

**LAPORAN PRAKTIKUM
PEMROGRAMAN BERORIENTASI OBJEK**

**MODUL VII
“STRUCK & ARRAY”**



Oleh:

MUHAMMAD RAGIEL PRASTYO

2311102183

S1IF-11-02

**S1 TEKNIK 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 dirubah dengan nama alias "bilangan". Caranya dengan menggunakan kata kunci "type".

	Notasi Algoritma	Notasi dalam bahasa Go
1	kamus	package main
2	type <nama alias> <tipe data>	
3		type <nama alias> <tipe data>
4	algoritma	
5	...	func main(){
6		...
7		}

2) Struct atau Record

Structure memungkinkan pemrograman untuk mengelompokkan beberapa data atau nilai yang memiliki relasi atau keterkaitan tertentu menjadi suatu kesatuan. Masing-masing nilai tersimpan dalam field dari structure tersebut.

	Notasi Algoritma	Notasi dalam bahasa Go
1	kamus	package main
2	type <nama struct> <	type <nama struct> struct {
3	<field 1> <tipe data>	<field 1> <tipe data>
4	<field 2> <tipe data>	<field 2> <tipe data>
5	<field 3> <tipe data>	<field 3> <tipe data>
6	>	}
7		

Berbeda dengan bahasa pemrograman lain, kesamaan tipe dari dua variabel sejenis structure 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 structure tersebut didefinisikan sebagai sebuah tipe

baru, sehingga deklarasi stucture tidak perlu lagi seluruh field-nya ditulis ulang berkali-kali.

7.2 Array

Array mempunyai ukuran (jumlah elemen) yang tetap (statls) selama eksekusi program, sehingga jumlah elemen array menjadi bagian dari deklarasi variabel dengan tipe array.

	Notasi dalam bahasa Go
1	var (
2	// array arr mempunyai 73 elemen, masing-masing bertipe CircType2
3	arr [73]CircType
4	
5	// array buf dengan 5 elemen, dengan nilai awal 7, 3, 5, 2, dan 11.
6	buf = [5]byte{7, 3, 5, 2, 11}
7	
8	// mhs adalah array dengan 2000 elemen bertipe NewType
9	mhs [2000]NewType
10	
11	// rec adalah array dari array, yaitu matriks, atau array berdimensi-2
12	rec [20][40]float64
13)

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

- 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.

- Sebuah slice dapat diprealokasi menggunakan fungsi built-in make
- Fungsi built-in len dapat digunakan untuk mengetahui ukuran slice. Fungsi lain, cap dapat digunakan untuk mengetahui total tempat yang disediakan untuk slice tersebut.
- Fungsi built-in append dapat digunakan untuk menambahkan elemen ke suatu slice, dan bila perlu memperbesar tempat untuk slice tersebut.
- Sebuah slice baru juga dapat terbentuk dengan mengambil slice dari suatu array atau slice yang lain.

- **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

1. Source Code

```
// MUHAMMAD RAGIEL PRASTYO
// 2311102183
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
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 (descend)
    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)
}
}

```

Screenshot Output

```

PS C:\Users\USER\OneDrive\Desktop\Alpro 2> go run "c:\Users\USER\OneDrive\Desktop\Alpro 2\Muhammad Ragiel Prastyo_2311102183_Modul7\Guided\tempCodeRunnerFile.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\USER\OneDrive\Desktop\Alpro 2>

```

Penjelasan:

Program diatas menyimpan data mahasiswa, menghitung rata-rata nilai tiga mata pelajaran, mengurutkannya dari rata-rata tertinggi, lalu menampilkan peringkat. Struct 'Mahasiswa' menyimpan data mahasiswa, dan fungsi 'hitungRataRata' menghitung rata-rata nilai. Di 'main', data mahasiswa diinisialisasi, dihitung rata-ratanya, diurutkan, dan ditampilkan.

2. Source Code

```

// MUHAMMAD RAGIEL PRASTYO
// 2311102183
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)
}

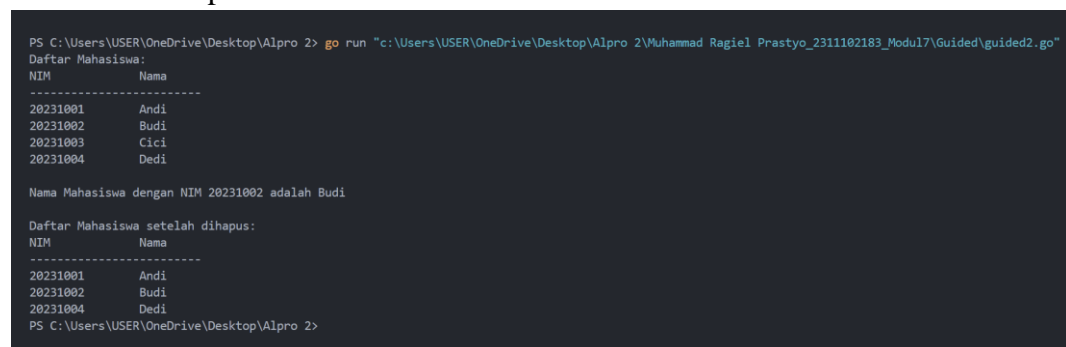
// Mengakses data berdasarkan NIM
nim := "20231002"
fmt.Println("\nNama Mahasiswa dengan NIM", nim, "adalah",
mahasiswa[nim])

// 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)
}
}

```

Screenshot Output



```

PS C:\Users\USER\OneDrive\Desktop\Alpro 2> go run "c:\Users\USER\OneDrive\Desktop\Alpro 2\Muhammad Ragiel Prastyo_2311102183_Modul7\Guided\guided2.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 C:\Users\USER\OneDrive\Desktop\Alpro 2>

```

Penjelasan:

Program diatas mengelola data mahasiswa menggunakan *map* dengan NIM sebagai kunci dan nama sebagai nilai. Setelah menginisialisasi *map* dengan data awal, program menambahkan satu mahasiswa baru, lalu menampilkan seluruh data dalam format tabel. Program juga mencari dan menampilkan nama

mahasiswa berdasarkan NIM tertentu (20231002). Setelah itu, program menghapus data mahasiswa dengan NIM 20231003 dan menampilkan kembali daftar mahasiswa yang tersisa.

III. UNGUIDED

1. Source Code

```
// MUHAMMAD RAGIEL PRASTYO
// 2311102183
package main
import (
    "fmt"
    "math"
)

type Point struct {
    x, y int
}

type Circle struct {
    center Point
    radius int
}

func distance(p1, p2 Point) float64 {
    return math.Sqrt(float64((p1.x-p2.x)*(p1.x-p2.x) + (p1.y-
p2.y)*(p1.y-p2.y)))
}

func isInside(circle Circle, point Point) bool {
    return distance(circle.center, point) <= float64(circle.radius)
}

func checkPointPosition(circle1, circle2 Circle, point Point) string {
    inCircle1 := isInside(circle1, point)
    inCircle2 := isInside(circle2, point)

    switch {
    case inCircle1 && inCircle2:
        return "Titik di dalam lingkaran 1 dan 2"
    case inCircle1:
        return "Titik di dalam lingkaran 1"
    case inCircle2:
        return "Titik di dalam lingkaran 2"
    default:
        return "Titik di luar lingkaran 1 dan 2"
    }
}
```

```

func main() {
    var cx1, cy1, r1 int
    var cx2, cy2, r2 int
    var px, py int

    fmt.Print("Masukkan koordinat pusat dan radius lingkaran 1 (cx1
cy1 r1): ")
    fmt.Scan(&cx1, &cy1, &r1)

    fmt.Print("Masukkan koordinat pusat dan radius lingkaran 2 (cx2
cy2 r2): ")
    fmt.Scan(&cx2, &cy2, &r2)


    fmt.Print("Masukkan koordinat titik sembarang (px py): ")
    fmt.Scan(&px, &py)

    circle1 := Circle{center: Point{x: cx1, y: cy1}, radius: r1}
    circle2 := Circle{center: Point{x: cx2, y: cy2}, radius: r2}
    point := Point{x: px, y: py}

    result := checkPointPosition(circle1, circle2, point)
    fmt.Println(result)
}

```

Screenshot Output



```

PS C:\Users\USER\OneDrive\Desktop\Alpro 2> go run "c:\Users\USER\OneDrive\Desktop\Alpro 2\Muhammad Ragiel Prastyo_2311102183_Modul7\Unguided\unguided1.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 (px py): 15 20
Titik di luar lingkaran 1 dan 2
PS C:\Users\USER\OneDrive\Desktop\Alpro 2>

```

Penjelasan:

Program diatas menentukan apakah suatu titik berada di dalam satu, kedua, atau di luar dua lingkaran. Struct ‘Point’ menyimpan koordinat titik, dan ‘Circle’ menyimpan pusat serta radius lingkaran. Fungsi ‘distance’ menghitung jarak antara dua titik, sementara ‘isInside’ mengecek apakah suatu titik berada dalam lingkaran. Fungsi ‘checkPointPosition’ mengembalikan posisi titik terhadap dua lingkaran. Di ‘main’, program meminta input untuk pusat dan radius kedua lingkaran, serta titik yang ingin diperiksa, lalu mencetak hasilnya.

2. Source Code

```
// MUHAMMAD RAGIEL PRASTYO
// 2311102183
package main
import (
    "fmt"
    "math"
)

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

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

    fmt.Println("a. Isi keseluruhan array:", array)

    fmt.Print("b. Elemen-elemen array dengan indeks ganjil: ")
    for i := 1; i < n; i += 2 {
        fmt.Print(array[i], " ")
    }
    fmt.Println()

    fmt.Print("c. Elemen-elemen array dengan indeks genap: ")
    for i := 0; i < n; i += 2 {
        fmt.Print(array[i], " ")
    }
    fmt.Println()

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

    var index int
```

```

fmt.Print("Masukkan indeks yang ingin dihapus: ")
fmt.Scan(&index)
if index >= 0 && index < n {
    array = append(array[:index], array[index+1:]...)
    fmt.Println("e. Isi array setelah penghapusan:", array)
} else {
    fmt.Println("Indeks tidak valid.")
}

sum := 0
for _, value := range array {
    sum += value
}
average := float64(sum) / float64(len(array))
fmt.Printf("f. Rata-rata dari bilangan dalam array: %.2f\n",
average)

varianceSum := 0.0
for _, value := range array {
    varianceSum += math.Pow(float64(value)-average, 2)
}
variance := varianceSum / float64(len(array))
stdDev := math.Sqrt(variance)
fmt.Printf("g. Standar deviasi dari bilangan dalam array: %.2f\n",
stdDev)

var target int
fmt.Print("Masukkan bilangan yang ingin dicari frekuensinya: ")
fmt.Scan(&target)
frequency := 0
for _, value := range array {
    if value == target {
        frequency++
    }
}
fmt.Printf("h. Frekuensi dari bilangan %d dalam array: %d\n",
target, frequency)
}

```

Screenshot Output

```
PS C:\Users\USER\OneDrive\Desktop\Alpro 2> go run "c:\Users\USER\OneDrive\Desktop\Alpro 2\Muhammad Ragiel Prastyo_2311102183_Modul7\Unguided\unguided2.go"
Masukkan jumlah elemen array: 9
Masukkan elemen-elemen array:
Elemen 0: 0
Elemen 1: 1
Elemen 2: 2
Elemen 3: 3
Elemen 4: 4
Elemen 5: 5
Elemen 6: 6
Elemen 7: 7
Elemen 8: 8
a. Isi keseluruhan array: [0 1 2 3 4 5 6 7 8]
b. Elemen-elemen array dengan indeks ganjil: 1 3 5 7
c. Elemen-elemen array dengan indeks genap: 0 2 4 6 8
Masukkan nilai x: 5
d. Elemen-elemen array dengan indeks kelipatan 5: 0 5
Masukkan indeks yang ingin dihapus: 7
e. Isi array setelah penghapusan: [0 1 2 3 4 5 6 8]
f. Rata-rata dari bilangan dalam array: 3.62
g. Standar deviasi dari bilangan dalam array: 2.50
Masukkan bilangan yang ingin dicari frekuensinya: 2
h. Frekuensi dari bilangan 2 dalam array: 1
PS C:\Users\USER\OneDrive\Desktop\Alpro 2> █
```

Penjelasan:

Program diatas meminta pengguna memasukkan jumlah elemen dalam array, lalu menginput setiap elemen satu per satu. Program menampilkan elemen-elemen array secara keseluruhan, dengan indeks ganjil, indeks genap, dan pada indeks yang merupakan kelipatan nilai x yang dimasukkan pengguna.

User juga dapat menghapus elemen pada indeks tertentu dan program menampilkan array setelah penghapusan. Program kemudian menghitung dan menampilkan rata-rata dan standar deviasi elemen-elemen dalam array, serta dapat mencari frekuensi kemunculan suatu bilangan dalam array.

3. Source Code

```
// MUHAMMAD RAGIEL PRASTYO
// 2311102183
package main
import (
    "fmt"
)

func main() {
    var clubA, clubB string
    var scoreA, scoreB int

    fmt.Print("Klub A: ")
    fmt.Scanln(&clubA)
    fmt.Print("Klub B: ")
    fmt.Scanln(&clubB)

    var winners []string
```

```

for i := 1; ; i++ {
    fmt.Printf("Pertandingan %d: ", i)
    fmt.Scanln(&scoreA, &scoreB)

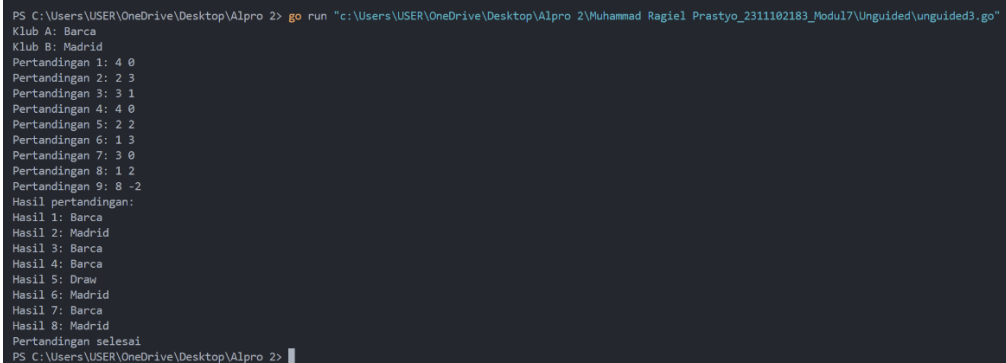
    if scoreA < 0 || scoreB < 0 {
        break
    }

    if scoreA > scoreB {
        winners = append(winners, clubA)
    } else if scoreB > scoreA {
        winners = append(winners, clubB)
    } else {
        winners = append(winners, "Draw")
    }
}

fmt.Println("Hasil pertandingan:")
for i, winner := range winners {
    fmt.Printf("Hasil %d: %s\n", i+1, winner)
}
fmt.Println("Pertandingan selesai")
}

```

Screenshot Output



```

PS C:\Users\USER\OneDrive\Desktop\Alpro 2> go run "c:\Users\USER\OneDrive\Desktop\Alpro 2\Muhammad Ragiel Prastyo_2311182183_Modul7\Unguided\unguided3.go"
Klub A: Barca
Klub B: Madrid
Pertandingan 1: 4 0
Pertandingan 2: 2 3
Pertandingan 3: 3 1
Pertandingan 4: 4 0
Pertandingan 5: 2 2
Pertandingan 6: 1 3
Pertandingan 7: 3 0
Pertandingan 8: 1 2
Pertandingan 9: 8 -2
Hasil pertandingan:
Hasil 1: Barca
Hasil 2: Madrid
Hasil 3: Barca
Hasil 4: Barca
Hasil 5: Draw
Hasil 6: Madrid
Hasil 7: Barca
Hasil 8: Madrid
Pertandingan selesai
PS C:\Users\USER\OneDrive\Desktop\Alpro 2>

```

Penjelasan:

Program diatas mencatat hasil pertandingan antara dua klub. User memasukkan nama klub A dan klub B, lalu memasukkan skor tiap pertandingan. Program menyimpan pemenang setiap pertandingan dalam winners atau draw jika skor seri. Input berhenti jika skor negatif dimasukkan.

Kemudian program akan menampilkan hasil semua pertandingan dan mengakhiri dengan "Pertandingan selesai".

4. Source Code

```
// MUHAMMAD RAGIEL PRASTYO
// 2311102183
package main
import (
    "fmt"
)

const NMAX int = 127

type tabel [NMAX]rune

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
    */
    var input string
    fmt.Print("Masukkan teks (akhiri dengan titik '.'): ")
    fmt.Scan(&input)

    for i, char := range input {
        if i >= NMAX || char == '.' {
            break
        }
        t[i] = char
        *n++
    }
}

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 */
    for i := 0; i < n; i++ {
        fmt.Printf("%c", t[i])
    }
    fmt.Println()
}

func balikanArray(t *tabel, n int) {
    /* I.S. Terdefinisi array t yang berisi sejumlah n karakter
```

```

    F.S. Urutan isi array t terbalik */
    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 {
    /* Mengembalikan true apabila susunan karakter di dalam t
    membentuk palindrom,
    dan false apabila sebaliknya. Petunjuk: Manfaatkan prosedur
    balikanArray */
    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)

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

    balikanArray(&tab, m)

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

```


Screenshot Output

```
PS C:\Users\USER\OneDrive\Desktop\Alpro 2> go run "c:\Users\USER\OneDrive\Desktop\Alpro 2\Muhammad Ragiel Prastyo_2311102183_Modul7\Unguided\unguided4.go"
Masukkan teks (akhiri dengan titik '.'): KATAK.
Teks: KATAK
Palindrom? true
Reverse teks: KATAK
PS C:\Users\USER\OneDrive\Desktop\Alpro 2>

PS C:\Users\USER\OneDrive\Desktop\Alpro 2> go run "c:\Users\USER\OneDrive\Desktop\Alpro 2\Muhammad Ragiel Prastyo_2311102183_Modul7\Unguided\unguided4.go"
Masukkan teks (akhiri dengan titik '.'): RAGIEL.
Teks: RAGIEL
Palindrom? false
Reverse teks: LEIGAR
PS C:\Users\USER\OneDrive\Desktop\Alpro 2>
```

Penjelasan:

Program diatas meminta input teks hingga karakter titik (.), menyimpannya dalam array tabel dan menampilkannya. Program memeriksa apakah teks adalah palindrom dengan membandingkan karakter dari ujung ke ujung, lalu menampilkan hasilnya (true atau false). Setelah itu, program membalik urutan karakter dalam array dan menampilkan teks yang telah dibalik.