

**LAPORAN PRAKTIKUM
ALGORITMA DAN PEMROGRAMAN 2**

**MODUL 7
STRUCT DAN ARRAY**



Disusun Oleh :

Muhammad Djoko Susilo / 2311102212

Dosen Pengampu :

Abednego Dwi Septiadi, S.Kom., M.Kom

PROGRAM STUDI S1 TEKNIK INFORMATIKA

FAKULTAS INFORMATIKA

TELKOM UNIVERSITY PURWOKERTO

2024

I. DASAR TEORI

Pengertian Struct

Struktur adalah koleksi dari variabel yang dinyatakan dengan sebuah nama, dengan sifat setiap variabel dapat memiliki tipe yang berlainan. Struktur biasa dipakai untuk mengelompokkan beberapa informasi yang berkaitan menjadi sebuah satu kesatuan.

Pengertian Array

Array adalah organisasi kumpulan data homogen yang ukuran atau jumlah elemen maksimumnya telah diketahui dari awal. Array umumnya disimpan di memori komputer secara kontigu (berurutan). Deklarasi dari array adalah sebagai berikut: `int A[5]`; artinya variabel A adalah kumpulan data sebanyak 5 bilangan bertipe integer. Operasi terhadap elemen di array dilakukan dengan pengaksesan langsung. Nilai di masing-masing posisi elemen dapat diambil dan nilai dapat disimpan tanpa melewati posisi-posisi lain.

Keunggulan dan Kelemahan Array

Keunggulan array adalah sebagai berikut:

1. Array sangat cocok untuk pengaksesan acak. Sembarang elemen di array dapat diacu secara langsung tanpa melalui elemen-elemen lain.
2. Jika berada di suatu lokasi elemen, maka sangat mudah menelusuri ke elemen-elemen tetangga, baik elemen pendahulu atau elemen penerus
3. Jika elemen-elemen array adalah nilai-nilai independen dan seluruhnya harus terjaga, maka penggunaan penyimpanannya sangat efisien.

Kelemahan array.

Array mempunyai fleksibilitas rendah, karena array mempunyai batasan sebagai berikut:

1. Array harus bertipe homogen. Kita tidak dapat mempunyai array dimana satu elemen adalah karakter, elemen lain bilangan, dan elemen lain adalah tipe-tipe lain
2. Kebanyakan bahasa pemrograman mengimplementasikan array statik yang sulit diubah ukurannya di waktu eksekusi. Bila penambahan dan pengurangan terjadi terus-menerus, maka representasi statis.

Tidak efisien dalam penggunaan memori Menyiaikan banyak waktu komputasi Pada suatu aplikasi, representasi statis tidak dimungkinkan

II. GUIDED

1. 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. **Gunakan tipe bentukan titik untuk mentimpan koordinat, dan tipe bentukan lingkaran untuk mentimpan 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 didalam lingkaran 1”, “Titik didalam lingkaran 2” atau “Titik diluar lingkaran 1 dan 2”.

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 titik 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}

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")
}
}

```

Screenshot Output

```
29 }
30
31 func main(){
32     // Input untuk lingkaran 1
33     var cx1, cy1, r1 int
34     fmt.Print("Masukan koordinat pusat dan radius lingkaran 1 (cx1, cy1, r1): ")
35     fmt.Scanln(&cx1, &cy1, &r1)
36     lingkaran1 := Lingkaran{pusat: Titik{x: cx1, y: cy1}, radius: r1}
37
38     // Input untuk lingkaran 2
39     var cx2, cy2, r2 int
40     fmt.Print("Masukan koordinat pusat dan radius lingkaran 2 (cx2, cy2, r2): ")
41     fmt.Scanln(&cx2, &cy2, &r2)
42     lingkaran2 := Lingkaran{pusat: Titik{x: cx2, y: cy2}, radius: r2}
43
44     var x, y int
45     fmt.Print("Masukan koordinat titik sembarang (x, y): ")
46     fmt.Scanln(&x, &y)
47     titik := Titik{x: x, y: y}
48 }
```

```
PS D:\code\2311102212_Djoko_Modul-6> go run "d:\code\2311102212_Djoko_Modul-6\guided1\guided1.go"
Masukan koordinat pusat dan radius lingkaran 1 (cx1, cy1, r1): 2 2 0
Masukan koordinat pusat dan radius lingkaran 2 (cx2, cy2, r2): 3 1 4
Masukan koordinat titik sembarang (x, y): 3 4
Titik di dalam lingkaran 2
PS D:\code\2311102212_Djoko_Modul-6>
```

Deskripsi Program

Program ini dibuat menggunakan struct untuk menyimpan beberapa variabel, Program ini bertujuan untuk mengetahui berada dimana titik itu berada, ada di lingkaran 1 atau 2 atau dua-duanya atau malah tidak ada di keduanya. Cara kerja dari program ini adalah pertama program akan meminta user untuk memasukkan koordinat pusat dan radius dari lingkaran 1 dan 2 lalu meminta untuk memasukkan koordinat titik sembarang, setelah itu program akan melakukan perhitungan menggunakan rumus yang sudah disediakan dan akan menampilkan hasil yang sesuai dengan perhitungan yang sudah dilakukan dan akan memunculkan sebuah kalimat yang menunjukkan berada dimana titik tersebut

III. UNGUIDED

1. Soal Studi Case

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 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 harus 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.

Sourcecode

```
package main

import (
    "fmt"
    "math"
)

func main() {
    var bilangan [10]int

    for i := 0; i < 10; i++ {
        bilangan[i] = i
    }

    count := 10
    // Jawaban A
    fmt.Print("A.Array : ")
    for i := 0; i < count; i++ {
        fmt.Printf("%d ", bilangan[i])
    }

    // Jawab B
    fmt.Print("\nB.Ganjil : ")
    for i := 0; i < count; i++ {
        if bilangan[i] % 2 == 1 {
            fmt.Printf("%d ", bilangan[i])
        }
    }

    // Jawab C
    fmt.Print("\nC.Genap : ")
    for i := 0; i < count; i++ {
        if bilangan[i] == 0 {
            fmt.Printf("%d ", bilangan[i])
        } else if bilangan[i] % 2 == 0 {
            fmt.Printf("%d ", bilangan[i])
        }
    }

    // Jawab D
    fmt.Print("\nD.Kelipatan : ")
    var kelipatan int
    fmt.Scan(&kelipatan)
    for i := 0; i < count; i++ {
        if bilangan[i] % kelipatan == 0 {
            fmt.Printf("%d ", bilangan[i])
        }
    }

    // Jawab E
```

```

var slice []int = make([]int, len(bilangan))
copy(slice, bilangan[:])

fmt.Print("\nE. Indeks : ")
var bilHapus int
fmt.Scan(&bilHapus)

// Pengecekan indeks valid
if bilHapus >= 0 && bilHapus < len(slice) {
    // Hapus elemen pada indeks
    slice = append(slice[:bilHapus],
slice[bilHapus+1:]...)
    fmt.Println("Array setelah penghapusan:",
slice)
} else {
    fmt.Println("Indeks tidak valid")
}

// Jawab F
fmt.Print("\nF.")
var resultAvg int
for i := 0; i < count; i++ {
    resultAvg += bilangan[i]
}
resultAvg = resultAvg/count
fmt.Printf("Rata rata : %d", resultAvg)

// Jawab G
fmt.Print("\nG.")
var variance float64
for _, val := range bilangan {
    variance += math.Pow(float64(val-
resultAvg), 2)
}
variance /= float64(count - 1)

stdDev := math.Sqrt(variance)

fmt.Printf("%.2f\n", stdDev)

// Jawab H
fmt.Print("\nH.")
var sliceForH []int = make([]int,
len(bilangan))
copy(sliceForH, bilangan[:])
frekuensi :=
hitungFrekuensiDenganArray(sliceForH)

// Cetak hasil
for i := 0; i <= 100; i++ {
    if frekuensi[i] > 0 {

```

```

                                fmt.Printf("Angka %d muncul
sebanyak %d kali\n", i, frekuensi[i])
                                }
                                }
                                }

func hitungFrekuensiDenganArray(data []int) []int {
    // Asumsi nilai maksimum dalam data adalah 100
    (sesuaikan jika perlu)
    maxVal := 100
    freq := make([]int, maxVal+1) // Index 0 sampai
    maxVal

    for _, val := range data {
        freq[val]++
    }

    return freq
}

```

Screenshot Output

```

PS D:\code\2311102212_Djoko_Modul-6> go run "d:\code\2311102212_Djoko_Modul-6\unGuided1\unGuided1.go"
A.Array : 0 1 2 3 4 5 6 7 8 9
B.Ganjil : 1 3 5 7 9
C.Genap : 0 2 4 6 8
D.Kelipatan : 3
0 3 6 9
E. Indeks : 3
Array setelah penghapusan: [0 1 2 4 5 6 7 8 9]

F.Rata rata : 4
G.3.07

H.Angka 0 muncul sebanyak 1 kali
Angka 1 muncul sebanyak 1 kali

H.Angka 0 muncul sebanyak 1 kali
Angka 1 muncul sebanyak 1 kali
Angka 2 muncul sebanyak 1 kali
Angka 3 muncul sebanyak 1 kali
Angka 4 muncul sebanyak 1 kali
Angka 5 muncul sebanyak 1 kali
Angka 6 muncul sebanyak 1 kali

```

Deskripsi Program

Program ini ditulis dalam bahasa Go untuk memanipulasi dan menganalisis array bilangan bulat dengan berbagai fitur. Program menerima input pengguna untuk beberapa operasi dan menyediakan hasil analisis, seperti:

- a. Menampilkan isi array - Menampilkan semua elemen array.
- b. Menampilkan elemen dengan indeks ganjil - Menampilkan elemen-elemen pada indeks 1, 3, 5, dst.
- c. Menampilkan elemen dengan indeks genap - Menampilkan elemen-elemen pada indeks 0, 2, 4, dst.
- d. Menampilkan elemen dengan indeks kelipatan bilangan tertentu (x) - Pengguna memberikan bilangan x, dan program menampilkan elemen pada indeks kelipatan x.
- e. Menghapus elemen pada indeks tertentu - Menghapus elemen berdasarkan indeks yang dimasukkan pengguna, lalu menampilkan array yang telah diperbarui.
- f. Menghitung rata-rata - Menghitung rata-rata nilai dari semua elemen array.
- g. Menghitung standar deviasi - Menghitung simpangan baku elemen-elemen dalam array untuk mengetahui sebaran data.
- h. Menghitung frekuensi kemunculan bilangan tertentu - Menghitung berapa kali setiap angka muncul dalam array.

Program ini memanfaatkan fungsi dasar dari bahasa Go, seperti iterasi dengan loop, manipulasi slice, dan penggunaan fungsi matematika untuk perhitungan standar deviasi.

2. Soal Studi Case

Buatlah sebuah program untuk menentukan siapa pemenang dari histori pertandingan dua klub, Pertama program akan meminta untuk memasukan kedua klub yang dipertandingkan dan akan dimulai untuk memasukan jumlah gol yang tercipta, mulai dari pertandingan 1, gol untuk tim 1 dan selanjutnya gol untuk tim 2 begitu terus sampai berhenti. Program akan berhenti ketika gol tim 1 atau tim 2 memiliki nilai minus(-).

Sourcecode

```
package main

import "fmt"

type Klub struct {
    skor1 int
    skor2 int
    hasil string
}

func main() {

    var klub1, klub2 string
    var jumlahMatch int
    var match [100]Klub
```

```

fmt.Print("Masukkan Klub 1: ")
fmt.Scanln(&klub1)
fmt.Print("Masukkan Klub 2: ")
fmt.Scanln(&klub2)

for i := 0; ; i++ {
    fmt.Printf("Skor Pertandingan %d :  ", i+1)
    fmt.Printf("\n %s : ", klub1)
    fmt.Scanln(&match[i].skor1)
    fmt.Printf("%s : ", klub2)
    fmt.Scanln(&match[i].skor2)

    if match[i].skor1 < 0 || match[i].skor2 < 0 {
        fmt.Print("Pertandingan Selesai")
        break
    }

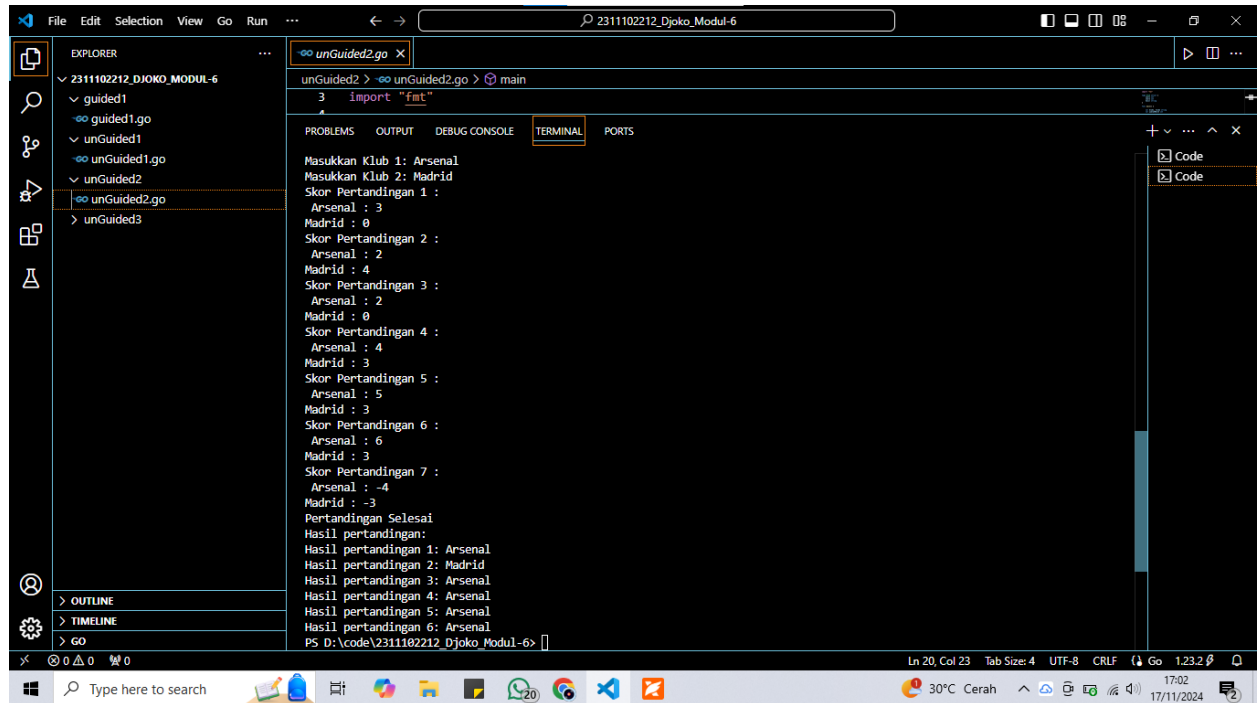
    if match[i].skor1 > match[i].skor2 {
        match[i].hasil = klub1
    } else if match[i].skor1 == match[i].skor2 {
        match[i].hasil = "Draw"
    } else {
        match[i].hasil = klub2
    }

    jumlahMatch++
}

fmt.Println("\nHasil pertandingan:")
for i := 0; i < jumlahMatch; i++ {
    fmt.Printf("Hasil pertandingan %d: %s\n", i+1,
match[i].hasil)
}
}

```

Screenshot Output



```
unGuided2.go
3 import "fmt"
4
5 func main() {
6     fmt.Println("Masukkan Klub 1: Arsenal")
7     fmt.Println("Masukkan Klub 2: Madrid")
8     Skor Pertandingan 1 :
9     Arsenal : 3
10    Madrid : 0
11    Skor Pertandingan 2 :
12    Arsenal : 2
13    Madrid : 4
14    Skor Pertandingan 3 :
15    Arsenal : 2
16    Madrid : 0
17    Skor Pertandingan 4 :
18    Arsenal : 4
19    Madrid : 3
20    Skor Pertandingan 5 :
21    Arsenal : 5
22    Madrid : 3
23    Skor Pertandingan 6 :
24    Arsenal : 6
25    Madrid : 3
26    Skor Pertandingan 7 :
27    Arsenal : -4
28    Madrid : -3
29    Pertandingan Selesai
30    Hasil pertandingan:
31    Hasil pertandingan 1: Arsenal
32    Hasil pertandingan 2: Madrid
33    Hasil pertandingan 3: Arsenal
34    Hasil pertandingan 4: Arsenal
35    Hasil pertandingan 5: Arsenal
36    Hasil pertandingan 6: Arsenal
37    Hasil pertandingan 7: Arsenal
38    PS D:\code\2311102212_Djoko_Modul-6>
```

Deskripsi Program

Program ini adalah simulasi pertandingan sepak bola antara dua klub, di mana pengguna dapat memasukkan skor untuk setiap pertandingan.

Pertama menginput nama klub: Pengguna diminta memasukkan nama dua klub yang akan bertanding.

Kedua mencatatkan skor: Untuk setiap pertandingan, pengguna memasukkan skor masing-masing klub.

Ketiga menentukan pemenang: Program menentukan hasil pertandingan berdasarkan skor:

- Jika skor klub pertama lebih besar, klub pertama menang.
- Jika skor klub kedua lebih besar, klub kedua menang.
- Jika skor sama, hasilnya seri (Draw).

Terakhir menghentikan input: Pengguna dapat menghentikan input dengan memberikan skor negatif untuk salah satu klub.

Setelah semua pertandingan dicatat, program menampilkan hasil masing-masing pertandingan, termasuk pemenang atau jika hasilnya seri.

Program ini memanfaatkan array untuk menyimpan data skor dan hasil setiap pertandingan, serta menggunakan perulangan untuk mencatat skor hingga pengguna memutuskan berhenti.

3. Soal Studi Case

Buatlah program dari kode yang sudah disiapkan dan melengkapi potongan potongan kode yang sudah disiapkan, tujuan dari ini adalah untuk melakukan pengecekan kalimat yang diinputkan polindrom atau bukan.

Sourcecode

```
package main

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

const NMAX int = 127

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

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

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

func isiArray(t *tabel) {
    reader := bufio.NewReader(os.Stdin)
    fmt.Print("Masukkan teks: ")
    input, _ := reader.ReadString('\n')
    input = strings.TrimSpace(input)

    t.m = len(input)
    for i, c := range input {
        t.tab[i] = c
    }
}

func main() {
    var tab tabel

    isiArray(&tab)

    fmt.Print("Teks: ")
}
```

```

        cetakArray(tab, tab.m)

        balikanArray(&tab, tab.m)

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

```

Screenshoot Program

The screenshot shows a Go IDE with the following components:

- EXPLORER:** A file tree on the left showing a project named '2311102212_DJOKO_MODUL-6' with subfolders 'guided1' and 'unGuided3'. The file 'unGuided3.go' is selected.
- Source Code:** The main editor displays the code for 'unGuided3.go'. It includes a function 'isiArray' that reads input into a character array 'tab', and a 'main' function that calls 'isiArray', prints the text, calls 'balikanArray' to reverse it, and prints the reversed text.
- TERMINAL:** The bottom panel shows the output of running the program. It displays the prompt 'Masukkan teks:', the user input 'Kasur Rusak', and the program's output 'Reverse teks: kasuR rusak'.

Deskripsi Program

Program ini membaca teks dari pengguna, menampilkan teks tersebut, kemudian membalik urutan karakter dalam teks dan menampilkannya kembali. Berikut adalah detail fungsinya:

1. Fungsi isiArray:

Membaca teks dari pengguna menggunakan buffer.

Menyimpan karakter teks ke dalam array tab dan mencatat panjang teks di variabel m.

2. Fungsi cetakArray:

Menampilkan elemen array tab hingga panjang yang ditentukan (n).

3. Fungsi balikanArray:

Membalik urutan elemen dalam array tab dengan cara menukar elemen pertama dengan terakhir, kedua dengan kedua terakhir, dan seterusnya.

Proses program: Pertama membaca teks dari pengguna dan menyimpannya di array lalu menampilkan teks asli setelah itu membalik teks menggunakan balikanArray dan terakhir akan menampilkan teks yang sudah dibalik.

