

**LAPORAN PRAKTIKUM
ALGORITMA DAN PEMROGRAMAN 2**

MODUL 7

STRUCK DAN ARRAY



Disusun Oleh : Brian Farrel Evandhika

2311102037

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

7.1 Tipe Bentuk

Tipe bentuk memungkinkan pemrograman untuk mendefinisikan suatu tipe data baru pada suatu

bahasa pemrograman. Tipe bentuk ini dapat dibedakan atas dua jenis, yaitu Alias dan Struct.

1) Alias (Type)

Bahasa pemrograman pada umumnya mengizinkan pemrograman untuk mengubah nama

suatu tipe data dengan nama baru yang lebih ringkas dan familiar. Sebagai contoh "Integer"

dapat diubah dengan nama alias "bilangan" Caranya dengan menggunakan kata kunci

Sebagai contoh perhatikan program Go berikut beserta hasil eksekusinya!

```
package main

import "fmt"

type bilangan int

type pecahan float64

func main() {

    var a, b bilangan

    var hasil pecahan

    a = 9

    b = 5

    hasil = pecahan(a) / pecahan(b)

    fmt.Println(hasil)

}
```

2)

Stucture memungkinkan pemrograman untuk mengelompokkan beberapa data atau nilai yang

memiliki relasi atau keterkaitan tertentu menjadi suatu kesatuan. Masing-masing nilai tersimpan dalam field dari stucture tersebut.

Berbeda dengan bahasa pemrograman lain. kesamaan tipe dari dua variabel berjenis Stucture

bukan karena namanya tetapi karena strukturnya. Dua variabel dengan nama-nama field dan

tipe field yang sama (dan dalam urutan yang Sama) dianggap mempunyai tipe yang sama.

Tentunya akan lebih memudahkan Jika stucture tersebut didefinisikan sebagai sebuah tipe baru, sehingga deklarasi Stucture tidak perlu lagi seluruh field-nya ditulis ulang berkali-kali.

```
package main
```

```
import "fmt"
```

```
type waktu struct {
```

```
    jam, menit, detik int
```

```
}
```

```
func main() {
```

```
    var wParkir, wPulang, durasi waktu
```

```
    var dParkir, dPulang, lParkir int
```

```
    fmt.Scan(&wParkir.jam, &wParkir.menit, &wParkir.detik)
```

```
    fmt.Scan(&wPulang.jam, &wPulang.menit, &wPulang.detik)
```

```
    dParkir = wParkir.detik + wParkir.menit*60 + wParkir.jam*3600
```

```
dPulang = wPulang.detik + wPulang.menit*60 + wPulang.jam*3600
```

```
lParkir = dPulang - dParkir
```

```
durasi.jam = lParkir / 3600
```

```
durasi.menit = lParkir % 3600 / 60
```

```
durasi.detik = lParkir % 3600 % 60
```

```
fmt.Printf("Lama parkir: %d jam %d menit %d detik",
```

```
    durasi.jam, durasi.menit, durasi.detik)
```

```
}
```

7.2 Array

```
var (
```

```
    // array arr mempunyai 73 elemen, masing-masing bertipe CircType2
```

```
    arr [73]CircType2
```

```
    // array buf dengan 5 elemen, dengan nilai awal 7, 3, 5, 2, dan 11.
```

```
    buf [5]byte = {7, 3, 5, 2, 11}
```

```
    // mhs adalah array dengan 2000 elemen bertipe NewType
```

```
    mhs [2000]NewType
```

```
    // rec adalah array dari array, yaitu matriks, atau array berdimensi-2
```

```
    rec [20][40]float64
```

```
)
```

Jumlah elemen array dapat diminta dengan fungsi `len` yang tersedia. Sebagai contoh `len(arr)` akan menghasilkan 73 untuk contoh di atas.

Indeks array dimulai dari 0, sehingga indeks pada contoh adalah 0, 1, ..., len(arr)-1

```
// Mengganti isi elemen ke-8 dengan nilai dari elemen ke-7
```

```
arr[8] = arr[7]
```

```
// Mengambil data field x dari elemen ke-1
```

```
currX = arr[1].center.x
```

```
// Mengambil elemen terakhir
```

```
n := len(arr)
```

```
buf = arr[n-1]
```

Slice (Array dinamik)

Array dalam Go juga dapat mempunyai ukuran yang dinamik. (Tidak digunakan di kelas Algoritma

Pemrograman). Deklarasinya mirip dengan deklarasi array, tetapi jumlah elemennya dikosongkan.

```
1 // declaring chop as an empty slice of float64
```

```
2 var chop []float64
```

```
3
```

```
4 // declaring s101 as a slice
```

```
5 var s101 []int = {11, 2, 3, 5, 7, 13}
```

Sebuah slice dapat dialokasikan menggunakan fungsi built-in make

```
1 // Prealokasi 10 elemen untuk s102 dan sejumlah tempat tambahan
```

```
2 var s102 []int = make([]int, 10, 20)
```

```
3
```

```
4 // Prealokasi 7 elemen untuk s103 tanpa tempat tambahan
```

```
5 var s103 []circType = make([]circType, 7)
```

Fungsi built-in len dapat digunakan untuk mengetahui ukuran slice. Fungsi lain, cap, dapat digunakan

untuk mengetahui total tempat yang disediakan untuk slice tersebut,

```
1 // Cetak jumlah elemen dan tempat yang tersedia untuk s102
```

```
2 fmt.Println( len(s102), cap(s102) )
```

append dapat digunakan untuk menambahkan elemen ke suatu slice, dan bila perlu

Fungsi built-in

memperbesar tempat untuk slice tersebut.

```
// Append elemen baru, membuat slice baru, dan menyimpan kembali slice baru
```

```
// ke variabel semula. Boleh juga disimpan ke variabel lain, sehingga variabel
```

```
// semula masih menyimpan slice yang asli. */
```

```
s101 = append(s101, 17)
```

```
s101 = append(s101, 19, 23)
```

// Sebuah slice baru juga dapat terbentuk dengan mengambil slice dari suatu array atau slice yang lain.

```
s104 = arr[:4]
```

```
// Ambil beberapa elemen terakhir, dimulai dari indeks 5
```

```

s105 = s101[5:]

// Salin semua dari slice/array aslinya

s106 = s105[:]

// Salin element dari indeks 3 sampai, tapi tidak termasuk, 5.

// Jadi dalam contoh hanya 2 elemen s106[3] dan s106[4] yang disalin

s107 = s106[3:5]


// Map


// Tipe array lain, sebuah array dinamik. Indeksnya (di sini disebut kunci) tidak
// harus berbentuk integer.

// Indeks dapat berasal dari tipe data saja. Struktur ini disebut map.


var dct map[string]int

// Deklarasi map variabel dct sebagai bilangan bulat dengan kunci string

var dct1 map[string]string // Deklarasi map lain dct1 dengan kunci juga string

dct1 = map[string]string{"john": "hi", "anne": "darling"}

// Mempunyai nilai awal dct1["john"] = "hi", dct1["anne"] = "darling"

var dct2 map[float64]int = make(map[float64]int, 10)

// Deklarasi dan prealokasi tempat untuk map dct2


// Mengambil nilai yang tersimpan dengan kunci "john"

fmt.Println(dct1["john"])

```

```
// Mengganti nilai yang tersimpan pada kunci "anne", dan
```

```
// Membuat entri baru dengan kunci "boy"
```

```
dct1["anne"] = "lovely"
```

```
dct1["boy"] = "runaround"
```

```
// Menghapus entri dengan kunci "john"
```

```
delete(dct1, "john")
```


GUIDED

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, dan benarkan ungkapan "Titik pusat 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. berupa String yang menyatakan posisi titik di I 2", di dalam ungkapan 1" atau 1 dm.

Sourcecode

```
//2311102037_BRIAN FARREL EVANDHIKA_IF 11 06

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 titikdalamlingkaran(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 kordinat pusat dan radius lingkaran 1 (cx1
cy1 r1): ")
    fmt.Scanln(&cx1, &cy1, &r1)

    // Membuat objek lingkaran1
    lingkaran1 := lingkaran{pusat: titik{x: cx1, y: cy1}, radius:
r1}

    var cx2, cy2, r2 int
    fmt.Print("Masukkan kordinat pusat dan radius lingkaran 12 (cx2
cy2 r2): ")
    fmt.Scanln(&cx2, &cy2, &r2)

    // Membuat objek lingkaran2
    lingkaran2 := lingkaran{pusat: titik{x: cx2, y: cy2}, radius:
r2}

    var x, y int
    fmt.Print("Masukkan kordinat titik t (x y): ")
    fmt.Scanln(&x, &y)

    // Membuat objek titik yang akan diperiksa
    titik2 := titik{x: x, y: y}

    // Memeriksa apakah titik berada di dalam lingkaran1 dan
lingkaran2
    didalam1 := titikdalamlingkaran(titik2, lingkaran1)
    didalam2 := titikdalamlingkaran(titik2, lingkaran2)

    if didalam1 && didalam2 {
        fmt.Println("Titik berada di dalam kedua lingkaran.")
    } else if didalam1 {
        fmt.Println("Titik berada di dalam lingkaran 1 saja.")
    } else if didalam2 {

```

```

        fmt.Println("Titik berada di dalam lingkaran 2 saja.")
    } else {
        fmt.Println("Titik berada di luar kedua lingkaran.")
    }
}

```

Screenshot Program

```

-Module PSReadLine'.

PS C:\Users\MSI GAMING> go run "c:\Users\MSI GAMING\Documents\TELKOM UNIVERS
ITY\SEMESTER 3\PRAKTIKUM ALGORITMA DAN PEMROGRAMAN 2\Pertemuan Ke 7\Guided-1
.go"
Masukkan kordinat pusat dan radius lingkaran 1 (cx1 cy1 r1): 1 2 3
Masukkan kordinat pusat dan radius lingkaran 12 (cx2 cy2 r2): 1 2 3
Masukkan kordinat titik t (x y): 1 2
Titik berada di dalam kedua lingkaran.
PS C:\Users\MSI GAMING>

```

Deskripsi Program

Program ini bertujuan untuk menentukan posisi sebuah titik sembarang (x, y) relatif terhadap dua buah lingkaran yang didefinisikan oleh koordinat titik pusat (cx, cy) dan radius r masing-masing. Program membaca input yang terdiri dari tiga baris: baris pertama dan kedua berisi koordinat titik pusat dan radius dari dua lingkaran, sedangkan baris ketiga berisi koordinat titik sembarang. Dengan asumsi bahwa semua koordinat dan radius direpresentasikan sebagai bilangan bulat, program akan mengevaluasi apakah titik (x, y) berada di dalam lingkaran pertama, di dalam lingkaran kedua, di dalam kedua lingkaran, atau di luar keduanya. Hasil akhirnya adalah sebuah string yang menyatakan posisi titik tersebut: "di dalam Lingkaran 1", "di dalam Lingkaran 2", "di dalam kedua lingkaran", atau "di luar kedua lingkaran".

II. UNGUIDED

Soal Studi Case 1

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 indeks kea 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 pastikan data yang dihapus tidak tampil

f. Menampilkan rata-rata dari bilangan yang ada di 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

```
//2311102037_BRIAN FARREL EVANDHIKA_IF 11 06

package main

import (
    "fmt"
    "math"
)

// Fungsi untuk menghitung rata-rata array
func calculateAverage(arr []int) float64 {
```

```

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

// Fungsi untuk menghitung standar deviasi
func calculateStdDev(arr []int, mean float64) float64 {
    sum := 0.0
    for _, val := range arr {
        sum += math.Pow(float64(val)-mean, 2)
    }
    return math.Sqrt(sum / float64(len(arr)))
}

// Fungsi untuk menghitung frekuensi bilangan tertentu dalam array
func calculateFrequency(arr []int, target int) int {
    count := 0
    for _, val := range arr {
        if val == target {
            count++
        }
    }
    return count
}

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

    // Inisialisasi array
    arr := make([]int, n)

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

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

```

```

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

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

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

// e. Menghapus elemen array pada indeks tertentu
var index int
fmt.Print("\nMasukkan indeks yang ingin dihapus: ")
fmt.Scan(&index)
if index >= 0 && index < len(arr) {
    arr = append(arr[:index], arr[index+1:]...)
    fmt.Println("Array setelah penghapusan:")
    fmt.Println(arr)
} else {
    fmt.Println("Indeks tidak valid!")
}

// f. Menampilkan rata-rata dari bilangan dalam array
average := calculateAverage(arr)
fmt.Printf("\nRata-rata bilangan dalam array: %.2f\n", average)

// g. Menampilkan standar deviasi dari bilangan dalam array

```

```

    stdDev := calculateStdDev(arr, average)
    fmt.Printf("Standar deviasi bilangan dalam array: %.2f\n",
stdDev)

    // h. Menampilkan frekuensi dari suatu bilangan tertentu
    var target int
    fmt.Print("\nMasukkan bilangan yang ingin dihitung frekuensinya:
")
    fmt.Scan(&target)
    frequency := calculateFrequency(arr, target)
    fmt.Printf("Frekuensi bilangan %d dalam array: %d\n", target,
frequency)
}

```

Screenshot Program

```

-Module PSReadLine'.

PS C:\Users\MSI GAMING> go run "c:\Users\MSI GAMING\Documents\TELKOM UNIVERS
ITY\SEMESTER 3\PRAKTIKUM ALGORITMA DAN PEMROGRAMAN 2\Pertemuan Ke 7\Unguided
-1.go"
Masukkan jumlah elemen array (N): 10
Masukkan elemen array:
Elemen ke-0: 0
Elemen ke-1: 1
Elemen ke-2: 2
Elemen ke-3: 3
Elemen ke-4: 4
Elemen ke-5: 5
Elemen ke-6: 6
Elemen ke-7: 7
Elemen ke-8: 8
Elemen ke-9: 9

Keseluruhan isi array:
[0 1 2 3 4 5 6 7 8 9]

Elemen array dengan indeks ganjil:
1 3 5 7 9

Elemen array dengan indeks genap:
0 2 4 6 8

Masukkan bilangan x untuk kelipatan indeks: 3
Elemen array dengan indeks kelipatan 3 :
0 3 6 9

Masukkan indeks yang ingin dihapus: 9
Array setelah penghapusan:
[0 1 2 3 4 5 6 7 8]

Rata-rata bilangan dalam array: 4.00
Standar deviasi bilangan dalam array: 2.58

Masukkan bilangan yang ingin dihitung frekuensinya: 1
Frekuensi bilangan 1 dalam array: 1
PS C:\Users\MSI GAMING>

```

Deskripsi Program

Program di atas adalah implementasi dalam bahasa Go yang mengelola sebuah array bilangan bulat dengan berbagai fungsi. Program ini mengisi array dengan NN elemen yang dimasukkan oleh pengguna dan menyediakan beberapa fitur. Fitur-fitur tersebut meliputi: menampilkan seluruh isi array, menampilkan elemen dengan indeks ganjil dan genap, serta elemen pada indeks kelipatan suatu bilangan xx yang dimasukkan pengguna. Program juga memungkinkan penghapusan elemen pada indeks tertentu, serta menghitung dan menampilkan rata-rata, standar deviasi, dan frekuensi kemunculan suatu bilangan tertentu dalam array. Dengan demikian, program ini memberikan cara komprehensif untuk mengelola dan menganalisis data dalam array.

Soal Studi Case 2

Sebuah program digunakan untuk menyirnpkan dan menampilkan nama•nama klub yang

memenangkan pertandingan bola pada 5 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 masuhan shor hasil pertandingan hedua tersebut- Yang disimpan dalam

array adalah nama•nama Klub yang menang

Proses input Skor berhenti hetika Shor salah satu atau kedua klub tidak valid (negati0. Di akhir

program, tampilkan daftar Hlub yang memenanghan pertandingan.

Perhatikan sesi interaksi pada contoh borikut ini bawah

Source Code


```
//2311102037_BRIAN FARREL EVANDHIKA_IF 11 06

package main

import (
    "bufio"
    "fmt"
    "os"
    "strings"
)

func main() {
    reader := bufio.NewReader(os.Stdin)

    // Menyimpan nama klub yang menang
    var winners []string

    fmt.Print("Masukkan nama Klub A: ")
    clubA, _ := reader.ReadString('\n')
    clubA = strings.TrimSpace(clubA)

    fmt.Print("Masukkan nama Klub B: ")
    clubB, _ := reader.ReadString('\n')
    clubB = strings.TrimSpace(clubB)

    // Loop untuk menerima skor pertandingan
    for {
        var scoreA, scoreB int

        fmt.Print("\nMasukkan skor Klub A (atau negatif untuk
berhenti): ")
        fmt.Scan(&scoreA)
        if scoreA < 0 {
            break
        }

        fmt.Print("Masukkan skor Klub B (atau negatif untuk
berhenti): ")
        fmt.Scan(&scoreB)
        if scoreB < 0 {
            break
        }

        // Menentukan pemenang
        if scoreA > scoreB {
```

```
        winners = append(winners, clubA)
        fmt.Printf("Hasil: %s menang!\n", clubA)
    } else if scoreB > scoreA {
        winners = append(winners, clubB)
        fmt.Printf("Hasil: %s menang!\n", clubB)
    } else {
        fmt.Println("Hasil: Seri (Draw). Tidak ada pemenang!")
    }
}

// Menampilkan daftar pemenang
fmt.Println("\nDaftar klub yang memenangkan pertandingan:")
for i, winner := range winners {
    fmt.Printf("Pertandingan %d: %s\n", i+1, winner)
}

fmt.Println("Program selesai.")
}
```

Screenshot Program

```
PS C:\Users\MSI GAMING> go run "c:\Users\MSI GAMING\Documents\TELKOM UNIVERS
ITY\SEMESTER 3\PRAKTIKUM ALGORITMA DAN PEMROGRAMAN 2\Pertemuan Ke 7\tempCode
RunnerFile.go"
Masukkan nama Klub A: MU
Masukkan nama Klub B: Inter

Masukkan skor Klub A (atau negatif untuk berhenti): 2 0
Masukkan skor Klub B (atau negatif untuk berhenti): Hasil: MU menang!

Masukkan skor Klub A (atau negatif untuk berhenti): 1 2
Masukkan skor Klub B (atau negatif untuk berhenti): Hasil: Inter menang!

Masukkan skor Klub A (atau negatif untuk berhenti): 2 2
Masukkan skor Klub B (atau negatif untuk berhenti): Hasil: Seri (Draw). Tida
k ada pemenang!

Masukkan skor Klub A (atau negatif untuk berhenti): 0 1
Masukkan skor Klub B (atau negatif untuk berhenti): Hasil: Inter menang!

Masukkan skor Klub A (atau negatif untuk berhenti): 3 2
Masukkan skor Klub B (atau negatif untuk berhenti): Hasil: MU menang!

Masukkan skor Klub A (atau negatif untuk berhenti): 1 0
Masukkan skor Klub B (atau negatif untuk berhenti): Hasil: MU menang!

Masukkan skor Klub A (atau negatif untuk berhenti): 5 2
Masukkan skor Klub B (atau negatif untuk berhenti): Hasil: MU menang!

Masukkan skor Klub A (atau negatif untuk berhenti): 2 3
Masukkan skor Klub B (atau negatif untuk berhenti): Hasil: Inter menang!

Masukkan skor Klub A (atau negatif untuk berhenti): -1 2

Daftar klub yang memenangkan pertandingan:
Pertandingan 1: MU
Pertandingan 2: Inter
Pertandingan 3: Inter
Pertandingan 4: MU
Pertandingan 5: MU
Pertandingan 6: MU
Pertandingan 7: Inter
Program selesai.
PS C:\Users\MSI GAMING> █
```

Deskripsi Program

Program ini adalah implementasi dalam bahasa Go untuk menerima nama dua klub sepak bola (Klub A dan Klub B) dan mencatat skor pertandingan mereka. Program akan terus menerima input skor dari **pengguna sampai**

skor negatif dimasukkan, menandakan berhenti. Setelah setiap input skor, program menentukan dan mencatat pemenang berdasarkan skor yang lebih tinggi, atau menyatakan hasil seri jika skor sama. Setelah input selesai, program menampilkan daftar klub yang memenangkan setiap pertandingan yang telah dicatat. Dengan demikian, program ini memfasilitasi pencatatan dan penentuan pemenang dari beberapa pertandingan antara dua klub yang sama.

Soal Studi Case 3

Sebuah array digunakan untuk menampung sekumpulan karakter, Anda diminta untuk

membuat sebuah subprogram untuk melakukan membalikkan urutan isi array dan memeriksa

membentuk palindrom-

Lengkapi potongan algoritma berikut ini!

Source Code

```
//2311102037_BRIAN FARREL EVANDHIKA_IF 11 06

package main

import (
    "fmt"
)

const NMAX int = 127

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

// Fungsi untuk mengisi array
func isiArray(t *tabel, n *int) {
    var input rune
    *n = 0
    fmt.Println("Masukkan karakter (akhiri dengan '.'): ")
    for {
```

```

        fmt.Scanf("%c\n", &input)
        if input == '.' || *n >= NMAX {
            break
        }
        t.tab[*n] = input
        *n++
    }
    t.m = *n
}

// Fungsi untuk mencetak array
func cetakArray(t tabel, n int) {
    for i := 0; i < n; i++ {
        fmt.Printf("%c", t.tab[i])
    }
    fmt.Println()
}

// Fungsi untuk membalik isi array
func balikanArray(t *tabel, n int) {
    for i := 0; i < n/2; i++ {
        t.tab[i], t.tab[n-1-i] = t.tab[n-1-i], t.tab[i]
    }
}

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

func main() {
    var tab tabel
    var n int

    // Isi array
    isiArray(&tab, &n)

    // Cetak array
    fmt.Print("Teks: ")
    cetakArray(tab, n)
}

```

```

// Periksa apakah array adalah palindrom
if palindrom(tab, n) {
    fmt.Println("Palindrom: true")
} else {
    fmt.Println("Palindrom: false")
}
}

```

Screenshot Program

```

PS C:\Users\MSI GAMING> go run "c:\Users\MSI GAMING\Documents\TELKOM UNIVERS
ITY\SEMESTER 3\PRAKTIKUM ALGORITMA DAN PEMROGRAMAN 2\Pertemuan Ke 7\Unguided
-3.go"
Masukkan karakter (akhiri dengan '.'):
K
A
T
A
K
.
Teks: KATAK
Palindrom: true
PS C:\Users\MSI GAMING> go run "c:\Users\MSI GAMING\Documents\TELKOM UNIVERS
ITY\SEMESTER 3\PRAKTIKUM ALGORITMA DAN PEMROGRAMAN 2\Pertemuan Ke 7\Unguided
-3.go"
Masukkan karakter (akhiri dengan '.'):
S
E
N
A
N
G
.
Teks: SENANG
Palindrom: false
PS C:\Users\MSI GAMING>

```

Deskripsi Program

Program ini adalah implementasi dalam bahasa Go untuk memanipulasi array karakter dan memeriksa apakah array tersebut adalah palindrom. Program dimulai dengan mengisi array dengan karakter yang dimasukkan oleh pengguna sampai

titik ('.') atau batas maksimum tercapai. Setelah array diisi, program akan mencetak isi array. Kemudian, program menyediakan fungsi untuk membalik isi array dan memeriksa apakah array tersebut adalah palindrom, yaitu apakah urutan karakter dari awal ke akhir sama dengan urutan dari akhir ke awal. Terakhir, program mencetak hasil pemeriksaan tersebut dengan menyatakan apakah array adalah palindrom atau tidak.