
    }
}

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 m int

    isiArray(&tab, &m)

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

    balikanArray(&tab, m)

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

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

```

## Screenshoot Output



```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS
PS C:\Users\ASUS K3402Z\Documents\Collage Works\Mata Kuliah (Semester 3)\Praktikum Algoritma dan Pemrograman 2\Tugas Individu (Coding)\Modul 7>
                                                                    > go run "c:\Users\ASUS K3402Z\Documents\Collage Works\Mata Kuliah (Semester 3)\Praktikum Algoritma dan Pemrograman 2\Tugas Individu (Coding)\Modul 7\Modul 7_Unguided 3\main4.go"
Masukkan karakter ke dalam array (akhiri dengan karakter '.'):
K
A
T
A
K
.
Teks: KATAK
Reverse teks: KATAK
Palindrom: true
PS C:\Users\ASUS K3402Z\Documents\Collage Works\Mata Kuliah (Semester 3)\Praktikum Algoritma dan Pemrograman 2\Tugas Individu (Coding)\Modul 7> |
```

## Deskripsi Program

Program ini adalah program untuk mengetahui apakah kata yang dimasukkan adalah kata palindrom. Kata palindrom adalah kata yang jika kata itu dibalik, maka tetap akan menjadi kata itu sendiri. Program ini menggunakan array untuk menyimpan masing-masing karakter kata.