

**LAPORAN PRAKTIKUM  
ALGORITMA DAN PEMROGRAMAN 2**

**MODUL VII  
STRUCK & ARRAY**



**Disusun Oleh :  
PETRA PRIADI S.P GINTING (2311102273)  
IF-06**

**Dosen Pengampu :  
ABEDNEGO DWI SEPTIADI**

**PROGRAM STUDI S1 TEKNIK INFORMATIKA  
FAKULTAS INFORMATIKA  
TELKOM UNIVERSITY PURWOKERTO  
2024**

## I. DASAR TEORI

**Struct** adalah tipe data yang digunakan untuk mengelompokkan beberapa variabel dengan tipe data yang berbeda menjadi satu kesatuan. Struct sangat berguna untuk merepresentasikan objek yang memiliki berbagai atribut. Dalam Go, struct didefinisikan menggunakan kata kunci `type` diikuti dengan nama struct dan kata kunci `struct`. Struct memungkinkan kita untuk membuat tipe data yang lebih kompleks dengan menggabungkan beberapa tipe data dasar. Misalnya, jika kita ingin merepresentasikan data seorang pengguna, kita bisa membuat struct yang berisi nama, umur, dan alamat pengguna tersebut. Struct juga mendukung metode, yang memungkinkan kita untuk mendefinisikan fungsi yang terkait dengan tipe struct tertentu. Ini sangat berguna untuk mengorganisir kode dan membuatnya lebih modular dan mudah dipahami.

**Array** adalah kumpulan elemen yang memiliki tipe data yang sama dan diakses menggunakan indeks. Ukuran array harus ditentukan saat deklarasi dan tidak dapat diubah setelahnya. Array di Go memiliki sifat statis, artinya ukuran array ditentukan pada saat kompilasi dan tidak bisa diubah selama runtime. Array sangat berguna ketika kita perlu menyimpan sejumlah elemen dengan tipe data yang sama dan kita tahu jumlah elemen tersebut tidak akan berubah. Misalnya, kita bisa menggunakan array untuk menyimpan daftar nilai ujian siswa atau daftar nama hari dalam seminggu. Array di Go diakses menggunakan indeks yang dimulai dari 0. Kita bisa mengakses elemen array dengan menggunakan notasi indeks, misalnya `array[0]` untuk elemen pertama, `array[1]` untuk elemen kedua, dan seterusnya.

Perbedaan Struct dan Array

- **Struct:**
  - Dapat menyimpan berbagai tipe data yang berbeda.
  - Digunakan untuk merepresentasikan objek yang kompleks dengan berbagai atribut.
  - Mendukung metode yang memungkinkan kita untuk mendefinisikan fungsi yang terkait dengan tipe struct tertentu.
  - Cocok untuk mengelompokkan data yang berhubungan tetapi memiliki tipe data yang berbeda.
- **Array:**
  - Hanya dapat menyimpan elemen dengan tipe data yang sama.
  - Ukuran array harus ditentukan saat deklarasi dan tidak dapat diubah.
  - Digunakan untuk menyimpan kumpulan data yang sejenis.
  - Cocok untuk menyimpan data dalam jumlah tetap yang tidak akan berubah selama runtime.

Struct sangat berguna untuk mengelompokkan data yang beragam, sementara array sangat efisien untuk menyimpan data yang homogen dalam jumlah tetap.

## II. GUIDED

### 1. Sourcecode

```
package main

import (
    "fmt"
    "math"
)

// menyimpan titik dengan koordinat (x,y)
type titik struct {
    x int
    y int
}

// menyimpan lingkaran dengan pusat dan radius
type lingkaran struct {
    pusat titik
    radius int
}

// fungsi untuk menghitung jarak antara dua titik
func hitungjarak(a, b titik) float64 {
    return math.Sqrt(float64((a.x - b.x) + (a.y -
b.y)*(a.y-b.y)))
}

// fungsi untuk memeriksa apakah titik berada di dalam
lingkaran
func titikDiDalamLingkaran(t titik, l lingkaran) bool {
    jarak := hitungjarak(t, l.pusat)
    return jarak <= float64(l.radius)
}

func main() {
    // input untuk lingkaran 1
    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}

    //input untuk lingkaran 2
    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}

        //input untuk titik sembarang
        var x, y int
        fmt.Print("Masukkan koordinat titik sembarang(x y):
")
        fmt.Scanln(&x, &y)
        titik := titik{x: x, y: y}

        // pengecekan posisi titik
        diDalam1 := titikDiDalamLingkaran(titik, lingkaran1)
        diDalam2 := titikDiDalamLingkaran(titik, lingkaran2)

        //menampilkan hasil sesuai kondisi
        if diDalam1 && diDalam2 {
            fmt.Println("Titik di dalam lingkaran 1 dan
2")
        } else if diDalam1 {
            fmt.Println("Titik di dalam lingkaran 1")
        } else if diDalam2 {
            fmt.Println("Titik di dalam lingkaran 2")
        } else {
            fmt.Println("Titik berada di luar lingkaran 1
dan 2")
        }
    }
}

```

## Screenshoot Output

```

PS C:\Users\Lenovo> go run "d:\Pemrograman\Pemrograman Semester 3\2311102273_Petra Priadi S.P Ginting\MODUL 7\Guided1_modul7.go"
Masukkan koordinat pusat dan radius lingkaran 1 (cx1, cy1, r1): 1 1 5
Masukkan koordinat pusat dan radius lingkaran 2 (cx2, cy2, r2): 8 8 4
Masukkan koordinat titik sembarang(x y): 2 2
Titik di dalam lingkaran 1
PS C:\Users\Lenovo> go run "d:\Pemrograman\Pemrograman Semester 3\2311102273_Petra Priadi S.P Ginting\MODUL 7\Guided1_modul7.go"
Masukkan koordinat pusat dan radius lingkaran 1 (cx1, cy1, r1): 1 2 3
Masukkan koordinat pusat dan radius lingkaran 2 (cx2, cy2, r2): 4 5 6
Masukkan koordinat titik sembarang(x y): 7 8
Titik di dalam lingkaran 2
PS C:\Users\Lenovo> go run "d:\Pemrograman\Pemrograman Semester 3\2311102273_Petra Priadi S.P Ginting\MODUL 7\Guided1_modul7.go"
Masukkan koordinat pusat dan radius lingkaran 1 (cx1, cy1, r1): 5 10 15
Masukkan koordinat pusat dan radius lingkaran 2 (cx2, cy2, r2): -15 4 20
Masukkan koordinat titik sembarang(x y): 0 0
Titik di dalam lingkaran 1 dan 2
PS C:\Users\Lenovo> go run "d:\Pemrograman\Pemrograman Semester 3\2311102273_Petra Priadi S.P Ginting\MODUL 7\Guided1_modul7.go"
Masukkan koordinat pusat dan radius lingkaran 1 (cx1, cy1, r1): 1 1 5
Masukkan koordinat pusat dan radius lingkaran 2 (cx2, cy2, r2): 8 8 4
Masukkan koordinat titik sembarang(x y): 15 20
Titik berada di luar lingkaran 1 dan 2
PS C:\Users\Lenovo>

```

## **Deskripsi Program**

Program menggunakan dua struct utama: 'titik' untuk menyimpan koordinat (x,y) dan 'lingkaran' yang memiliki pusat (menggunakan struct titik) dan radius. Terdapat fungsi 'hitungjarak' yang menghitung jarak antara dua titik menggunakan rumus jarak Euclidean, serta fungsi 'titikDiDalamLingkaran' yang memeriksa apakah suatu titik berada di dalam lingkaran dengan membandingkan jarak titik ke pusat lingkaran dengan radius lingkaran. Dalam fungsi main, program meminta input dari pengguna berupa koordinat pusat dan radius untuk dua lingkaran (cx1,cy1,r1 dan cx2,cy2,r2) serta koordinat sebuah titik sembarang (x,y). Setelah mendapatkan input, program melakukan pengecekan posisi titik terhadap kedua lingkaran menggunakan fungsi titikDiDalamLingkaran dan menampilkan hasil sesuai kondisi: apakah titik berada di dalam kedua lingkaran, hanya di dalam lingkaran 1, hanya di dalam lingkaran 2, atau di luar kedua lingkaran. Program ini memanfaatkan package "fmt" untuk input/output dan "math" untuk perhitungan akar kuadrat dalam fungsi hitungjarak.

### 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:

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

#### Sourcecode

```
package main

import (
    "fmt"
    "math"
)

func main() {
    var p, pilihan int
    fmt.Print("Masukkan jumlah elemen array: ")
    fmt.Scan(&p)

    arr := make([]int, p)

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

    for {
        fmt.Println("\nMenu
(PetraGinting_2311102273):")
```

```

        fmt.Println("1. Tampilkan seluruh array")
        fmt.Println("2. Tampilkan elemen indeks
ganjil")
        fmt.Println("3. Tampilkan elemen indeks
genap")
        fmt.Println("4. Tampilkan elemen kelipatan X")
        fmt.Println("5. Hapus elemen pada indeks
tertentu")
        fmt.Println("6. Tampilkan rata-rata")
        fmt.Println("7. Tampilkan standar deviasi")
        fmt.Println("8. Tampilkan frekuensi bilangan")
        fmt.Println("9. Keluar")
        fmt.Print("Pilih menu : ")
        fmt.Scan(&pilihan)

        switch pilihan {
        case 1:
            tampilkanArray(arr)
        case 2:
            tampilkanIndeksGanjil(arr)
        case 3:
            tampilkanIndeksGenap(arr)
        case 4:
            var x int
            fmt.Print("Masukkan nilai X: ")
            fmt.Scan(&x)
            tampilkanKelipatanX(arr, x)
        case 5:
            var idx int
            fmt.Print("Masukkan indeks yang akan
dihapus: ")
            fmt.Scan(&idx)
            arr = hapusElemen(arr, idx)
        case 6:
            fmt.Printf("Rata-rata: %.2f\n",
hitungRataRata(arr))
        case 7:
            fmt.Printf("Standar deviasi: %.2f\n",
hitungStandarDeviasi(arr))
        case 8:
            var num int
            fmt.Print("Masukkan bilangan yang
dicari: ")
            fmt.Scan(&num)
            fmt.Printf("Frekuensi %d: %d\n", num,
hitungFrekuensi(arr, num))
        case 9:
            fmt.Println("Program selesai")
            return

```

```

        default:
            fmt.Println("Pilihan tidak valid")
        }
    }

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

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

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

func tampilkanKelipatanX(arr []int, x int) {
    fmt.Printf("Elemen kelipatan %d: ", x)
    for _, v := range arr {
        if v%x == 0 {
            fmt.Printf("%d ", v)
        }
    }
    fmt.Println()
}

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

func hitungRataRata(arr []int) float64 {
    sum := 0
    for _, v := range arr {
        sum += v
    }
}

```



```

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

    func hitungStandarDeviasi(arr []int) float64 {
        mean := hitungRataRata(arr)
        var sumSquares float64
        for _, v := range arr {
            diff := float64(v) - mean
            sumSquares += diff * diff
        }
        variance := sumSquares / float64(len(arr))
        return math.Sqrt(variance)
    }

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

```

## Screenshoot Output

```

PS C:\Users\Lenovo> go run "d:\Pemrograman\Pemrograman Semester 3\2311102273_Petra Priadi S.P Ginting\MODUL 7\Unguided1_modul7.go"
Masukkan jumlah elemen array: 5
Masukkan elemen ke-0: 1
Masukkan elemen ke-1: 2
Masukkan elemen ke-2: 3
Masukkan elemen ke-3: 4
Masukkan elemen ke-4: 5

Menu (PetraGinting_2311102273):
1. Tampilkan seluruh array
2. Tampilkan elemen indeks ganjil
3. Tampilkan elemen indeks genap
4. Tampilkan elemen kelipatan X
5. Hapus elemen pada indeks tertentu
6. Tampilkan rata-rata
7. Tampilkan standar deviasi
8. Tampilkan frekuensi bilangan
9. Keluar
Pilih menu : 1
Isi array: 1 2 3 4 5

Menu (PetraGinting_2311102273):
1. Tampilkan seluruh array
2. Tampilkan elemen indeks ganjil
3. Tampilkan elemen indeks genap
4. Tampilkan elemen kelipatan X
5. Hapus elemen pada indeks tertentu
6. Tampilkan rata-rata
7. Tampilkan standar deviasi
8. Tampilkan frekuensi bilangan
9. Keluar
Pilih menu : 2
Elemen indeks ganjil: 2 4

Menu (PetraGinting_2311102273):
1. Tampilkan seluruh array
2. Tampilkan elemen indeks ganjil
3. Tampilkan elemen indeks genap
4. Tampilkan elemen kelipatan X
5. Hapus elemen pada indeks tertentu
6. Tampilkan rata-rata
7. Tampilkan standar deviasi
8. Tampilkan frekuensi bilangan
9. Keluar
Pilih menu : 3
Elemen indeks genap: 1 3 5

```

```
Menu (PetraGinting_2311102273):
1. Tampilkan seluruh array
2. Tampilkan elemen indeks ganjil
3. Tampilkan elemen indeks genap
4. Tampilkan elemen kelipatan X
5. Hapus elemen pada indeks tertentu
6. Tampilkan rata-rata
7. Tampilkan standar deviasi
8. Tampilkan frekuensi bilangan
9. Keluar
Pilih menu : 4
Masukkan nilai X: 2
Elemen kelipatan 2: 2 4
```

```
Menu (PetraGinting_2311102273):
1. Tampilkan seluruh array
2. Tampilkan elemen indeks ganjil
3. Tampilkan elemen indeks genap
4. Tampilkan elemen kelipatan X
5. Hapus elemen pada indeks tertentu
6. Tampilkan rata-rata
7. Tampilkan standar deviasi
8. Tampilkan frekuensi bilangan
9. Keluar
Pilih menu : 5
Masukkan indeks yang akan dihapus: 4
```

```
Menu (PetraGinting_2311102273):
1. Tampilkan seluruh array
2. Tampilkan elemen indeks ganjil
3. Tampilkan elemen indeks genap
4. Tampilkan elemen kelipatan X
5. Hapus elemen pada indeks tertentu
6. Tampilkan rata-rata
7. Tampilkan standar deviasi
8. Tampilkan frekuensi bilangan
9. Keluar
Pilih menu : 1
Isi array: 1 2 3 4
```

```
Menu (PetraGinting_2311102273):
1. Tampilkan seluruh array
2. Tampilkan elemen indeks ganjil
3. Tampilkan elemen indeks genap
4. Tampilkan elemen kelipatan X
5. Hapus elemen pada indeks tertentu
6. Tampilkan rata-rata
7. Tampilkan standar deviasi
8. Tampilkan frekuensi bilangan
9. Keluar
Pilih menu : 6
Rata-rata: 2.50
```

```
Menu (PetraGinting_2311102273):
1. Tampilkan seluruh array
2. Tampilkan elemen indeks ganjil
3. Tampilkan elemen indeks genap
4. Tampilkan elemen kelipatan X
5. Hapus elemen pada indeks tertentu
6. Tampilkan rata-rata
7. Tampilkan standar deviasi
8. Tampilkan frekuensi bilangan
9. Keluar
Pilih menu : 7
Standar deviasi: 1.12
```

```
Menu (PetraGinting_2311102273):
1. Tampilkan seluruh array
2. Tampilkan elemen indeks ganjil
3. Tampilkan elemen indeks genap
4. Tampilkan elemen kelipatan X
5. Hapus elemen pada indeks tertentu
6. Tampilkan rata-rata
7. Tampilkan standar deviasi
8. Tampilkan frekuensi bilangan
9. Keluar
Pilih menu : 8
Masukkan bilangan yang dicari: 2
Frekuensi 2: 1
```

## **Deskripsi Program**

Program dimulai dengan meminta pengguna memasukkan jumlah elemen array dan nilai-nilainya, kemudian menyajikan menu interaktif dengan 9 pilihan operasi. Program menggunakan struktur switch-case untuk menangani pilihan menu dan memiliki beberapa fungsi utama: 'tampilkanArray' untuk menampilkan seluruh isi array, 'tampilkanIndeksGanjil' dan 'tampilkanIndeksGenap' untuk menampilkan elemen pada indeks ganjil atau genap, 'tampilkanKelipatanX' untuk menampilkan elemen yang merupakan kelipatan dari nilai X yang diinput, 'hapusElemen' untuk menghapus elemen pada indeks tertentu menggunakan teknik slice, 'hitungRataRata' untuk menghitung nilai rata-rata dari seluruh elemen array, 'hitungStandarDeviasi' untuk menghitung standar deviasi menggunakan rumus statistik (melibatkan package math untuk fungsi akar kuadrat), dan 'hitungFrekuensi' untuk menghitung berapa kali suatu bilangan muncul dalam array. Program menggunakan loop tak terbatas untuk terus menampilkan menu sampai pengguna memilih opsi 9 untuk keluar, dan setiap fungsi menggunakan teknik slice Go untuk manipulasi array secara efisien. Program ini mendemonstrasikan penggunaan berbagai fitur Go seperti slices, loops, switch statements, functions, dan perhitungan matematis, serta menerapkan konsep modularitas dengan memisahkan setiap operasi ke dalam fungsi tersendiri.

## 2. 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 klub 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.

### Sourcecode

```
package main

import "fmt"

func main() {

    var klub1, klub2 string

    var totalhasilPertandingan []string

    fmt.Print("Nama Klub A : ")

    fmt.Scan(&klub1)

    fmt.Print("Nama Klub B : ")

    fmt.Scan(&klub2)

    var skor1, skor2 int

    pertandinganKe := 1

    for {

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

        fmt.Scan(&skor1, &skor2)
```

```
        if skor1 < 0 || skor2 < 0 {

            break

        }

        if skor1 > skor2 {

            totalhasilPertandingan =
append(totalhasilPertandingan, klub1)

        } else if skor2 > skor1 {

            totalhasilPertandingan =
append(totalhasilPertandingan, klub2)

        } else {

            totalhasilPertandingan =
append(totalhasilPertandingan, "Draw")

        }

        pertandinganKe++

    }

    fmt.Println("\nHasil pertandingan Akhir:")
    for i, hasil := range totalhasilPertandingan {

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

    }

    fmt.Println("Pertandingan selesai")

}
```

## Screenshot Output

```
PS C:\Users\Lenovo> go run "d:\Pemrograman\Pemrograman Semester 3\2311102273_Petra Priadi S.P Ginting\MODUL 7\Unguided2_modul7.go"
Nama Klub A : RM
Nama Klub B : BC
Pertandingan 1 : 12 1
Pertandingan 2 : 11 1
Pertandingan 3 : 4 4
Pertandingan 4 : 3 2
Pertandingan 5 : 1 0
Pertandingan 6 : 3 3
Pertandingan 7 : 1 0
Pertandingan 8 : 1 0
Pertandingan 9 : -1 1

Hasil pertandingan Akhir:
Hasil 1 : RM
Hasil 2 : RM
Hasil 3 : Draw
Hasil 4 : RM
Hasil 5 : RM
Hasil 6 : Draw
Hasil 7 : RM
Hasil 8 : RM
Pertandingan selesai
PS C:\Users\Lenovo>
```

## Deskripsi Program

Program dimulai dengan meminta input nama dua klub (klub1 dan klub2) dari pengguna, kemudian menggunakan slice string 'totalhasilPertandingan' untuk menyimpan hasil setiap pertandingan. Program menggunakan loop tak terbatas (infinite loop) untuk terus meminta skor pertandingan (skor1 dan skor2) dari kedua tim, dimana pertandingan akan terus berlanjut sampai pengguna memasukkan skor negatif yang berfungsi sebagai kondisi untuk mengakhiri input. Dalam setiap iterasi, program membandingkan skor kedua tim: jika skor1 lebih besar dari skor2 maka klub1 dinyatakan sebagai pemenang, jika skor2 lebih besar dari skor1 maka klub2 dinyatakan sebagai pemenang, dan jika skornya sama maka hasilnya adalah "Draw". Hasil setiap pertandingan ditambahkan ke dalam slice totalhasilPertandingan menggunakan fungsi append. Setelah input selesai (ketika dimasukkan skor negatif), program menampilkan seluruh hasil pertandingan menggunakan loop range yang mengiterasi melalui slice totalhasilPertandingan, menampilkan nomor pertandingan (dimulai dari 1) dan hasil masing-masing pertandingan, dan diakhiri dengan pesan "Pertandingan selesai".

### 3. Studi Case

Sebuah array digunakan untuk menampung sekumpulan karakter, Anda diminta untuk membuat 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 bergaris bawah adalah input/read)

```
Teks      : S E N A N G .
Reverse teks : G N A N E S

Teks      : K A T A K .
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.

**\*Palindrom adalah teks yang dibaca dari awal atau akhir adalah sama, contoh: KATAK, APA, KASUR\_RUSAK.**

```
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 */
```

Perhatikan sesi interaksi pada contoh berikut ini (teks bergaris bawah adalah Input/read)

```
Teks      : K A T A K
Palindrom  : ? true

Teks      : S E N A N G
Palindrom  : ? false
```

## Sourcecode

```
package main

import "fmt"

const NMAX int = 127

type tabel [NMAX]rune

// Fungsi untuk mengecek palindrom
func isPalindrom(t tabel, n int) bool {
    for i := 0; i < n/2; i++ {
        if t[i] != t[n-1-i] {
            return false
        }
    }
    return true
}

func isiArray(t *tabel, n *int) {
    var char rune
    *n = 0
    fmt.Print("Masukkan karakter (akhiri dengan titik):")

    for {
        _, err := fmt.Scanf("%c", &char)
        if err != nil || char == '.' || *n >= NMAX {
            break
        }

        // Mengabaikan karakter newline dan spasi
        if char == '\n' || char == '\r' || char == ' ' {
            continue
        }

        t[*n] = char
        *n++
    }
}

func cetakArray(t tabel, n int) {
    fmt.Print("Isi array: ")
    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 main() {
    var tab tabel
    var n int

    isiArray(&tab, &n)

    isPalindrome := isPalindrom(tab, n)

    fmt.Println("\nArray sebelum dibalik:")
    cetakArray(tab, n)

    if isPalindrome {
        fmt.Println("Kata yang dimasukkan adalah
palindrom")
    } else {
        fmt.Println("Kata yang dimasukkan bukan
palindrom")
    }

    balikanArray(&tab, n)

    fmt.Println("\nArray setelah dibalik:")
    cetakArray(tab, n)
}

```

## Screenshot Output

```

PS C:\Users\Lenovo> go run "d:\Pemrograman\Pemrograman Semester 3\2311102273_Petra Priadi S.P Ginting\MODUL 7\Unguided3_modul7.go"
Masukkan karakter (akhiri dengan titik): apa.

Array sebelum dibalik:
Isi array: apa
Kata yang dimasukkan adalah palindrom

Array setelah dibalik:
Isi array: apa
PS C:\Users\Lenovo> go run "d:\Pemrograman\Pemrograman Semester 3\2311102273_Petra Priadi S.P Ginting\MODUL 7\Unguided3_modul7.go"
Masukkan karakter (akhiri dengan titik): angsa.

Array sebelum dibalik:
Isi array: angsa
Kata yang dimasukkan bukan palindrom

Array setelah dibalik:
Isi array: asgna
PS C:\Users\Lenovo>

```

### **Deskripsi Program**

Program menggunakan array bertipe rune dengan konstanta NMAX sebesar 127 karakter sebagai batas maksimum. Program memiliki beberapa fungsi utama: 'isPalindrom' yang mengecek apakah array karakter adalah palindrom dengan membandingkan karakter dari depan dan belakang, 'isiArray' yang menerima input karakter dari pengguna hingga ditemukan titik (.) atau mencapai batas NMAX sambil mengabaikan karakter newline dan spasi, 'cetakArray' untuk menampilkan isi array karakter, dan 'balikanArray' yang membalik urutan karakter dalam array. Dalam fungsi main, program pertama memanggil isiArray untuk mendapatkan input dari pengguna, kemudian memeriksa apakah input tersebut palindrom menggunakan isPalindrom, menampilkan array asli menggunakan cetakArray, memberitahu pengguna apakah input adalah palindrom atau bukan, membalik array menggunakan balikanArray, dan akhirnya menampilkan hasil array yang sudah dibalik. Program ini memanfaatkan pointer untuk memodifikasi array dan jumlah elemen secara langsung dalam fungsi, serta menggunakan tipe data rune untuk menangani karakter.