

**LAPORAN PRAKTIKUM
ALGORITMA PEMROGRAMAN 2**

**MODUL 7
STRUCK & ARRAY**



Oleh:

PANDIA ARYA BRATA

2311102076

IF – 11 - 02

**S1 TEKNIK INFORMATIKA
INSTITUT TEKNOLOGI TELKOM PURWOKERTO**

2024

MODUL 7. STRUCK & ARRAY

7.1 Tipe Bentukkan

Tipe bentukkan memungkinkan pemrograman untuk mendefinisikan suatu tipe data baru pada suatu bahasa pemrograman. Tipe bentukkan 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		}

Sebagai contoh perhatikan program Go berikut beserta hasil eksekusinya!

```
1 package main
2 import "fmt"
3 type bilangan int
4 type pecahan float64
5 func main(){
6     var a,b bilangan
7     var hasil pecahan
8     a = 9
9     b = 5
10    hasil = pecahan(a) / pecahan(b)
11    fmt.Println(hasil)
12 }
```

```
E:\DEV\GO>go build Demo.go
```

```
E:\DEV\GO> Demo.exe
```

```
1.8
```

2) Struct atau Record

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.

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

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.

```
1 package main
2 import "fmt"
3 type waktu struct {
4     jam, menit, detik int
5 }
6
7
8 func main(){
9     var wParkir, wPulang, durasi waktu
10    var dParkir, dPulang, lParkir int
11    fmt.Scan(&wParkir.jam, &wParkir.menit, &wParkir.detik)
12    fmt.Scan(&wPulang.jam, &wPulang.menit, &wPulang.detik)
13    dParkir = wParkir.detik + wParkir.menit*60 + wParkir.jam*3600
14    dPulang = wPulang.detik + wPulang.menit*60 + wPulang.jam*3600
15    lParkir = dPulang - dParkir
16    durasi.jam = lParkir / 3600
17    durasi.menit = lParkir % 3600 / 60
18    durasi.detik = lParkir % 3600 % 60
19    fmt.Printf("Lama parkir: %d jam %d menit %d detik",
20        durasi.jam, durasi.menit, durasi.detik)
21 }
E:\DEV\GO>go build Demo.go
E:\DEV\GO> Demo.exe
7 30 0
10 45 15
Lama parkir: 3 jam 15 menit 15 detik
```

7.2 Array

Array mempunyai ukuran (jumlah elemen) yang tetap (**statik**) 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.

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

Contoh:

```
1 // Mengganti isi elemen ke-0 dengan nilai dari elemen ke-7
2 arr[0] = arr[7]
3
4 // Mengambil data field x dari elemen ke-i
5 currX = arr[i].center.x
6
7 // Mengambil elemen terakhir
8 n := len(arr)
9 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 sl01 as a slice
5 var sl01 = []int{ 11, 2, 3, 5, 7, 13 }
```

Sebuah slice dapat diprealokasi menggunakan fungsi built-in **make**

```
1 // Prealokasi 10 elemen untuk sl02 dan sejumlah tempat tambahan
2 var sl02 []int = make([]int, 10, 20)
3
4 // Prealokasi 7 elemen untuk sl03 tanpa tempat tambahan
5 var sl03 []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 sl02
2 fmt.Println( len(sl02), cap(sl02) )
```

Fungsi built-in **append** dapat digunakan untuk menambahkan elemen ke suatu slice, dan bila perlu memperbesar tempat untuk slice tersebut.

```
1 /* 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. */
2 sl01 = append(sl01, 17)
3 sl01 = append(sl01, 19, 23)
```

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

```
1 // Ambil 3 elemen pertama dari suatu slice atau array
2 sl04 = arr[:4]
3
4 // Ambil beberapa elemen terakhir, dimulai dari indeks 5
5 sl05 = sl01[5:]
6
7 // Salin semua dari slice/array aslinya
8 sl06 = sl05[:]
9
10 // Salin element dari indeks 3 sampai, tapi tidak termasuk, 5.
11 // Jadi dalam contoh hanya 2 elemen sl06[3] dan sl06[4] yang disalin
12 sl07 = sl06[3:5]
```

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**.

```
1 // Deklarasi variabel dct sebagai map bilangan bulat dengan kunci string
2 var dct map[string]int
3
4 // Deklarasi map lain dct1 dari elemen string dengan kunci juga string
5 // Mempunyai nilai awal dct1["john"] = "hi", dct1["anne"] = "darling"
6 var dct1 = map[string]string{ "john":"hi", "anne":"darling" }
7
8 // Deklarasi dan prealokasi tempat untuk map dct2
9 var dct2 map[float64]int = make(map[float64]int, 10)
10
11 // Mengambil nilai yang tersimpan dengan kunci "john"
12 fmt.Println( dct1["john"] )
13
14 // Mengganti nilai yang tersimpan pada kunci "anne", dan
15 // Membuat entri baru dengan kunci "boy"
16 dct1["anne"] = "lovely"
17 dct1["boy"] = "runaround"
```

```
18  
19 // Menghapus entri dengan kunci "john"  
20 delete(dct1, "john")
```

II. GUIDED

1. Sourcecode :

```
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)
    }
}
```

Screenshot program :

```
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:\Alpro_Praktikum\Pandia Arya Brata_2311102076_Modul7>
```

Deskripsi program :

Program diatas memiliki program menghitung nilai rata2 mahasiswa berdasarkan skor dari tiga mata Pelajaran, yaitu Matematika, Fisika, Kimia, kemudian menampilkan peringkat mahasiswa dari yang tertinggi ke terendah.

2. Sourcecode :

```
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 program :

```
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  
-----  
20231004     Dedi  
20231001     Andi  
20231002     Budi  
PS C:\Alpro_Praktikum\Pandia Arya Brata_2311102076_Modul7>
```

Deskripsi program :

Program diatas merupakan program mengelola data mahasiswa menggunakan map dengan nim sebagai kunci dan nama sebagai nilai, serta menyediakan fungsionalitas untuk menambah ataupun menghapus data mahasiswa dalam format table.

III. UNGUIDED

1. Sourcecode :

```
package main  
  
import (  
    "fmt"  
    "math"  
)  
  
type Titik struct {  
    X int  
    Y int  
}  
  
type Circle struct {  
    Center Titik  
    Radius int  
}  
  
func jarak(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 priksa_titik(titik Titik, circle1, circle2 Circle) string {
    dist1 := jarak(titik, circle1.Center)
    dist2 := jarak(titik, circle2.Center)

    if dist1 <= float64(circle1.Radius) && dist2 <=
float64(circle2.Radius) {
        return "Titik di dalam lingkaran 1 dan 2"
    } else if dist1 <= float64(circle1.Radius) {
        return "Titik di dalam lingkaran 1"
    } else if dist2 <= float64(circle2.Radius) {
        return "Titik di dalam lingkaran 2"
    } else {
        return "Titik di luar lingkaran 1 dan 2"
    }
}

func main() {
    var (
        circle1 Circle
        circle2 Circle
        titik    Titik
    )

    fmt.Println("masukan kordinat titik pusat satu dan radius : ")
    fmt.Scanln(&circle1.Center.X, &circle1.Center.Y,
&circle1.Radius)

    fmt.Println("masukan kordinat titik pusat dua dan radius : ")
    fmt.Scanln(&circle2.Center.X, &circle2.Center.Y,
&circle2.Radius)

    fmt.Println("Masukkan koordinat titik sembarang (x, y):")
    fmt.Scanln(&titik.X, &titik.Y)

    result := priksa_titik(titik, circle1, circle2)
    fmt.Println(result)
}

```

Screenshot program :

```
masukan kordinat titik pusat satu dan radius :  
1 1 5  
masukan kordinat titik pusat dua dan radius :  
8 8 4  
Masukkan koordinat titik sembarang (x, y):  
2 2  
Titik di dalam lingkaran 1  
PS C:\Alpro_Praktikum\Pandia Arya Brata_2311102076_Modul7>  
masukan kordinat titik pusat satu dan radius :  
1 2 3  
masukan kordinat titik pusat dua dan radius :  
4 5 6  
Masukkan koordinat titik sembarang (x, y):  
7 8  
Titik di dalam lingkaran 2  
PS C:\Alpro_Praktikum\Pandia Arya Brata_2311102076_Modul7>
```

Deskripsi program :

Program ini memiliki program untuk menentukan sebuah titik relative terhadap dua lingkaran berdasarkan pusat dan radius yang dimasukkan oleh pengguna, serta memberikan informasi apakah titik tersebut berada di dalam, diluar, atau tepat pada batas lingkaran.

2. Sourcecode :

```
package main  
  
import (  
    "fmt"  
    "math"  
)  
  
func displayarr(arr []int) {  
    fmt.Println("element array:", arr)  
}  
  
func bilangan_ganjil(arr []int) {  
    fmt.Print("Elemen ganjil pada array: ")  
    for i := 1; i < len(arr); i += 2 {  
        fmt.Print(arr[i], " ")  
    }  
    fmt.Println()  
}  
  
func display_elementindx(arr []int) {  
    fmt.Print("Element genap pada array: ")  
    for i := 0; i < len(arr); i += 2 {  
        fmt.Print(arr[i], " ")  
    }  
}
```

```

    }
    fmt.Println()
}

func display_kelipatan(arr []int, x int) {
    fmt.Printf("Kelipatan dari element index %d: ", x)
    for i := x; i < len(arr); i += x {
        fmt.Print(arr[i], " ")
    }
    fmt.Println()
}

func hapus_element(arr []int, index int) []int {
    if index < 0 || index >= len(arr) {
        fmt.Println("Index tidk ditemukan")
        return arr
    }
    fmt.Printf(" index %d telah dihapus\n", index)
    return append(arr[:index], arr[index+1:]...)
}

func nilai_ratarata(arr []int) float64 {
    sum := 0
    for _, value := range arr {
        sum += value
    }
    return float64(sum) / float64(len(arr))
}

func rata_bilangan(arr []int) float64 {
    mean := nilai_ratarata(arr)
    sum := 0.0
    for _, value := range arr {
        sum += math.Pow(float64(value)-mean, 2)
    }
    return math.Sqrt(sum / float64(len(arr)))
}

func simpangbaku(arr []int, number int) int {
    frequency := 0
    for _, value := range arr {
        if value == number {
            frequency++
        }
    }
    return frequency
}

func main() {
    var n, x, index, target int

    fmt.Print("Masukan angka untuk element di dalam array: ")
    fmt.Scan(&n)

    arr := make([]int, n)
    fmt.Println("masukan: ")
    for i := 0; i < n; i++ {
        fmt.Scan(&arr[i])
    }

    fmt.Println("\nTABEL ARRAY")

```

```

fmt.Println("a. Tampilkan seluruh array")
fmt.Println("b. Tampilkan array dengan ideks ganjil")
fmt.Println("c. Tampilkan array dengan indeks genap")
fmt.Println("d. Kelipatan")
fmt.Println("e. Hapus element")
fmt.Println("f. Menghitung rata-rata")
fmt.Println("g. Mengitung standar deviasi atau simpang baku")
fmt.Println("h. Menghitung fekuensi dari suatu bilangan")

var pilih string
fmt.Print("Masukan pilihan anda (a-h): ")
fmt.Scan(&pilih)

switch pilih {
case "a":
    displayarr(arr)
case "b":
    bilangan_ganjil(arr)
case "c":
    display_elementindx(arr)
case "d":
    fmt.Print("Masukan nilai kelipatan x: ")
    fmt.Scan(&x)
    display_kelipatan(arr, x)
case "e":
    fmt.Print("Index yang ingin di hapus: ")
    fmt.Scan(&index)
    arr = hapus_element(arr, index)
    displayarr(arr)
case "f":
    average := nilai_ratarata(arr)
    fmt.Printf("Rata-rata dari element: %.2f\n", average)
case "g":
    stdDev := rata_bilangan(arr)
    fmt.Printf("standar atau simpang bakunya: %.2f\n", stdDev)
case "h":
    fmt.Print("Masukan angka untuk mencari freuensi: ")
    fmt.Scan(&target)
    frequency := simpangbaku(arr, target)
    fmt.Printf("Frekuensi dari %d: %d\n", target, frequency)
default:
    fmt.Println("ERRORR")
}
}

```

Screenshot program :

```
a. Tampilkan seluruh array
b. Tampilkan array dengan ideks ganjil
c. Tampilkan array dengan indeks genap
d. Kelipatan
e. Hapus element
f. Menghitung rata-rata
g. Mengitung standar deviasi atau simpang baku
h. Menghitung fekuensi dari suatu bilangan
Masukan pilihan anda (a-h): a
element array: [1 2 3 4 5]
PS C:\Alpro_Praktikum\Pandia Arya Brata_2311102076_Modul7>
Masukan angka untuk element di dalam array: 5
masukan:
1
2
3
4
5

TABEL ARRAY
a. Tampilkan seluruh array
b. Tampilkan array dengan ideks ganjil
c. Tampilkan array dengan indeks genap
d. Kelipatan
e. Hapus element
f. Menghitung rata-rata
g. Mengitung standar deviasi atau simpang baku
h. Menghitung fekuensi dari suatu bilangan
Masukan pilihan anda (a-h): b
Elemen ganjil pada array: 2 4
PS C:\Alpro_Praktikum\Pandia Arya Brata_2311102076_Modul7>
```

Deskripsi program :

Program ini mengelola suatu array bilangan bulat dengan menyediakan berbagai fungsi untuk menampilkan elemen, menghitung rata2, menghitung standar deivasi, menghapus elemen, dan menghitung frekuensi kemunculan bilangan tertentu, serta memungkinkan interaksi pengguna untuk memilih operasi yang di inginkan.

3. Sourcecode :

```
package main

import (
    "fmt"
)

func main() {
    var timA, timB string
    var scoreA, scoreB int
    var winners []string

    // Input club names
    fmt.Print("Tim A: ")
    fmt.Scanln(&timA)
    fmt.Print("Tim B: ")
    fmt.Scanln(&timB)

    matchNumber := 1

    for {
        fmt.Printf("Pertandingan %d : ", matchNumber)
        fmt.Scan(&scoreA, &scoreB)

        if scoreA < 0 || scoreB < 0 {
            fmt.Println("ERROR")
            break
        }

        if scoreA > scoreB {
            winners = append(winners, timA)
        } else if scoreB > scoreA {
            winners = append(winners, timB)
        } else {
            winners = append(winners, "Draw")
        }

        matchNumber++
    }

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

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

Screenshot program :

```
Tim A: MU
Tim B: Inter
Pertandingan 1 : 2 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
ERROR

Hasil Pertandingan:
Hasil 1: MU
Hasil 2: Inter
Hasil 3: Draw
Hasil 4: Inter
Hasil 5: MU
Hasil 6: MU
Hasil 7: MU
Hasil 8: Inter
Program Selesai
PS C:\Alpro_Praktikum\Pandia Arya Brata_2311102076_Modul7>
```

Deskripsi program :

Program ini memiliki program yang bisa mensimulasikan sebuah system yang mencatat hasil pertandingan speak bola antar dua klub hanya dengan user menginputkan kedua score tim tersebut.

4. Sourcecode :

```
package main

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

const NMAX int = 127

type tabel [NMAX]rune

func isiArray(tab *tabel, n *int) {
    fmt.Print("Masukkan teks (maksimal 127 karakter): ")
    input := bufio.NewReader(os.Stdin)
    text, _ := input.ReadString('\n')
```



```

    text = strings.TrimSpace(text)

    *n = len(text)
    for i := 0; i < *n && i < NMAX; i++ {
        tab[i] = rune(text[i])
    }
}

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

func balikkanArray(tab *tabel, n int) {
    for i, j := 0, n-1; i < j; i, j = i+1, j-1 {
        tab[i], tab[j] = tab[j], tab[i]
    }
}

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

func main() {
    var tab tabel
    var n int

    isiArray(&tab, &n)

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

    balikkanArray(&tab, n)
    fmt.Print("Reverse: ")
    cetakArray(tab, n)

    if palindrom(tab, n) {
        fmt.Println("Palindrome")
    } else {
        fmt.Println("Bukan Palindrome")
    }
}

```

Screenshot program :

```

PS C:\Alpro_Praktikum\Pandia Arya Brata_2311102076_Modul7>
Masukkan teks (maksimal 127 karakter): S E N A N G
Teks: S E N A N G
Reverse: G N A N E S
Bukan Palindrome
PS C:\Alpro_Praktikum\Pandia Arya Brata_2311102076_Modul7>

```

Deskripsi program :

Program ini memiliki program yang membaca teks dari inputan pengguna, dan mengonversikan teks menjadi array rune, dan kemudian mencetak array tersebut ke konsol, baik secara langsung maupun dalam urutan terbalik.