

**LAPORAN PRAKTIKUM
ALGORITMA DAN PEMROGRAMAN 2**

**MODUL 7
STRUCK & ARRAY**



Disusun Oleh :

Aby Hakim Al Yasiry Faozi/2311102208

IF-11-06

Dosen Pengampu :

Abednego Dwi Septiadi

PROGRAM STUDI S1 TEKNIK INFORMATIKA

FAKULTAS INFORMATIKA

TELKOM UNIVERSITY PURWOKERTO

2024

I. DASAR TEORI

A. Struct (Structure)

Struct adalah sebuah tipe data terstruktur dalam bahasa pemrograman, seperti C atau C++, yang memungkinkan pengguna untuk mengelompokkan berbagai tipe data menjadi satu unit logis. Struct digunakan untuk merepresentasikan sebuah entitas yang memiliki beberapa atribut dengan tipe data yang berbeda.

Karakteristik Struct:

- **Pengelompokan Data:** Struct memungkinkan pengelompokan berbagai tipe data seperti int, float, atau char dalam satu struktur.
- **Definisi Custom:** Struct didefinisikan oleh programmer untuk merepresentasikan objek atau entitas tertentu.
- **Akses Elemen:** Anggota dari sebuah struct dapat diakses menggunakan operator titik (.) jika menggunakan variabel, atau panah (->) jika menggunakan pointer.

B. Array

Array adalah sebuah struktur data yang digunakan untuk menyimpan sekumpulan elemen dengan tipe data yang sama dalam satu blok memori. Elemen-elemen array diakses melalui indeks yang dimulai dari 0.

Karakteristik Array:

- **Homogen:** Semua elemen dalam array harus memiliki tipe data yang sama.
- **Ukuran Tetap:** Ukuran array harus ditentukan saat deklarasi dan tidak dapat diubah selama program berjalan.
- **Akses Cepat:** Elemen array dapat diakses langsung menggunakan indeks.

II. GUIDED

1. Soal Studi Case

Sourcecode

```
package main

import (
    "fmt"
    "math"
)

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

// Struktur untuk 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.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("Masukan 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("Masukan 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("Masukan 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 di luar kedua lingkaran")
    }
}

```

Screenshoot Output

```

akim\Kuliah\Semester 3\prak alpro 2>
go run "c:\Aby Hakim\Kuliah\Semester
3\prak alpro 2\modul7\guided1\guided1
.go"
Masukan koordinat pusat dan radius li
ngkaran 1 (cx1, cy1, r1): 1 2 3
Masukan koordinat pusat dan radius li
ngkaran 2 (cx2, cy2, r2): 4 5 6
Masukan koordinat titik sembarang (x,
y): 7
Titik di dalam lingkaran 2
PS C:\Aby Hakim\Kuliah\Semester 3\pra
k alpro 2> 

```

Deskripsi Program

- Struktur Data (Struct)
 - Titik: Ini adalah struktur yang digunakan untuk menyimpan koordinat dari sebuah titik (x dan y).
 - Lingkaran: Ini adalah struktur yang digunakan untuk menyimpan pusat lingkaran (yang diwakili oleh struktur Titik) dan radius lingkaran.

- Fungsi untuk Menghitung Jarak

Fungsi ini digunakan untuk menghitung jarak antara dua titik menggunakan rumus jarak Euclidean. Jarak Euclidean antara dua titik dihitung dengan mengukur garis lurus di antara kedua titik tersebut di dalam sistem koordinat Cartesian.

- Fungsi untuk Memeriksa Posisi Titik dalam Lingkaran

Fungsi ini memeriksa apakah jarak antara titik sembarang dan pusat lingkaran lebih kecil atau sama dengan radius lingkaran. Jika jarak tersebut lebih kecil atau sama dengan radius, maka titik berada di dalam lingkaran.

- Input Pengguna

Program meminta pengguna untuk memasukkan koordinat pusat dan radius dua lingkaran, serta koordinat dari titik sembarang.

- Pemeriksaan Posisi Titik

Program memeriksa apakah titik sembarang tersebut berada di dalam lingkaran pertama, lingkaran kedua, atau kedua lingkaran berdasarkan hasil dari fungsi pemeriksaan posisi.

- Output Hasil

Berdasarkan hasil pemeriksaan, program akan menampilkan pesan yang menunjukkan apakah titik berada di dalam lingkaran pertama, lingkaran kedua, kedua lingkaran, atau di luar kedua lingkaran.

III. UNGUIDED

1. Soal Studi Case

Sourcecode

```
package main

import (
    "fmt"
    "math"
)

func main() {
    var n, x, hapusIndex, nomer int

    fmt.Print("Masukkan jumlah elemen array (N): ")
    fmt.Scan(&n)

    arr := make([]int, n)

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

    fmt.Println("\nIsi array:")
    fmt.Println(arr)

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

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

    fmt.Print("\nMasukkan nilai x untuk indeks
kelipatan: ")
    fmt.Scan(&x)
    fmt.Printf("\nElemen dengan indeks kelipatan %d:\n",
x)
    for i := 0; i < len(arr); i++ {
        if i%x == 0 {
            fmt.Printf("%d ", arr[i])
        }
    }
    fmt.Println()
}
```

```

        fmt.Print("\nMasukkan indeks yang ingin dihapus: ")
        fmt.Scan(&hapusIndex)
        if hapusIndex >= 0 && hapusIndex < len(arr) {
            arr = append(arr[:hapusIndex],
arr[hapusIndex+1:]...)
            fmt.Println("Isi array setelah elemen dihapus:")
            fmt.Println(arr)
        } else {
            fmt.Println("Indeks tidak valid.")
        }

        var sum float64
        for _, value := range arr {
            sum += float64(value)
        }
        average := sum / float64(len(arr))
        fmt.Printf("\nRata-rata elemen dalam array: %.2f\n",
average)

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

        fmt.Print("\nMasukkan bilangan yang ingin dihitung
frekuensinya: ")
        fmt.Scan(&nomer)
        count := 0
        for _, value := range arr {
            if value == nomer {
                count++
            }
        }
        fmt.Printf("Frekuensi bilangan %d: %d kali\n",
nomer, count)
    }

```

Screenshoot Output

```
akim\Kuliah\Semester 3\prak alpro 2> go run
"c:\Aby Hakim\Kuliah\Semester 3\prak alpro
2\modul7\unguided1\unguided.go"
Masukkan jumlah elemen array (N): 5
Masukkan elemen array:
Elemen ke-0: 1
Elemen ke-1: 2
Elemen ke-2: 3
Elemen ke-3: 4
Elemen ke-4: 5

Isi array:
[1 2 3 4 5]

Elemen dengan indeks ganjil:
2 4

Elemen dengan indeks genap:
1 3 5

Masukkan nilai x untuk indeks kelipatan: 2

Elemen dengan indeks kelipatan 2:
1 3 5

Masukkan indeks yang ingin dihapus: 1
Isi array setelah elemen dihapus:
[1 3 4 5]

Rata-rata elemen dalam array: 3.25
Standar deviasi elemen dalam array: 1.48

Masukkan bilangan yang ingin dihitung freku
ensinya: 2
Frekuensi bilangan 2: 0 kali
PS C:\Aby Hakim\Kuliah\Semester 3\prak alpr
o 2> █
```


Deskripsi Program

- **Inisialisasi Array:**
 - Pengguna diminta untuk memasukkan jumlah elemen array yang ingin dibuat.
 - Berdasarkan jumlah elemen yang diberikan, program membuat array dengan ukuran tersebut.
- **Pengisian Array:**
 - Pengguna memasukkan elemen-elemen array satu per satu.
 - Program menyimpan setiap elemen yang dimasukkan oleh pengguna ke dalam array.
- **Menampilkan Isi Array:**
 - Program menampilkan seluruh elemen yang ada dalam array.
- **Menampilkan Elemen Berdasarkan Indeks Ganjil:**
 - Program menampilkan elemen-elemen array yang berada pada indeks ganjil (misalnya, indeks 1, 3, 5, dst.).
- **Menampilkan Elemen Berdasarkan Indeks Genap:**
 - Program menampilkan elemen-elemen array yang berada pada indeks genap (misalnya, indeks 0, 2, 4, dst.).
- **Menampilkan Elemen Berdasarkan Kelipatan Indeks:**
 - Pengguna diminta untuk memasukkan nilai x .
 - Program menampilkan elemen-elemen array yang berada pada indeks yang merupakan kelipatan dari x (misalnya, jika $x = 2$, maka elemen pada indeks 0, 2, 4, dst.).
- **Menghapus Elemen Berdasarkan Indeks:**
 - Pengguna diminta untuk memasukkan indeks elemen yang ingin dihapus.
 - Program menghapus elemen yang berada pada indeks tersebut, jika indeks valid.

- **Menghitung Rata-rata:**
 - Program menghitung rata-rata dari seluruh elemen yang ada dalam array.
 - Rata-rata dihitung dengan menjumlahkan semua elemen kemudian membagi dengan jumlah elemen.
- **Menghitung Standar Deviasi:**
 - Program menghitung standar deviasi dari elemen-elemen dalam array.
 - Standar deviasi mengukur seberapa jauh data dalam array menyimpang dari rata-rata.
- **Menghitung Frekuensi Elemen Tertentu:**
 - Pengguna diminta untuk memasukkan nilai elemen yang ingin dihitung frekuensinya.
 - Program menghitung berapa kali elemen tersebut muncul dalam array.

2. Soal Studi Case

Sourcecode

```
package main

import "fmt"

type Match struct {
    clubA  string
    clubB  string
    scoreA int
    scoreB int
    winner string
}

func main() {
    var clubA, clubB string
    fmt.Print("Klub A: ")
    fmt.Scan(&clubA)
    fmt.Print("Klub B: ")
    fmt.Scan(&clubB)

    var matches []Match
```

```

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

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

        var winner string
        if scoreA > scoreB {
            winner = clubA
        } else if scoreA < scoreB {
            winner = clubB
        } else {
            winner = "Draw"
        }

        match := Match{
            clubA: clubA,
            clubB: clubB,
            scoreA: scoreA,
            scoreB: scoreB,
            winner: winner,
        }
        matches = append(matches, match)
    }

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

```

Screenshoot Output

```
akim\Kuliah\Semester 3\prak alpro 2> go run "c:\Aby Hakim\Kuliah\Semester 3\prak alpro 2\modul 7\unguided2\unguided2.go"
Klub A: MU
Klub 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
Pertandingan selesai

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
PS C:\Aby Hakim\Kuliah\Semester 3\prak alpro 2>
```

Deskripsi Program

- **Struktur Data (struct) Match:**
 - **Match** adalah sebuah struktur yang menyimpan informasi tentang sebuah pertandingan, termasuk nama kedua klub (clubA dan clubB), skor masing-masing klub (scoreA dan scoreB), dan pemenang pertandingan (winner).
- **Pengambilan Input Nama Klub:**
 - Pengguna diminta untuk memasukkan nama kedua klub yang akan bertanding.
- **Loop untuk Mengambil Skor Pertandingan:**
 - Program memasuki loop yang meminta pengguna untuk memasukkan skor untuk setiap pertandingan.
 - Loop ini berlanjut hingga pengguna memasukkan nilai negatif untuk skor salah satu klub, yang menandakan akhir dari input pertandingan.

- **Penentuan Pemenang Pertandingan:**
 - Setelah menerima skor untuk sebuah pertandingan, program menentukan pemenangnya:
 - Jika skor klub A lebih besar dari klub B, maka klub A adalah pemenang.
 - Jika skor klub B lebih besar dari klub A, maka klub B adalah pemenang.
 - Jika skor keduanya sama, maka pertandingan berakhir imbang (draw).
- **Penyimpanan Hasil Pertandingan:**
 - Informasi tentang setiap pertandingan (klub yang bertanding, skor, dan pemenang) disimpan dalam sebuah slice dari struktur Match.
- **Penghentian Loop dan Pemberhentian Pertandingan:**
 - Jika pengguna memasukkan skor negatif, program keluar dari loop dan mencetak pesan bahwa pertandingan telah selesai.
- **Penampilan Hasil Pertandingan:**
 - Setelah semua pertandingan selesai, program menampilkan hasil setiap pertandingan, menunjukkan pemenangnya atau jika pertandingan berakhir imbang.

3. Soal Studi Case

Sourcecode

```
package main

import "fmt"

const NMAX = 127

type Tabel struct {
    tab [NMAX]rune
    n    int
}

func isiArray(t *Tabel) {
    fmt.Print("Teks: ")
```

```

var input string
fmt.Scanln(&input)
for i, char := range input {
    t.tab[i] = char
}
t.n = len(input)
}

func cetakArray(t Tabel) {
    for i := 0; i < t.n; i++ {
        fmt.Print(string(t.tab[i]))
    }
    fmt.Println()
}

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

func palindrome(t Tabel) bool {
    for i, j := 0, t.n-1; i < j; i, j = i+1, j-1 {
        if t.tab[i] != t.tab[j] {
            return false
        }
    }
    return true
}

func main() {
    var tab Tabel

    isiArray(&tab)

    fmt.Print("Reverse teks: ")
    cetakArray(tab)

    balikkanArray(&tab)

    fmt.Print("Reverse teks: ")
    cetakArray(tab)

    if palindrome(tab) {
        fmt.Println("Palindrome: true")
    } else {
        fmt.Println("Palindrome: false")
    }
}

```

Screenshoot Output

```
akim\Kuliah\Semester 3\prak alpro > go run "c:\
Aby Hakim\Kuliah\Semester 3\prak alpro 2\modul7
\unguided3\unguided3.go"
Teks: SENANG
Reverse teks: SENANG
Reverse teks: GNANES
Palindrome: false
PS C:\Aby Hakim\Kuliah\Semester 3\prak alpro 2>
```

Deskripsi Program

- **Deklarasi Konstanta dan Struktur:**
 - Program menetapkan sebuah konstanta NMAX yang menunjukkan jumlah maksimum karakter yang dapat disimpan dalam array.
 - Program mendeklarasikan struktur Tabel yang berisi array karakter (tab) dan sebuah integer (n) yang melacak panjang teks.
- **Fungsi untuk Mengisi Array:**
 - Program meminta pengguna untuk memasukkan teks.
 - Teks yang dimasukkan oleh pengguna dipecah menjadi karakter-karakter individu dan disimpan dalam array tab di dalam struktur Tabel.
 - Panjang teks disimpan dalam n.
- **Fungsi untuk Mencetak Array:**
 - Program mencetak isi array tab sebagai teks.
- **Fungsi untuk Membalikkan Array:**
 - Program membalikkan urutan karakter dalam array tab.
 - Karakter pertama ditukar dengan karakter terakhir, karakter kedua ditukar dengan karakter kedua dari akhir, dan seterusnya.
- **Fungsi untuk Memeriksa Palindrom:**
 - Program memeriksa apakah teks yang disimpan dalam array tab adalah palindrom.

- Palindrom berarti teks sama jika dibaca dari depan ke belakang dan sebaliknya.
- Fungsi ini membandingkan karakter dari ujung ke ujung, bergerak ke tengah. Jika semua karakter cocok, maka teks adalah palindrom.
- **Fungsi Utama:**
 - Program memanggil fungsi untuk mengisi array dengan teks dari pengguna.
 - Program mencetak teks yang dimasukkan pengguna.
 - Program membalikkan teks tersebut dan mencetak hasilnya.
 - Program memeriksa apakah teks yang dibalikkan adalah palindrom dan menampilkan hasilnya.