

**LAPORAN PRAKTIKUM
ALGORITMA DAN PEMROGRAMAN 2**

**MODUL VII
STRUCK & ARRAY**



Disusun Oleh :

Rafi Bintang Maulana / 2311102327

Kelas IF-11-06

Dosen Pengampu :

ABEDNEGO DWI SEPTIADI

PROGRAM STUDI S1 TEKNIK INFORMATIKA

FAKULTAS INFORMATIKA

TELKOM UNIVERSITY PURWOKERTO

2024

I. DASAR TEORI

Struct (singkatan dari structure) dalam bahasa pemrograman adalah tipe data bentukan yang memungkinkan pengguna untuk mengelompokkan berbagai tipe data yang berbeda dalam satu entitas. dalam sebuah struct, setiap elemen disebut field atau anggota, yang dapat memiliki tipe data yang berbeda.

Array adalah struktur data yang menyimpan sejumlah elemen dengan tipe data yang sama di dalam sebuah urutan memori yang berkesinambungan. Elemen-elemen dalam array diakses melalui indeks, di mana indeks pertama biasanya dimulai dari nol.

II. GUIDED

1. 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 radlusnya.

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".

Sourcecode

```
package main

import (
    "fmt"
    "math"
)

type Titik struct {
    x int
    y int
}

type Lingkaran struct {
    pusat Titik
    radius int
}

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)))
}

func titikDidalamLingkaran(t Titik, l Lingkaran) bool {
    jarak := hitungJarak(t, l.pusat)
    return jarak <= float64(l.radius)
}

func main(){
    var cx1, cy1, r1 int
    fmt.Print("Masukan koordinat pusat dan radius lingkaran 1 (cx1 cv1 r1):
```

```

")
    fmt.Scanln(&cx1, &cy1, &r1)
    lingkaran1 := Lingkaran{pusat: Titik{x: cx1, y: cy1}, radius: r1}

    var cx2, cy2, r2 int
    fmt.Print("Masukan koordinat pusat dan radius lingkaran 1 (cx1 cv1 r1):
")
    fmt.Scanln(&cx2, &cy2, &r2)
    lingkaran2 := Lingkaran{pusat: Titik{x: cx2, y: cy2}, radius: r2}

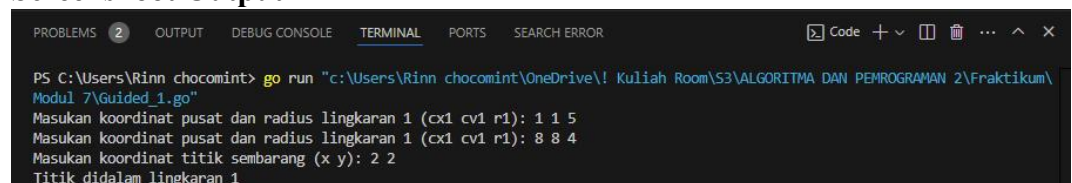
    var x, y int
    fmt.Print("Masukan koordinat titik sembarang (x y): ")
    fmt.Scanln(&x, &y)
    titik := Titik{x: x, y: y}

    diDalamL1 := titikDidalamLingkaran(titik, lingkaran1)
    diDalamL2 := titikDidalamLingkaran(titik, lingkaran2)

    if diDalamL1 && diDalamL2 {
        fmt.Println("Titik di dalam lingkaran 1 dan 2")
    } else if diDalamL1 {
        fmt.Println("Titik didalam lingkaran 1")
    } else if diDalamL2 {
        fmt.Println("Titik didalam lingkaran 2")
    } else {
        fmt.Println("Titik berada di luar kedua lingkaran")
    }
}

```

Screenshoot Output



```

PS C:\Users\Rinn chocomint> go run "c:\Users\Rinn chocomint\OneDrive\! Kuliah Room\S3\ALGORITMA DAN PEMROGRAMAN 2\Fraktikum\
Modul 7\Guided_1.go"
Masukan koordinat pusat dan radius lingkaran 1 (cx1 cv1 r1): 1 1 5
Masukan koordinat pusat dan radius lingkaran 1 (cx1 cv1 r1): 8 8 4
Masukan koordinat titik sembarang (x y): 2 2
Titik didalam lingkaran 1

```

Deskripsi Program

Program digunakan untuk menentukan apakah suatu titik berada di dalam satu atau dua lingkaran yang didefinisikan oleh pengguna. Menggunakan dua struktur data utama, yaitu Titik, untuk merepresentasikan koordinat suatu titik, dan Lingkaran, yang terdiri dari titik pusat dan radiusnya. Untuk menghitung jarak antara dua titik, program memanfaatkan fungsi “**hitungJarak**” yang menerapkan rumus Euclidean dengan akar kuadrat dari selisih kuadrat koordinat x dan y kedua titik tersebut.

III. UNGUIDED

1. 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:
 - a. Menampilkan keseluruhan isi dari array.
 - b. Menampilkan elemen-elemen array dengan indeks ganjil saja.
 - c. Menampilkan elemen-elemen array dengan indeks genap saja (asumsi indeks ke-0 adalah genap).
 - d. Menampilkan elemen-elemen array dengan indeks kelipatan bilangan dari masukan pengguna. X. X bisa diperoleh
 - e. Menghapus elemen array pada indeks tertentu, asumsi indeks yang hapus selalu valid. Tampilkan keseluruhan isi dari arraynya, pastikan data yang dihapus tidak tampil
 - f. Menampilkan rata-rata dari bilangan yang ada di dalam array.
 - g. Menampilkan standar deviasi atau simpangan baku dari bilangan yang ada di dalam array tersebut.
 - h. Menampilkan frekuensi dari suatu bilangan tertentu di dalam array yang telah diisi tersebut.

Sourcecode

```
package main
import (
    "fmt"
    "math"
)

func main() {
    var daftarAngka []int
    var jumlah int
    fmt.Print("Masukkan jumlah angka: ")
    fmt.Scan(&jumlah)

    if jumlah > 0 {
        daftarAngka = make([]int, jumlah)
        fmt.Println("Masukkan angka-angka:")
        for i := 0; i < jumlah; i++ {
            fmt.Printf("Angka ke-%d: ", i)
            fmt.Scan(&daftarAngka[i])
        }
    } else {
```

```

        fmt.Println("Jumlah angka harus lebih besar dari 0!")
        return
    }
    fmt.Println("\nSemua angka:", daftarAngka)

    fmt.Print("Angka dengan indeks ganjil: ")
    for i := 1; i < len(daftarAngka); i += 2 {
        fmt.Print(daftarAngka[i], " ")
    }
    fmt.Println()

    fmt.Print("Angka dengan indeks genap: ")
    for i := 0; i < len(daftarAngka); i += 2 {
        fmt.Print(daftarAngka[i], " ")
    }
    fmt.Println()

    var kelipatan int
    fmt.Print("Masukkan kelipatan x: ")
    fmt.Scan(&kelipatan)
    if kelipatan > 0 {
        fmt.Printf("Angka dengan indeks kelipatan %d: ",
kelipatan)
        for i := 0; i < len(daftarAngka); i++ {
            if i%kelipatan == 0 {
                fmt.Print(daftarAngka[i], " ")
            }
        }
        fmt.Println()
    } else {
        fmt.Println("x harus lebih besar dari 0!")
    }

    for {
        fmt.Println("\n--- Ambil Tindakan ---")
        fmt.Println("1. Hapus angka pada indeks tertentu")
        fmt.Println("2. Hitung rata-rata angka")
        fmt.Println("3. Hitung standar deviasi angka")
        fmt.Println("4. Hitung frekuensi angka tertentu")
        fmt.Println("0. Keluar")
        fmt.Print("Pilih menu: ")
    }

```

```

var pilihan int
fmt.Scan(&pilihan)

switch pilihan {
case 1:
    var indeks int
    fmt.Print("Masukkan indeks yang mau dihapus: ")
    fmt.Scan(&indeks)
    if indeks >= 0 && indeks < len(daftarAngka) {
        daftarAngka =
append(daftarAngka[:indeks], daftarAngka[indeks+1:]...)
        fmt.Println("Daftar angka setelah
dihapus:", daftarAngka)
    } else {
        fmt.Println("Indeks nggak valid!")
    }

case 2:
    if len(daftarAngka) > 0 {
        total := 0
        for _, angka := range daftarAngka {
            total += angka
        }
        rata := float64(total) /
float64(len(daftarAngka))
        fmt.Println("Rata-rata angka:", rata)
    } else {
        fmt.Println("Daftar angka kosong!")
    }

case 3:
    if len(daftarAngka) > 0 {
        total := 0
        for _, angka := range daftarAngka {
            total += angka
        }
        rata := float64(total) /
float64(len(daftarAngka))

        var variansi float64

```

```

                                for _, angka := range daftarAngka {
                                    variansi +=
math.Pow(float64(angka)-rata, 2)
                                }
                                variansi /= float64(len(daftarAngka))
                                stdDev := math.Sqrt(variansi)
                                fmt.Println("Standar deviasi angka:",
stdDev)
                                } else {
                                    fmt.Println("Daftar angka kosong!")
                                }

                                case 4:
                                    if len(daftarAngka) > 0 {
                                        var angkaDicari int
                                        fmt.Print("Masukkan angka yang mau
dihitung frekuensinya: ")
                                        fmt.Scan(&angkaDicari)
                                        hitung := 0
                                        for _, angka := range daftarAngka {
                                            if angka == angkaDicari {
                                                hitung++
                                            }
                                        }
                                        fmt.Println("Frekuensi angka",
angkaDicari, ":", hitung)
                                    } else {
                                        fmt.Println("Daftar angka kosong!")
                                    }

                                case 0:
                                    fmt.Println("Keluar dari program. Makasih!")
                                    return

                                default:
                                    fmt.Println("Pilihan nggak valid! Pilih yang ada di
menu.")
                                }
                            }
                        }
                    }
                }
            }
        }
    }
}

```


Screenshoot Output

```
Masukkan jumlah angka: 10
Masukkan angka-angka:
Angka ke-0: 2
Angka ke-1: 3
Angka ke-2: 1
Angka ke-3: 1
Angka ke-4: 1
Angka ke-5: 0
Angka ke-6: 2
Angka ke-7: 3
Angka ke-8: 2
Angka ke-9: 7

Semua angka: [2 3 1 1 1 0 2 3 2 7]
Angka dengan indeks ganjil: 3 1 0 3 7
Angka dengan indeks genap: 2 1 1 2 2
Masukkan kelipatan x: 2
Angka dengan indeks kelipatan 2: 2 1 1 2 2

--- Ambil Tindakan ---
1. Hapus angka pada indeks tertentu
2. Hitung rata-rata angka
3. Hitung standar deviasi angka
4. Hitung frekuensi angka tertentu
0. Keluar
Pilih menu: 1
Masukkan indeks yang mau dihapus: 9
Daftar angka setelah dihapus: [2 3 1 1 1 0 2 3 2]

--- Ambil Tindakan ---
1. Hapus angka pada indeks tertentu
2. Hitung rata-rata angka
3. Hitung standar deviasi angka
4. Hitung frekuensi angka tertentu
0. Keluar
Pilih menu: 2
Rata-rata angka: 1.6666666666666667

--- Ambil Tindakan ---
1. Hapus angka pada indeks tertentu
2. Hitung rata-rata angka
3. Hitung standar deviasi angka
4. Hitung frekuensi angka tertentu
0. Keluar
Pilih menu: 3
Standar deviasi angka: 0.9428090415820634

--- Ambil Tindakan ---
1. Hapus angka pada indeks tertentu
2. Hitung rata-rata angka
3. Hitung standar deviasi angka
4. Hitung frekuensi angka tertentu
0. Keluar
Pilih menu: 4
Masukkan angka yang mau dihitung frekuensinya: 3
Frekuensi angka 3 : 2
```

Deskripsi Program

Program ini berfungsi untuk mengolah daftar angka yang diinputkan pengguna, Setelah angka diinputkan, Program akan menampilkan angka-angka berdasarkan indeks ganjil, genap, atau kelipatan tertentu. Program juga menyediakan menu tindakan seperti menghapus angka pada indeks tertentu, menghitung rata-rata, standar deviasi, atau frekuensi kemunculan angka tertentu setelah pengguna menginputkan semua yang diminta oleh program.

2. 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 berlaga. 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
```

Sourcecode

```
package main
import (
    "fmt"
)

func main() {
    var klubA, klubB string
    var skorA, skorB int
```

```

var yangMenang []string

fmt.Print("Klub A : ")
fmt.Scanln(&klubA)

fmt.Print("Klub B : ")
fmt.Scanln(&klubB)

fmt.Println()

pertandingan := 1
for {
    fmt.Printf("Pertandingan %d : ", pertandingan)
    _, err := fmt.Scan(&skorA, &skorB)

    if err != nil {
        fmt.Println("Cara inputnya salah, input 2 nomer
sekaligus!")
        continue
    }

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

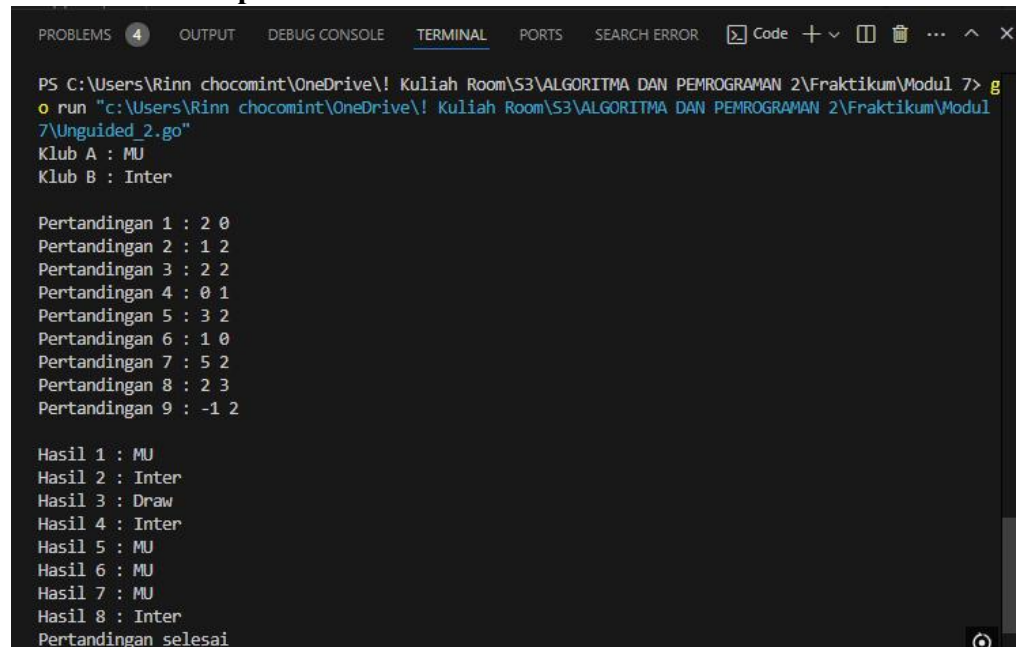
    if skorA > skorB {
        yangMenang = append(yangMenang, klubA)
    } else if skorA < skorB {
        yangMenang = append(yangMenang, klubB)
    } else {
        yangMenang = append(yangMenang, "Draw")
    }

    pertandingan++
}

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

```

Screenshoot Output

A screenshot of a Go IDE terminal window. The terminal shows the execution of a program that takes two club names as input and outputs the results of nine matches. The input is 'Klub A : MU' and 'Klub B : Inter'. The output lists nine matches with scores and the final result for each match. The matches are: 1: MU 2, 2: Inter 1, 3: MU 2, 4: Inter 0, 5: MU 3, 6: MU 1, 7: MU 5, 8: Inter 2, 9: MU -1. The final result is 'Pertandingan selesai'.

Deskripsi Program

Program itu berguna untuk mencatat hasil pertandingan antara dua klub yang nama klub nya di inputkan pengguna. Untuk setiap pertandingan, program menentukan pemenang berdasarkan skor, Jika skor Klub A lebih besar, maka Klub A dianggap menang, Jika skor Klub B lebih besar, Klub B dianggap menang, dan jika skor sama, hasilnya adalah "Draw". Dan program hanya akan berhenti jika pengguna memasukkan skor negatif.

3. 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 yang ada di modul.

Sourcecode

```
package main
import (
    "bufio"
    "fmt"
    "os"
    "strings"
)

const NMAX int = 127
```

```

type tabel [NMAX]rune

func isiArray(t *tabel, n *int) {
    fmt.Print("Teks\t\t: ")
    reader := bufio.NewReader(os.Stdin)
    input, _ := reader.ReadString('\n')
    input = strings.TrimSpace(input)
    input = strings.ToUpper(input)
    *n = 0
    for _, char := range input {
        if char == '!' {
            break
        }
        if *n < NMAX {
            t[*n] = char
            *n++
        }
    }
}

func palindrom(t tabel, n int) bool {
    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.Print("Palindrom\t\t? ")
    if palindrom(tab, m) {
        fmt.Println("true")
    } else {
        fmt.Println("false")
    }
}

```

Screenshoot Output

```
PS C:\Users\Rinn chocomint\OneDrive\! Kuliah Room\S3\ALGORITMA DAN PEMROGRAMAN 2\Fraktikum\Modul 7> g
o run "c:\Users\Rinn chocomint\OneDrive\! Kuliah Room\S3\ALGORITMA DAN PEMROGRAMAN 2\Fraktikum\Modul
7\Unguided_3.go"
Teks      : S E N A N G
Palindrom  ? false
PS C:\Users\Rinn chocomint\OneDrive\! Kuliah Room\S3\ALGORITMA DAN PEMROGRAMAN 2\Fraktikum\Modul 7>
```

Deskripsi Program

Program itu untuk memeriksa apakah sebuah teks yang diinputkan pengguna itu palindrom atau tidak. Membaca teks dari input pengguna hingga karakter titik ('.') atau hingga jumlah maksimum karakter (127). Teks diubah menjadi huruf kapital untuk menyederhanakan pemeriksaan. Teks yang diinput disimpan di array, lalu diperiksa apakah susunannya sama saat dibaca dari depan maupun belakang. Jika teks adalah palindrom, program mencetak "true" dan Jika tidak, mencetak "false".