

**LAPORAN PRAKTIKUM
ALGORITMA DAN PEMROGRAMAN II**

**MODUL VI
“STUCK & ARRAY”**



Oleh:

ZAHRINA ANTIKA MALAHATI

2311102109

IF 11 02

**S1 TEKNIK INFORMATIKA
TELKOM UNIVERSITY PURWOKERTO**

2024

I. DASAR TEORI

- Tipe bentukan memungkinkan pemrogram untuk mendefinisikan tipe data baru. Tipe ini dibedakan menjadi dua jenis, yaitu Alias (Type) dan Struct. Alias memungkinkan penggantian nama tipe data dengan nama yang lebih ringkas, sedangkan Struct mengelompokkan beberapa data yang memiliki relasi menjadi satu kesatuan. Bahasa pemrograman pada umumnya mengizinkan pemrograman untuk mengubah nama suatu tipe data dengan nama baru yang lebih ringkas dan familiar. Sebagai contoh "integer" dapat dirubah dengan nama alias "bllangan ". Caranya dengan menggunakan kata kunci "type".

```
package main

type <nama alias> <tipe data>

func main(){
...
}
```

- **Array:** Array adalah struktur data yang memiliki ukuran tetap selama eksekusi program. Elemen-elemen dalam array dideklarasikan dengan tipe data tertentu dan jumlah elemen menjadi bagian dari deklarasi variable. Ada beberapa tipe array, yaitu :
 - Slice (Array Dinamik)
Array dalam Go juga dapat mempunyai ukuran yang dinamik. (Tidak digunakan di kelas Algoritma Pemrograman).
 - Map
Tipe array lain, sebuah array dinamik. Indeksnya (di sini disebut kunci) tidak harus berbentuk integer. Indeks dapat berasal dari tipe apa saja. Struktur ini disebut map.

II. GUIDED

❖ 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)
```

```

        i+1, m.Nama, m.RataRata, m.Matematika, m.Fisika,
        m.Kimia)
    }
}

```

Output

```

PS C:\Users\HP> go run "c:\PRAKALPRO2\modul7\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 C:\Users\HP>

```

Deskripsi

Program adalah program dalam bahasa Go untuk mengelola data mahasiswa beserta nilai-nilai mereka dalam tiga mata pelajaran: Matematika, Fisika, dan Kimia. Setiap mahasiswa memiliki nama serta nilai dalam ketiga mata pelajaran tersebut. Program ini kemudian menghitung nilai rata-rata untuk setiap mahasiswa dan mengurutkan mereka berdasarkan nilai rata-rata tersebut dari yang tertinggi ke terendah. Hasil akhirnya adalah daftar peringkat mahasiswa berdasarkan rata-rata nilai yang dicetak pada layar.

❖ 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)
    }
}

```

Output

```

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 C:\Users\HP>

```

Deskripsi

Program adalah program dalam bahasa Go untuk mengelola data mahasiswa menggunakan struktur map. map digunakan untuk menyimpan pasangan data NIM (Nomor Induk Mahasiswa) sebagai kunci dan Nama mahasiswa sebagai nilai. Program ini mencakup beberapa operasi utama, seperti menambahkan data mahasiswa baru, menampilkan seluruh data mahasiswa, mengakses data berdasarkan NIM tertentu, serta menghapus data mahasiswa berdasarkan NIM. Hasil akhir ditampilkan dalam format tabel dengan kolom dan baris yang rapi.

III. UNGUIDED

❖ Unguided 1

Soal

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".

Contoh

No	Masukan	Keluaran
1	1 1 5 8 8 4 2 2	Titik di dalam lingkaran 1
2	1 2 3 4 5 6 7 8	Titik di dalam lingkaran 2
3	5 10 15 -15 4 20 0 0	Titik di dalam lingkaran 1 dan 2
4	1 1 5 8 8 4 15 20	Titik di luar lingkaran 1 dan 2

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

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

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}
```

Catatan: Lihat paket **math** dalam lampiran untuk menggunakan fungsi **math.Sqrt()** untuk menghitung akar kuadrat.

Source code

```
package main

import (
    "fmt"
    "math"
)

// Titik mepresentasikan sebuah titik dalam ruang 2D
type Titik struct {
    X int
    Y int
}

// Lingkaran mepresentasikan sebuah lingkaran dengan titik
pusat dan jari-jari
type Lingkaran struct {
    Pusat Titik
    Radius int
}

// Jarak menghitung anantara dua titik
func Jarak(p, q Titik) float64 {
    return math.Sqrt(math.Pow(float64(p.X-q.X), 2) +
math.Pow(float64(p.Y-q.Y), 2))
}

// Didalam memeriksa apakah sebuah titik berada dalam
lingkaran
func Didalam(c Lingkaran, p Titik) bool {
    return Jarak(c.Pusat, p) <= float64(c.Radius)
}

func main() {
    var l1, l2 Lingkaran
    var p Titik

    // Input untuk lingkaran 1
    fmt.Println("Masukkan koordinat titik pusat dan radius
lingkaran 1 (x y r):")
    fmt.Scanln(&l1.Pusat.X, &l1.Pusat.Y, &l1.Radius)

    // Input untuk lingkaran 2
    fmt.Println("Masukkan koordinat titik pusat dan radius
lingkaran 2 (x y r):")
    fmt.Scanln(&l2.Pusat.X, &l2.Pusat.Y, &l2.Radius)

    // Input untuk titik yang akan diperiksa
    fmt.Println("Masukkan koordinat titik (x y):")
    fmt.Scanln(&p.X, &p.Y)

    // Menentukan posisi titik terhadap lingkaran
    if Didalam(l1, p) && Didalam(l2, p) {
        fmt.Println("Titik di dalam lingkaran 1 dan 2")
    } else if Didalam(l1, p) {
```

```

        fmt.Println("Titik di dalam lingkaran 1")
    } else if Didalam(l2, p) {
        fmt.Println("Titik di dalam lingkaran 2")
    } else {
        fmt.Println("Titik di luar lingkaran 1 dan 2")
    }
}

// Zahrina Antika Malahati_2311102109

```

Output

```

PS C:\Users\HP> go run "c:\PRAKALPRO2\modul7\unguided1.go"
Masukkan koordinat titik pusat dan radius lingkaran 1 (x y r):
1 1 5
Masukkan koordinat titik pusat dan radius lingkaran 2 (x y r):
8 8 4
Masukkan koordinat titik (x y):
2 2
Titik di dalam lingkaran 1
PS C:\Users\HP> go run "c:\PRAKALPRO2\modul7\unguided1.go"
Masukkan koordinat titik pusat dan radius lingkaran 1 (x y r):
1 2 3
Masukkan koordinat titik pusat dan radius lingkaran 2 (x y r):
4 5 6
Masukkan koordinat titik (x y):
7 8
Titik di dalam lingkaran 2
PS C:\Users\HP> go run "c:\PRAKALPRO2\modul7\unguided1.go"
Masukkan koordinat titik pusat dan radius lingkaran 1 (x y r):
5 10 15
Masukkan koordinat titik pusat dan radius lingkaran 2 (x y r):
-15 4 20
Masukkan koordinat titik (x y):
0 0
Titik di dalam lingkaran 1 dan 2
PS C:\Users\HP> go run "c:\PRAKALPRO2\modul7\unguided1.go"
Masukkan koordinat titik pusat dan radius lingkaran 1 (x y r):
1 1 5
Masukkan koordinat titik pusat dan radius lingkaran 2 (x y r):
8 8 4
Masukkan koordinat titik (x y):
15 20
Titik di luar lingkaran 1 dan 2
PS C:\Users\HP>

```

Deskripsi

Program ini adalah program dalam bahasa Go untuk menentukan apakah suatu titik berada di dalam, di luar, atau tepat pada tepi dua lingkaran yang ditentukan. Setiap lingkaran memiliki titik pusat dan jari-jari. Program meminta user untuk memasukkan koordinat titik pusat dan radius dari dua lingkaran, serta koordinat dari titik yang akan diperiksa. Dengan menghitung jarak antara titik dan pusat lingkaran menggunakan rumus jarak Euclidean, program ini kemudian memeriksa apakah titik tersebut berada di dalam, pada tepi, atau di luar masing-masing lingkaran.

❖ Unguided 2

Soal

- 2) 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 indeks 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.

Source code

```
package main

import (
    "fmt"
    "math"
)

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

    // Membuat array dengan ukuran N
    arr := make([]int, N)

    // Meminta pengguna untuk mengisi elemen array
    fmt.Println("Masukkan elemen array:")
    for i := 0; i < N; i++ {
        fmt.Scan(&arr[i])
    }

    // Menampilkan keseluruhan isi array
    fmt.Println("Seluruh isi array:")
    fmt.Println(arr)
```

```

// Menampilkan elemen dengan indeks ganjil
fmt.Println("Elemen dengan indeks ganjil:")
for i := 1; i < N; i += 2 {
    fmt.Print(arr[i], " ")
}
fmt.Println()

// Menampilkan elemen dengan indeks genap
fmt.Println("Elemen dengan indeks genap:")
for i := 0; i < N; i += 2 {
    fmt.Print(arr[i], " ")
}
fmt.Println()

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

// Menghapus elemen pada indeks tertentu
var index int
fmt.Print("Masukkan indeks yang ingin dihapus: ")
fmt.Scan(&index)
arr = append(arr[:index], arr[index+1:]...)
fmt.Println("Array setelah penghapusan:")
fmt.Println(arr)

// Menghitung rata-rata
var sum int
for _, num := range arr {
    sum += num
}
rataRata := float64(sum) / float64(len(arr))
fmt.Printf("Rata-rata: %.2f\n", rataRata)

// Menghitung standar deviasi
var variance float64
for _, num := range arr {
    variance += math.Pow(float64(num)-rataRata, 2)
}
variance /= float64(len(arr))
standarDeviasi := math.Sqrt(variance)
fmt.Printf("Standar deviasi: %.2f\n",
standarDeviasi)

// Menghitung frekuensi suatu bilangan
var bilangan int

```

```

        fmt.Print("Masukkan bilangan yang ingin dicari
frekuensinya: ")
        fmt.Scan(&bilangan)
        var frekuensi int
        for _, num := range arr {
            if num == bilangan {
                frekuensi++
            }
        }
        fmt.Printf("Frekuensi   %d:   %d\n",   bilangan,
frekuensi)
    }

// Zahrina Antika Malahati_2311102109

```

Output

```

PS C:\Users\HP> go run "c:\PRAKALPRO2\modul7\unguided2.go"
Masukkan jumlah elemen array: 9
Masukkan elemen array:
0 1 2 3 4 5 6 7 8
Seluruh isi array:
[0 1 2 3 4 5 6 7 8]
Elemen dengan indeks ganjil:
1 3 5 7
Elemen dengan indeks genap:
0 2 4 6 8
Masukkan nilai x: 2
Elemen dengan indeks kelipatan 2:
0 2 4 6 8
Masukkan indeks yang ingin dihapus: 3
Array setelah penghapusan:
[0 1 2 4 5 6 7 8]
Rata-rata: 4.12
Standar deviasi: 2.71
Masukkan bilangan yang ingin dicari frekuensinya: 4
Frekuensi 4: 1
PS C:\Users\HP>

```

Deskripsi

Program ini adalah program dalam bahasa pemrograman Go dan dirancang untuk melakukan berbagai operasi pada sebuah array yang diinputkan oleh user. Program ini akan meminta user untuk memasukkan sejumlah elemen, kemudian melakukan serangkaian operasi, termasuk menampilkan elemen berdasarkan indeks, menghapus elemen, menghitung rata-rata, menghitung standar deviasi, dan mencari frekuensi suatu bilangan dalam array.

❖ Unguided 3

Soal

- 3) 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 berliga.

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.

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

```
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.Println("Masukkan nama klub A:")
```

```
fmt.Scanln(&klubA)
fmt.Println("Masukkan nama klub B:")
fmt.Scanln(&klubB)

for {
    fmt.Println("Masukkan skor klub A:")
    fmt.Scanln(&skorA)
    fmt.Println("Masukkan skor klub B:")
    fmt.Scanln(&skorB)

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

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

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

// Zahrina Antika Malahati_2311102109
```

Output

```

PS C:\Users\HP> go run "c:\PRAKALPRO2\modul7\unguided3.go"
Masukkan nama klub A:
MU
Masukkan nama klub B:
Inter
Masukkan skor klub A:
2
Masukkan skor klub B:
0
Masukkan skor klub A:
1
Masukkan skor klub B:
2
Masukkan skor klub A:
2
Masukkan skor klub B:
2
Masukkan skor klub A:
0
Masukkan skor klub B:
1
Masukkan skor klub A:
3
Masukkan skor klub B:
2
Masukkan skor klub A:
1
Masukkan skor klub B:
0
Masukkan skor klub A:
5
Masukkan skor klub B:
2
Masukkan skor klub A:
2
Masukkan skor klub B:
3
Masukkan skor klub A:
-1
Masukkan skor klub B:
2

Hasil pertandingan:
Pertandingan 1: MU
Pertandingan 2: Inter
Pertandingan 3: Draw
Pertandingan 4: Inter
Pertandingan 5: MU
Pertandingan 6: MU
Pertandingan 7: MU
Pertandingan 8: Inter
PS C:\Users\HP> 

```

Deskripsi

Program ini adalah program dalam bahasa Go untuk mencatat dan menampilkan hasil pertandingan antara dua klub sepak bola. User diminta untuk memasukkan nama dari kedua klub, lalu program akan terus meminta skor dari setiap pertandingan sampai skor negatif dimasukkan, yang akan menghentikan proses input. Setelah itu, program akan menentukan pemenang dari setiap pertandingan berdasarkan skor yang diberikan. Hasil dari setiap pertandingan akan disimpan dalam sebuah slice dan ditampilkan di akhir program.

❖ Unguided 4

Soal

-) 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 */
```

man 67 | Modul Praktikum Algoritma dan Pemrograman 2

```
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 I 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 I A K
Palindrom : ? true

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

Source code

```
package main

import "fmt"
```

```

const NMAX int = 127

type tabel [NMAX]rune

func isiArray(t *tabel, n *int) {
    // Implementasi untuk mengisi array t dengan input
    dari pengguna
    fmt.Print("Masukkan teks: ")
    var input string
    fmt.Scanln(&input)
    *n = len(input)
    for i := 0; i < *n; i++ {
        (*t)[i] = rune(input[i])
    }
}

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

func balikArray(t *tabel, n int) {
    // Implementasi untuk membalikkan isi array t
    for i, j := 0, n-1; i < j; i, j = i+1, j-1 {
        t[i], t[j] = t[j], t[i]
    }
}

func palindrom(t tabel, n int) bool {
    // Implementasi untuk mengecek apakah t adalah
    palindrom
    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.Println("Reverse teks: ")
    balikArray(&tab, m)
    cetakArray(tab, m)

    fmt.Println("Palindrom:", palindrom(tab, m))
}

```



```
// Zahrina Antika Malahati_2311102109
```

Output

```
PS C:\Users\HP> go run "c:\PRAKALPRO2\modul7\unguided4.go"
Masukkan teks: KATAK
Reverse teks:
Teks: KATAK
Palindrom: true
PS C:\Users\HP> go run "c:\PRAKALPRO2\modul7\unguided4.go"
Masukkan teks: SENANG
Reverse teks:
Teks: GNANES
Palindrom: false
PS C:\Users\HP> 
```

Deskripsi

Program ini adalah dalam bahasa Go untuk mengolah teks yang dimasukkan oleh user. Program ini memungkinkan user untuk memasukkan teks, kemudian menyimpan teks tersebut dalam array bertipe rune yang dapat menangani karakter Unicode. Program memiliki beberapa fungsi: untuk mengisi array dengan input dari user, mencetak isi array, membalikkan isi array, dan memeriksa apakah teks tersebut adalah palindrom (teks yang dibaca sama dari depan dan belakang). Setelah user memasukkan teks, program akan membalikkan teks dan mencetak hasilnya, serta memeriksa dan memberi tahu apakah teks tersebut palindrom.