

**LAPORAN PRAKTIKUM
ALGORITMA DAN PEMROGRAMAN 2**

**MODUL 6
STRUCK & ARRAY**



Oleh: Dimas Fanny Hebrasianto Permadi

NAMA: Dimas Ramadhani

NIM: 103112400065

KELAS: IF-12-02

**S1 TEKNIK INFORMATIKA
TELKOM UNIVERSITY PURWOKERTO**

2025

I. DASAR TEORI

1. Tipe Bentuk

Tipe bentuk memungkinkan kita membuat tipe data baru, yang bisa membantu menyederhanakan kode dan membuat lebih mudah dipahami.

- **Alias (Type)**

Alias digunakan untuk memberikan nama baru pada tipe data yang sudah ada. Misalnya, daripada menulis `int` berulang kali, kita bisa membuat alias type seperti `bilangan`. Ini berguna ketika kita ingin memperjelas konteks dari data yang sedang digunakan.

- **Struct**

Struct atau struktur data adalah cara untuk mengelompokkan beberapa data menjadi satu kesatuan. Misalnya, untuk mempresentasikan waktu, kita bisa buat struct waktu yang berisi jam, menit, dan detik. Dengan begitu, semua data terkait waktu bisa dikelola bersama-sama dalam satu variabel.

2. Array

Array adalah struktur data yang digunakan untuk menyimpan kumpulan elemen dengan tipe yang sama. Ukuran array bersifat tetap, sehingga kita harus tahu sejak awal beberapa jumlah data yang akan disimpan. Di dalam bahasa Go, array memiliki indeks yang dimulai dari 0.

3. Slice

Slice adalah versi fleksibel dari array. Slice bisa berubah ukuran selama program berjalan. Kita bisa menambahkan elemen baru ke dalam slice menggunakan fungsi `append`.

Slice juga mendukung berbagai operasi lain seperti mengambil sebagian elemen, mengecek panjang (`len()`), dan kapasitas (`cap()`).

4. Map

Map adalah struktur data yang menyimpan data dalam bentuk pasangan key-value. Kunci pada map tidak harus berupa angka, bisa juga berupa string atau tipe lainnya.

II. GUIDED

1. Nomor 1

- **Source Code:**

```
package main

import (
    "fmt"
    "time"
)

// Struct untuk barang dalam struk belanja
type Item struct {
    Name    string
    Price   float64
    Quantity int
}

// Struct untuk struk belanja
type Receipt struct {
    StoreInfo string
    Date      time.Time
    Items     []Item
    TotalAmount float64
}

// Method untuk menghitung total harga semua item
func (r *Receipt) CalculateTotal() {
    var total float64
    for _, item := range r.Items {
        total += item.Price * float64(item.Quantity)
    }
    r.TotalAmount = total
}

// Method untuk mencetak struk belanja
func (r Receipt) PrintReceipt() {
    fmt.Println("=====")
    fmt.Println(r.StoreInfo)
    fmt.Println("Tanggal:", r.Date.Format("02-01-2006 15:04"))
}
```

```

fmt.Println("=====
====")
fmt.Printf("%-15s %-10s %-8s %-10s\n", "Item", "Harga",
"Jumlah", "Total")
fmt.Println("-----")

for _, item := range r.Items {
    itemTotal := item.Price * float64(item.Quantity)
    fmt.Printf("%-15s Rp%-9.2f %-8d Rp%-9.2f\n",
item.Name, item.Price, item.Quantity, itemTotal)
}

fmt.Println("=====
====")
fmt.Printf("%-35s Rp%-9.2f\n", "Total Belanja:", r.TotalAmount)
fmt.Println("=====
====")
fmt.Println("Terima kasih telah berbelanja!")
}

func main() {
receipt := Receipt{
    StoreInfo: "Toko Sembako Makmur\nJl. Raya No. 123,
Jakarta",
    Date:    time.Now(),
    Items: []Item{
        {Name: "Beras", Price: 12000, Quantity: 5},
        {Name: "Gula", Price: 15000, Quantity: 2},
        {Name: "Minyak", Price: 20000, Quantity: 1},
        {Name: "Telur", Price: 2000, Quantity: 10},
    },
}

receipt.CalculateTotal()
receipt.PrintReceipt()
}

```

- **Screenshot Hasil Program:**

```
PS C:\Users\Dimas\OneDrive\Collage\Semester 2\Algoritma dan Pemrograman 2\Praktikum\Laparak\10
=====
Toko Sembako Makmur
Jl. Raya No. 123, Jakarta
Tanggal: 24-04-2025 19:34
=====
Item          Harga          Jumlah    Total
-----
Beras         Rp12000.00     5         Rp60000.00
Gula          Rp15000.00     2         Rp30000.00
Minyak        Rp20000.00     1         Rp20000.00
Telur         Rp2000.00      10        Rp20000.00
=====
Total Belanja:                Rp130000.00
=====
Terima kasih telah berbelanja!
```

- **Penjelasan:**

Program ini bertujuan untuk mengelola dan mencetak struk belanja menggunakan bahasa pemrograman Go. Program menerima data seperti nama barang, harga, jumlah barang yang dibeli, lalu menghitung total harga dari semua barang, dan menampilkannya dalam format struk belanja.

Program ini menggunakan struct dengan nama Item yang terdiri dari tiga field, yaitu Name untuk menyimpan nama barang, Price untuk menyimpan harga satuan barang, dan Quantity untuk menyimpan jumlah barang yang dibeli. Semua field tersebut digunakan untuk merepresentasikan satu jenis barang dalam belanjaan.

Selain itu, program juga menggunakan struct lain dengan nama Receipt. Struct ini berfungsi sebagai tempat penyimpanan informasi struk belanja secara keseluruhan, yang terdiri dari StoreInfo untuk informasi toko, Date untuk menyimpan tanggal dan waktu transaksi, Items yang merupakan kumpulan dari struct Item, serta TotalAmount untuk menyimpan hasil total perhitungan harga semua barang.

Program ini memiliki method dengan nama CalculateTotal yang bertipe void dan dimiliki oleh struct Receipt. Method ini berfungsi untuk menghitung total harga dari semua barang yang terdapat

dalam slice Items. Di dalam method ini dideklarasikan variabel total bertipe float64 dengan nilai awal 0. Lalu dilakukan perulangan terhadap setiap item dalam r.Items. Di dalam perulangan, harga per barang akan dikalikan dengan jumlah barang, kemudian hasilnya dijumlahkan ke dalam variabel total. Setelah perulangan selesai, nilai total disimpan ke dalam field TotalAmount.

Program juga memiliki method dengan nama PrintReceipt yang bertipe void dan juga dimiliki oleh struct Receipt. Method ini bertugas untuk mencetak isi struk belanja ke layar dengan format yang rapi. Informasi yang dicetak meliputi informasi toko, tanggal pembelian, daftar barang (nama, harga, jumlah, total per barang), total belanja, serta ucapan terima kasih. Format tampilan struk dibuat agar menyerupai struk belanja asli.

Masuk ke dalam fungsi utama (main), program akan membuat objek Receipt dengan mengisi field StoreInfo, Date, dan Items secara langsung. Dalam slice Items terdapat beberapa barang seperti "Beras", "Gula", "Minyak", dan "Telur", masing-masing dengan harga dan jumlah tertentu. Setelah semua data terisi, program akan memanggil method CalculateTotal() untuk menghitung total harga seluruh barang, lalu memanggil PrintReceipt() untuk mencetak struk belanja ke layar.

2. Nomor 2

- **Source Code:**

```
package main

import (
    "fmt"
)

func main() {
    // Deklarasi dan inisialisasi array nilai mahasiswa
    nilaiMahasiswa := [5]int{85, 90, 78, 88, 95}

    fmt.Println("Data Nilai Mahasiswa:")
    fmt.Println("=====")

    // Menampilkan nilai per mahasiswa
    for i, nilai := range nilaiMahasiswa {
        fmt.Printf("Mahasiswa %d: %d\n", i+1, nilai)
    }

    // Menghitung rata-rata nilai
    var total int
    for _, nilai := range nilaiMahasiswa {
        total += nilai
    }
    rataRata := float64(total) / float64(len(nilaiMahasiswa))

    fmt.Println("=====")
    fmt.Printf("Rata-rata nilai: %.2f\n", rataRata)

    // Mencari nilai tertinggi dan terendah
    tertinggi := nilaiMahasiswa[0]
    terendah := nilaiMahasiswa[0]

    for _, nilai := range nilaiMahasiswa {
        if nilai > tertinggi {
            tertinggi = nilai
        }
        if nilai < terendah {
            terendah = nilai
        }
    }
}
```

```

    }
}

fmt.Printf("Nilai tertinggi: %d\n", tertinggi)
fmt.Printf("Nilai terendah: %d\n", terendah)

// Contoh array 2 dimensi
fmt.Println("\nContoh Array 2 Dimensi:")
fmt.Println("=====")

// Nilai ujian mahasiswa dalam 2 mata kuliah (Matematika,
Bahasa)
nilaiUjian := [3][2]int{
    {80, 85},
    {90, 75},
    {70, 95},
}

// Menampilkan nilai ujian per mahasiswa
fmt.Println("Nilai Ujian Mahasiswa (Matematika, Bahasa):")
for i, nilai := range nilaiUjian {
    fmt.Printf("Mahasiswa %d: Matematika = %d, Bahasa =
%d\n", i+1, nilai[0], nilai[1])
}
}

```


- **Screenshot Hasil Program:**

```
PS C:\Users\Dimas\OneDrive\Collage\Semester 2\er 2\Algoritma dan Pemrograman 2\Praktikum\Lap
Data Nilai Mahasiswa:
=====
Mahasiswa 1: 85
Mahasiswa 2: 90
Mahasiswa 3: 78
Mahasiswa 4: 88
Mahasiswa 5: 95
=====
Rata-rata nilai: 87.20
Nilai tertinggi: 95
Nilai terendah: 78

Contoh Array 2 Dimensi:
=====
Nilai Ujian Mahasiswa (Matematika, Bahasa):
Mahasiswa 1: Matematika = 80, Bahasa = 85
Mahasiswa 2: Matematika = 90, Bahasa = 75
Mahasiswa 3: Matematika = 70, Bahasa = 95
```

- **Penjelasan:**

Program ini bertujuan untuk mengolah data nilai mahasiswa menggunakan array satu dimensi dan dua dimensi dalam bahasa pemrograman Go. Program menampilkan nilai setiap mahasiswa, menghitung rata-rata nilai, mencari nilai tertinggi dan terendah, serta menampilkan nilai ujian mahasiswa dalam dua mata kuliah berbeda.

Program diawali dengan pendeklarasian dan inisialisasi array satu dimensi nilaiMahasiswa yang berisi lima buah nilai bertipe int, mewakili nilai dari lima orang mahasiswa.

Selanjutnya, program mencetak teks "Data Nilai Mahasiswa" dan menampilkan masing-masing nilai menggunakan perulangan for dengan fungsi range. Di dalam perulangan, program mencetak indeks mahasiswa (ditambah 1 agar dimulai dari 1, bukan 0) beserta nilainya.

Program kemudian melanjutkan dengan proses penghitungan rata-rata nilai mahasiswa. Pertama, dideklarasikan variabel total bertipe

int dengan nilai awal 0. Nilai total kemudian diperoleh dengan menjumlahkan semua elemen dalam array nilaiMahasiswa melalui perulangan for. Setelah total diperoleh, program menghitung rata-rata dengan membagi total nilai dengan jumlah mahasiswa. Karena hasil rata-rata bisa berupa pecahan, tipe data float64 digunakan.

Setelah menghitung rata-rata, program mencari nilai tertinggi dan terendah dari array. Pertama-tama, nilai awal dari variabel tertinggi dan terendah diset dari elemen pertama array. Kemudian dilakukan perulangan untuk membandingkan setiap nilai dengan tertinggi dan terendah, dan memperbarui nilainya jika ditemukan yang lebih tinggi atau lebih rendah.

Program juga menampilkan contoh penggunaan array dua dimensi, yang merepresentasikan nilai ujian tiga mahasiswa dalam dua mata kuliah yaitu Matematika dan Bahasa. Array dua dimensi nilaiUjian bertipe [3][2]int digunakan untuk menyimpan nilai tersebut, di mana setiap baris mewakili satu mahasiswa dan dua kolom mewakili dua mata kuliah.

Akhirnya, program mencetak isi array dua dimensi dengan menggunakan perulangan. Dalam setiap iterasi, program mencetak nilai Matematika dan Bahasa untuk masing-masing mahasiswa.

III. UNGUIDED

1. Nomor 1

- **Source Code:**

```
package main

import (
    "fmt"
    "math"
)

func jarakTitik(a, b, x, y float64) float64 {
    return math.Sqrt(math.Pow(a-x, 2) + math.Pow(b-y, 2))
}

func cekLingkaran(a, b, c, x, y float64) bool {
    return jarakTitik(a, b, x, y) <= c
}

func main() {
    var x1, x2, y1, y2, r1, r2, x, y float64
    fmt.Print("Masukkan koordinat lingkaran 1: ")
    fmt.Scan(&x1, &y1, &r1)
    fmt.Print("Masukkan koordinat lingkaran 2: ")
    fmt.Scan(&x2, &y2, &r2)
    fmt.Print("Masukkan koordinat titik: ")
    fmt.Scan(&x, &y)
    jarakLingkaran1 := cekLingkaran(x1, y1, r1, x, y)
    jarakLingkaran2 := cekLingkaran(x2, y2, r2, x, y)
    if jarakLingkaran1 && jarakLingkaran2 {
        fmt.Print("Titik di dalam lingkaran 1 dan 2")
    } else if jarakLingkaran1 {
        fmt.Print("Titik di dalam lingkaran 1")
    } else if jarakLingkaran2 {
        fmt.Print("Titik di dalam lingkaran 2")
    } else {
        fmt.Print("Titik di luar lingkaran 1 dan 2")
    }
}

/*
Dimas Ramadhani
103112400065
*/
```

- **Screenshot Hasil Pembahasan:**

```

PS C:\Users\Dimas\OneDrive\Collage\Semester 2\Algoritma dan Pemrograman 2\Praktikum
Masukkan koordinat lingkaran 1: 1 1 5
Masukkan koordinat lingkaran 2: 8 8 4
Masukkan koordinat titik: 2 2
Titik di dalam lingkaran 1
PS C:\Users\Dimas\OneDrive\Collage\Semester 2\Algoritma dan Pemrograman 2\Praktikum
Masukkan koordinat lingkaran 1: 1 2 3
Masukkan koordinat lingkaran 2: 4 5 6
Masukkan koordinat titik: 7 8
Titik di dalam lingkaran 2
PS C:\Users\Dimas\OneDrive\Collage\Semester 2\Algoritma dan Pemrograman 2\Praktikum
Masukkan koordinat lingkaran 1: 5 10 15
Masukkan koordinat lingkaran 2: -15 4 20
Masukkan koordinat titik: 0 0
Titik di dalam lingkaran 1 dan 2
PS C:\Users\Dimas\OneDrive\Collage\Semester 2\Algoritma dan Pemrograman 2\Praktikum
Masukkan koordinat lingkaran 1: 1 1 5
Masukkan koordinat lingkaran 2: 8 8 4
Masukkan koordinat titik: 15 20
Titik di luar lingkaran 1 dan 2

```

- **Penjelasan:**

Program ini bertujuan untuk mengecek posisi suatu titik terhadap dua buah lingkaran, apakah titik tersebut berada di dalam salah satu, kedua, atau di luar kedua lingkaran. Program menerima input berupa koordinat pusat dan jari-jari dari dua lingkaran serta koordinat titik yang akan dicek. Semua nilai yang dimasukkan bertipe float64.

Bilangan untuk menggambarkan koordinat pusat lingkaran pertama akan disimpan pada variabel x1 dan y1, sedangkan jari-jarinya disimpan pada variabel r1. Untuk lingkaran kedua, koordinat

pusatnya disimpan di x2 dan y2, dan jari-jarinya di r2. Sementara koordinat titik yang akan dicek disimpan pada variabel x dan y.

Program ini menggunakan prosedur dengan nama jarakTitik bertipe float64, dengan parameter a, b, x, dan y bertipe float64. Prosedur ini menghitung jarak antara dua titik menggunakan rumus: $Jarak = \sqrt{(a - x)^2 + (b - y)^2}$ dan mengembalikan nilai hasil perhitungan jarak tersebut.

Program ini juga menggunakan prosedur cekLingkaran bertipe bool, dengan parameter a, b, c, x, dan y bertipe float64. Prosedur ini memanggil prosedur jarakTitik untuk menghitung jarak antara titik pusat lingkaran (a, b) dengan titik (x, y), kemudian membandingkan hasilnya dengan jari-jari c. Prosedur ini akan mengembalikan true jika titik berada di dalam atau tepat di garis lingkaran, dan false jika di luar.

Masuk ke program utama, variabel-variabel x1, x2, y1, y2, r1, r2, x, dan y bertipe float64 dideklarasikan. Program kemudian meminta pengguna untuk memasukkan koordinat pusat dan jari-jari dari dua lingkaran, serta koordinat titik yang akan dicek.

Setelah input diterima, program memanggil prosedur cekLingkaran untuk kedua lingkaran guna mengecek posisi titik. Program kemudian melakukan percabangan berdasarkan hasil tersebut:

- Jika titik berada di dalam lingkaran 1 dan 2, program mencetak "Titik di dalam lingkaran 1 dan 2".
- Jika hanya berada di dalam lingkaran 1, program mencetak "Titik di dalam lingkaran 1".
- Jika hanya berada di dalam lingkaran 2, program mencetak "Titik di dalam lingkaran 2".
- Jika tidak berada di dalam keduanya, program mencetak "Titik di luar lingkaran 1 dan 2".

2. Nomor 2

- Source Code:

```
package main

import (
    "fmt"
    "math"
)

func main() {
    var N, n, hapus, hasil int
    fmt.Print("Masukkan jumlah elemen: ")
    fmt.Scan(&N)
    bilArr := make([]int, N)
    for i := 0; i < N; i++ {
        fmt.Printf("Masukkan indeks ke-%d: ", i)
        fmt.Scan(&bilArr[i])
    }

    // soal a
    fmt.Println("=== SOAL A ===")
    fmt.Println("Isi dari array:", bilArr)

    // soal b
    fmt.Println()
    fmt.Println("=== SOAL B ===")
    for i := 0; i < N; i++ {
        if i%2 != 0 {
            fmt.Println("Elemen dengan indeks ganjil:",
bilArr[i])
        }
    }

    // soal c
    fmt.Println()
    fmt.Println("=== SOAL C ===")
    for i := 0; i < N; i++ {
        if i%2 == 0 {
            fmt.Println("Elemen dengan indeks genap:",
bilArr[i])
        }
    }
}
```

```

    }
}

// soal d, menampilkan elemen array dengan indeks kelipatan n
fmt.Println()
fmt.Println("==== SOAL D ===")
fmt.Print("Masukkan kelipatan: ")
fmt.Scan(&n)
for i := 0; i < len(bilArr); i++ {
    if i%n == 0 && i != 0 {
        fmt.Printf("Elemen dengan indeks kelipatan ke-%d:
%d\n", n, bilArr[i])
    }
}

// soal e
fmt.Println()
fmt.Println("==== SOAL E ===")
fmt.Print("Menghapus elemen ke-")
fmt.Scan(&hapus)
bilArr = append(bilArr[:hapus], bilArr[hapus+1:]...)
fmt.Printf("Array setelah menghapus elemen ke-%d: %d\n", hapus,
bilArr)

// soal f
var rataRata float64
fmt.Println()
fmt.Println("==== SOAL F ===")
for _, f := range bilArr {
    hasil += f
}
rataRata = float64(hasil) / float64(len(bilArr))
fmt.Printf("Rata-ratanya dari isi array: %.2f\n", rataRata)

// soal g
fmt.Println()
fmt.Println("==== SOAL G ===")
var simpBaku, selisih, jumSelisih float64
for _, xi := range bilArr {
    selisih = float64(xi) - float64(rataRata)
}

```

```

        jumSelisih += math.Pow(selisih, 2)
    }
    simpBaku = math.Sqrt(jumSelisih / float64(len(bilArr)))
    fmt.Printf("Simpangan Baku: %.2f\n", simpBaku)

    fmt.Println("\n=====")
    fmt.Println("Dimas Ramadhani")
    fmt.Println("103112400065")
    fmt.Print("=====\n")
    // Soal h
    var h, frek int
    fmt.Println()
    fmt.Println("=== SOAL H ===")
    fmt.Print("Masukkan angka untuk mencari frekuensinya: ")
    fmt.Scan(&h)
    for _, ha:= range bilArr{
        if h==ha {
            frek++
        }
    }
    fmt.Printf("Jadi jumlah angka %d adalah %d", h, frek)
}

```


- **Screenshot Hasil Pembahasan:**

```
PS C:\Users\Dimas\OneDrive\Collage\Semester 2\Algoritma dan Pemrograman 2\Praktikum\Lapra
Masukkan jumlah elemen: 5
Masukkan indeks ke-0: 2
Masukkan indeks ke-1: 4
Masukkan indeks ke-2: 2
Masukkan indeks ke-3: 6
Masukkan indeks ke-4: 3
=== SOAL A ===
Isi dari array: [2 4 2 6 3]

=== SOAL B ===
Elemen dengan indeks ganjil: 4
Elemen dengan indeks ganjil: 6

=== SOAL C ===
Elemen dengan indeks genap: 2
Elemen dengan indeks genap: 2
Elemen dengan indeks genap: 3

=== SOAL D ===
Masukkan kelipatan: 3
Elemen dengan indeks kelipatan ke-3: 6

=== SOAL E ===
Menghapus elemen ke-2
Array setelah menghapus elemen ke-2: [2 4 6 3]
```

```
=== SOAL F ===  
Rata-ratanya dari isi array: 3.75  
  
=== SOAL G ===  
Simpangan Baku:1.48  
  
=====
```

Dimas Ramadhani
103112400065

```
=====
```

=== SOAL H ===
Masukkan angka untuk mencari frekuensinya: 2
Jadi jumlah angka 2 adalah 1

- **Penjelasan:**

Program ini bertujuan untuk mengolah sebuah array dari input pengguna dan melakukan berbagai operasi terhadap data tersebut, seperti menampilkan elemen berdasarkan indeks tertentu, menghitung rata-rata, simpangan baku, hingga mencari frekuensi suatu angka. Semua data disimpan dalam array bertipe int.

Masuk ke bagian utama program, pengguna diminta memasukkan jumlah elemen array yang akan disimpan pada variabel N. Kemudian program membuat array bilArr sebanyak N elemen. Pengguna diminta menginput nilai-nilai elemen satu per satu berdasarkan indeks array.

Soal A:

Menampilkan seluruh isi dari array bilArr.

Soal B:

Menampilkan elemen-elemen array yang berada pada indeks ganjil, yaitu indeks 1, 3, 5, dan seterusnya.

Soal C:

Menampilkan elemen-elemen array yang berada pada indeks genap, yaitu indeks 0, 2, 4, dan seterusnya.

Soal D:

Menampilkan elemen array berdasarkan indeks kelipatan dari nilai n yang dimasukkan oleh pengguna. Namun, pengecekan indeks kelipatan dimulai dari indeks lebih dari 0 (indeks 0 tidak termasuk meskipun merupakan kelipatan n).

Soal E:

Pengguna diminta untuk memilih indeks elemen yang ingin dihapus dari array. Program akan menghapus elemen pada indeks tersebut dan menampilkan array setelah penghapusan. Operasi dilakukan menggunakan fungsi `append()` untuk menggabungkan dua bagian array, tanpa elemen yang dihapus.

Soal F:

Program menghitung dan menampilkan rata-rata dari elemen-elemen array setelah penghapusan. Proses dilakukan dengan menjumlahkan semua elemen dan membaginya dengan jumlah total elemen.

Soal G:

Program menghitung simpangan baku dari elemen array menggunakan rumus:
$$\text{Simpangan Baku} = \sqrt{\left(\sum \frac{(x_i - \text{rata-rata})^2}{N}\right)}$$
 rumus ini digunakan untuk mengukur seberapa jauh penyebaran data terhadap nilai rata-ratanya.

Soal H:

Pengguna diminta memasukkan sebuah angka, lalu program akan mencari dan menampilkan frekuensi kemunculan angka tersebut di dalam array.

3. Nomor 3

- Source Code:

```
package main

import (
    "fmt"
)

func main() {
    var club1, club2 string
    var skor1, skor2 int
    var pemenang []string
    DimasRamadhani_103112400065 := 1
    fmt.Print("Klub A: ")
    fmt.Scan(&club1)
    fmt.Print("Klub B: ")
    fmt.Scan(&club2)
    for {
        fmt.Printf("Pertandingan ke-%d : ",
DimasRamadhani_103112400065)
        fmt.Scan(&skor1, &skor2)
        if skor1<0 || skor2<0{
            break
        } else if skor1 > skor2 {
            pemenang = append(pemenang, club1)
        } else if skor1 < skor2{
            pemenang = append(pemenang, club2)
        } else {
            pemenang = append(pemenang, "Draw")
        }
        DimasRamadhani_103112400065++
    }
    for i := 0; i < len(pemenang); i++ {
        fmt.Printf("Hasil %d: %s\n", i+1, pemenang[i])
    }
    fmt.Print("Pertandingan selesai")
}
```

- **Screenshoot Hasil Pembahasan:**

```
PS C:\Users\Dimas\OneDrive\Compuer 2\Algoritma dan Pemrograman>
Klub A: MU
Klub B: Inter
Pertandingan ke-1 : 2 0
Pertandingan ke-2 : 1 2
Pertandingan ke-3 : 2 2
Pertandingan ke-4 : 0 1
Pertandingan ke-5 : 3 2
Pertandingan ke-6 : 1 0
Pertandingan ke-7 : 5 2
Pertandingan ke-8 : 2 3
Pertandingan ke-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
```

- **Penjelasan:**

Program ini bertujuan untuk mencatat hasil beberapa pertandingan antara dua klub sepak bola berdasarkan input skor dari pengguna. Program akan terus menerima input hingga salah satu skor bernilai negatif, yang menjadi tanda bahwa pertandingan telah selesai.

Program mendeklarasikan dua variabel bertipe string yaitu club1 dan club2 untuk menyimpan nama masing-masing klub. Kemudian dua variabel skor1 dan skor2 bertipe integer digunakan untuk menyimpan skor masing-masing klub pada tiap pertandingan. Array pemenang bertipe slice dari string digunakan untuk menyimpan hasil pertandingan berdasarkan pemenangnya. Variabel DimasRamadhani_103112400065 bertipe integer digunakan untuk menandai nomor pertandingan, dimulai dari angka 1.

Pertama, program meminta pengguna memasukkan nama dua klub. Setelah itu, program masuk ke dalam perulangan tak hingga menggunakan for, di mana pada setiap iterasi pengguna diminta memasukkan skor pertandingan dari kedua klub.

Logika pengecekan skor pertandingan:

- Jika salah satu dari skor bernilai negatif, maka perulangan berhenti (berarti pertandingan selesai).
- Jika skor1 > skor2, maka klub pertama (club1) dianggap menang, dan string nama klub tersebut ditambahkan ke dalam array pemenang.
- Jika skor1 < skor2, maka klub kedua (club2) dianggap menang, dan string nama klub tersebut ditambahkan ke array pemenang.
- Jika skor sama, maka hasil dianggap seri dan string "Draw" ditambahkan ke array pemenang.

Setelah perulangan berhenti, program menampilkan hasil dari seluruh pertandingan yang telah diinputkan, dengan mencetak pemenang untuk tiap pertandingan berdasarkan urutan pertandingan.

Akhir program ditutup dengan mencetak "Pertandingan selesai" sebagai penanda bahwa seluruh proses input dan penilaian telah selesai dilakukan.

4. Nomor 4

- Source Code:

```
package main

import "fmt"

const NMAX int = 127

type tabel [NMAX]rune

func isiArray(t *tabel, n *int) {
    var Teks rune
    *n = 0
    fmt.Print("Teks: ")
    for {
        fmt.Scanf("%c", &Teks)
        if Teks == '.' || *n > NMAX {
            break
        }
        if Teks != ' ' && Teks != '\n' {
            (*t)[*n] = Teks
            *n++
        }
    }
}

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

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

/* Dimas Ramadhani
103112400065 */
```

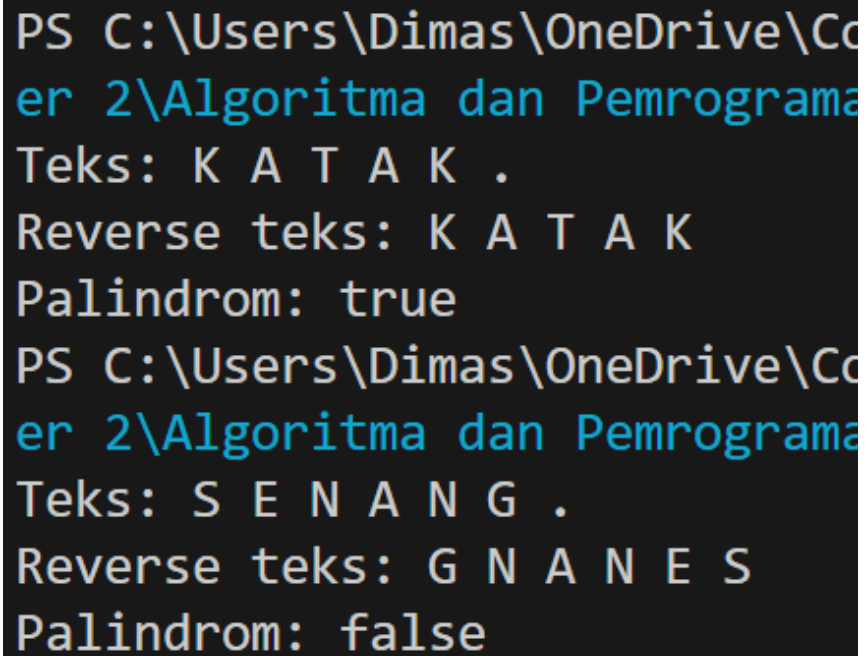
```

func palindrom(t tabel, n int) bool {
    var tabClone tabel
    copy(tabClone[:], t[:])
    balikanArray(&tabClone, n)
    for i := 0; i < n; i++ {
        if t[i] != tabClone[i] {
            return false
        }
    }
    return true
}

func main() {
    var tab tabel
    var m int
    isiArray(&tab, &m)
    balikanArray(&tab, m)
    cetakArray(tab, m)
    fmt.Print("Palindrom: ", palindrom(tab, m))
}

```

- **Screenshoot Hasil Pembahasan:**



```

PS C:\Users\Dimas\OneDrive\Co
er 2\Algoritma dan Pemrograma
Teks: K A T A K .
Reverse teks: K A T A K
Palindrom: true
PS C:\Users\Dimas\OneDrive\Co
er 2\Algoritma dan Pemrograma
Teks: S E N A N G .
Reverse teks: G N A N E S
Palindrom: false

```

- **Penjelasan:**

Program ini bertujuan untuk membaca sebuah teks karakter dari input pengguna, menyimpannya dalam array, kemudian membalik isi array tersebut, mencetaknya, dan mengecek apakah teks tersebut

merupakan palindrom atau bukan. Teks akan terus dibaca sampai ditemukan karakter titik (.) sebagai penanda akhir input. Program mengabaikan spasi dan karakter newline.

Tipe data tabel dideklarasikan sebagai array berdimensi NMAX (sebanyak 127) bertipe rune, yang digunakan untuk menyimpan karakter Unicode dari input teks. Variabel n bertipe integer digunakan untuk menghitung banyaknya karakter yang disimpan dalam array.

Program menggunakan prosedur isiArray untuk membaca karakter satu per satu dari input pengguna. Teks yang dimasukkan disimpan ke dalam array tabel jika bukan spasi (' ') atau newline ('\n'). Jika ditemukan karakter titik (.), maka proses input dihentikan. Prosedur ini juga memastikan bahwa jumlah karakter tidak melebihi kapasitas maksimum array NMAX.

Prosedur cetakArray digunakan untuk mencetak isi array ke layar, dengan menampilkan karakter-karakter dalam urutan tertentu. Format output dimulai dengan "Reverse teks:" diikuti isi array karakter.

Prosedur balikanArray berfungsi untuk membalik urutan elemen-elemen dalam array dari awal hingga akhir. Proses ini dilakukan dengan menukar elemen pertama dengan terakhir, elemen kedua dengan sebelum terakhir, dan seterusnya hingga titik tengah array.

Fungsi palindrom bertipe boolean digunakan untuk mengecek apakah urutan karakter dalam array membentuk palindrom. Fungsi ini membuat salinan dari array asli, lalu membalik array salinan menggunakan balikanArray. Setelah itu, program membandingkan karakter satu per satu dari array asli dengan array yang sudah dibalik. Jika semua karakter sama, maka fungsi akan mengembalikan true, menandakan bahwa teks adalah palindrom. Jika tidak, maka akan mengembalikan false.

Pada bagian program utama (main), variabel tab bertipe tabel dan m bertipe integer dideklarasikan. Program memanggil isiArray untuk membaca input, kemudian membalik isi array dengan balikanArray, mencetak array dengan cetakArray, dan terakhir menampilkan

apakah teks tersebut adalah palindrom atau bukan dengan memanggil fungsi palindrom.

IV. KESIMPULAN

Praktikum ini menggambarkan bagaimana konsep array, slice, dan tipe bentukan bisa digunakan dalam pemrograman untuk menyelesaikan masalah sehari-hari, seperti mengecek posisi titik terhadap lingkaran, mengolah data angka, atau memeriksa apakah sebuah kata merupakan palindrom. Melalui berbagai studi kasus, praktikum ini membantu memperjelas cara kerja logika program dan bagaimana struktur data sederhana bisa sangat berguna dalam berbagai konteks.

V. REFERENSI

Prayogo, N. A. (2024). *Dasar Pemrograman Golang* (Versi 4.0.20240830). Retrieved from <https://github.com/novalagung/dasarpemrogramangolang>

Selly Meliana, S.Kom., M.Kom.(2024) *Modul Praktikum ALGORITMA DAN PEMROGRAMAN 2*.