

**LAPORAN PRAKTIKUM
ALGORITMA DAN PEMROGRAMAN 2**

**MODUL 6
ARRAY DAN STRUCK**



Oleh:

ARIEL AHNAF KUSUMA

103112400050

12IF-01

**S1 TEKNIK INFORMATIKA
TELKOM UNIVERSITY PURWOKERTO
2025**

I. DASAR TEORI

STRUCT

Struct adalah kumpulan definisi variabel (atau property) dan atau fungsi (atau method), yang dibungkus sebagai tipe data baru dengan nama tertentu. Property dalam struct, tipe datanya bisa bervariasi. Mirip seperti map, hanya saja key-nya sudah didefinisikan di awal, dan tipe data tiap itemnya bisa berbeda.

Dari sebuah struct, kita bisa buat variabel baru, yang memiliki atribut sesuai skema struct tersebut. Kita sepakati dalam buku ini, variabel tersebut dipanggil dengan istilah object atau variabel object.

ARRAY

Array adalah kumpulan data bertipe sama, yang disimpan dalam sebuah variabel. Array memiliki kapasitas yang nilainya ditentukan pada saat pembuatan, menjadikan elemen/data yang disimpan di array tersebut jumlahnya tidak boleh melebihi yang sudah dialokasikan. Elemen array bisa dituliskan dalam bentuk horizontal ataupun dalam bentuk vertikal.

Khusus untuk deklarasi array dengan cara vertikal, tanda koma wajib dituliskan setelah setiap elemen (termasuk elemen terakhir), agar tidak memunculkan syntax error.

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".

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.

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.

II. GUIDED

Guided 1

Source code

```
1 package main
2
3 import (
4     "fmt"
5     "time"
6 )
7
8 // Struct untuk barang dalam struk belanja
9 type Item struct {
10     Name    string
11     Price   float64
12     Quantity int
13 }
14
15 // Struct untuk struk belanja
16 type Receipt struct {
17     StoreInfo string
18     Date       time.Time
19     Items      []Item
20     TotalAmount float64
21 }
22
23 // Method untuk menghitung total harga semua item
24 func (r *Receipt) CalculateTotal() {
25     var total float64
26     for _, item := range r.Items {
27         total += item.Price * float64(item.Quantity)
28     }
29     r.TotalAmount = total
30 }
31
32 // Method untuk mencetak struk belanja
33 func (r *Receipt) PrintReceipt() {
34     fmt.Println("=====")
35     fmt.Println(r.StoreInfo)
36     fmt.Println("Tanggal:", r.Date.Format("02-01-2006 15:04"))
37     fmt.Println("=====")
38     fmt.Printf("%-15s %-10s %-8s %-10s\n", "Item", "Harga", "Jumlah", "Total")
39     fmt.Println("=====")
40
41     for _, item := range r.Items {
42         itemTotal := item.Price * float64(item.Quantity)
43         fmt.Printf("%-15s Rp%-9.2f %-8d Rp%-9.2f\n", item.Name, item.Price, item.Quantity, itemTotal)
44     }
45
46     fmt.Println("=====")
47     fmt.Printf("%-35s Rp%-9.2f\n", "Total Belanja:", r.TotalAmount)
48     fmt.Println("=====")
49     fmt.Println("Terima kasih telah berbelanja!")
50 }
51
52 func main() {
53     receipt := Receipt{
54         StoreInfo: "Toko Sembako Makmur\nJl. Raya No. 123, Jakarta",
55         Date:       time.Now(),
56         Items: []Item{
57             {Name: "Beras", Price: 12000, Quantity: 5},
58             {Name: "Gula", Price: 15000, Quantity: 2},
59             {Name: "Minyak", Price: 20000, Quantity: 1},
60             {Name: "Telur", Price: 2000, Quantity: 10},
61         },
62     }
63
64     receipt.CalculateTotal()
65     receipt.PrintReceipt()
66 }
67
```

Output code

```
PS C:\ARIEL AHNAF KUSUMA\103112400050_Modul 6> go run "c:\ARIEL AHNAF KUSUMA\103112400050_Modul 6\main.go"
=====
Toko Sembako Makmur
Jl. Raya No. 123, Jakarta
Tanggal: 25-04-2025 17:16
=====
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!
```

Deskripsi code

Program ini berfungsi sebagai simulasi untuk struk belanja yang menggunakan konsep struct dan method, untuk membangun sistem pencatatan transaksi yang sederhana tetapi efektif.

Guided 2

Source code

```

1 package main
2
3 import (
4     "fmt"
5 )
6
7 func main() {
8     // Deklarasi dan inisialisasi array nilai mahasiswa
9     nilaiMahasiswa := []int{85, 90, 78, 88, 95}
10
11     fmt.Println("Data Nilai Mahasiswa:")
12     fmt.Println("=====")
13
14     // Menampilkan nilai per mahasiswa
15     for i, nilai := range nilaiMahasiswa {
16         fmt.Printf("Mahasiswa %d: %d\n", i+1, nilai)
17     }
18
19     // Menghitung rata-rata nilai
20     var total int
21     for _, nilai := range nilaiMahasiswa {
22         total += nilai
23     }
24     rataRata := float64(total) / float64(len(nilaiMahasiswa))
25
26     fmt.Println("=====")
27     fmt.Printf("Rata-rata nilai: %.2f\n", rataRata)
28
29     // Mencari nilai tertinggi dan terendah
30     tertinggi := nilaiMahasiswa[0]
31     terendah := nilaiMahasiswa[0]
32
33     for _, nilai := range nilaiMahasiswa {
34         if nilai > tertinggi {
35             tertinggi = nilai
36         }
37         if nilai < terendah {
38             terendah = nilai
39         }
40     }
41
42     fmt.Printf("Nilai tertinggi: %d\n", tertinggi)
43     fmt.Printf("Nilai terendah: %d\n", terendah)
44
45     // Contoh array 2 dimensi
46     fmt.Println("\nContoh Array 2 Dimensi:")
47     fmt.Println("=====")
48
49     // Nilai ujian mahasiswa dalam 2 mata kuliah (Matematika, Bahasa)
50     nilaiUjian := [3][2]int{
51         {80, 85},
52         {90, 75},
53         {70, 95},
54     }
55
56     // Menampilkan nilai ujian per mahasiswa
57     fmt.Println("Nilai Ujian Mahasiswa (Matematika, Bahasa):")
58     for i, nilai := range nilaiUjian {
59         fmt.Printf("Mahasiswa %d: Matematika = %d, Bahasa = %d\n", i+1, nilai[0], nilai[1])
60     }
61 }

```

Output code

```

PS C:\ARIEL AHNAF KUSUMA\103112400050_Modul 6> go run "c:\ARIEL AHNAF
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

```

Deskripsi code

Program ini berfungsi untuk mengelola data nilai mahasiswa, menggunakan array satu dimensi dan dua dimensi, dan program diatas akan menunjukan bagaimana caranya

III. UNGUIDED

Unguided 1

Source code

```
1  package main
2
3  //103112400050
4  import (
5      "fmt"
6      "math"
7  )
8
9  type Titik struct{ x, y int }
10 type Lingkaran struct{ x, y, r int }
11
12 func jarak(a, b Titik) float64 {
13     dx := a.x - b.x
14     dy := a.y - b.y
15     return math.Sqrt(float64(dx*dx + dy*dy))
16 }
17
18 func dalam(c Lingkaran, t Titik) bool {
19     return jarak(Titik{c.x, c.y}, t) <= float64(c.r)
20 }
21
22 func main() {
23     var c1, c2 Lingkaran
24     var A Titik
25
26     fmt.Scan(&c1.x, &c1.y, &c1.r)
27     fmt.Scan(&c2.x, &c2.y, &c2.r)
28     fmt.Scan(&A.x, &A.y)
29
30     d1 := dalam(c1, A)
31     d2 := dalam(c2, A)
32
33     if d1 && d2 {
34         fmt.Println("Titik di dalam lingkaran 1 dan 2")
35     } else if d1 {
36         fmt.Println("Titik di dalam lingkaran 1")
37     } else if d2 {
38         fmt.Println("Titik di dalam lingkaran 2")
39     } else {
40         fmt.Println("Titik di luar lingkaran 1 dan 2")
41     }
42 }
```

Output code

```

PS C:\ARIEL AHNAF KUSUMA\103112400050_Modul 6> go run "c:\ARIEL AHNAF KUSUMA\103112400050_Modul 6\unguided1.go"
1 1 5
8 8 4
2 2
Titik di dalam lingkaran 1
PS C:\ARIEL AHNAF KUSUMA\103112400050_Modul 6>
PS C:\ARIEL AHNAF KUSUMA\103112400050_Modul 6> go run "c:\ARIEL AHNAF KUSUMA\103112400050_Modul 6\unguided1.go"
1 2 3
4 5 6
7 8
Titik di dalam lingkaran 2
PS C:\ARIEL AHNAF KUSUMA\103112400050_Modul 6> go run "c:\ARIEL AHNAF KUSUMA\103112400050_Modul 6\unguided1.go"
5 10 15
-15 4 20
0 0
Titik di dalam lingkaran 1 dan 2
PS C:\ARIEL AHNAF KUSUMA\103112400050_Modul 6>
PS C:\ARIEL AHNAF KUSUMA\103112400050_Modul 6> go run "c:\ARIEL AHNAF KUSUMA\103112400050_Modul 6\unguided1.go"
1 1 5
8 8 4
15 20
Titik di luar lingkaran 1 dan 2

```

Deskripsi code

Program tersebut bertujuan untuk memeriksa sebuah titik menggunakan *Struct*, yang apakah titik tersebut berada dalam satu atau dua lingkaran

Unguided 2

Source code

```

1 package main
2
3 //103112400050
4 import (
5     "fmt"
6 )
7
8 func hitungRataRata(Ahnaf []int) float64 {
9     var total int
10    for _, nilai := range Ahnaf {
11        total += nilai
12    }
13    return float64(total) / float64(len(Ahnaf))
14 }
15
16 func hitungSimpanganBaku(Ahnaf []int) float64 {
17     rata := hitungRataRata(Ahnaf)
18     var jumlahSelisihKuadrat float64
19     for _, nilai := range Ahnaf {
20         selisih := float64(nilai) - rata
21         jumlahSelisihKuadrat += selisih * selisih
22     }
23     return jumlahSelisihKuadrat / float64(len(Ahnaf))
24 }
25
26 func hitungFrekuensi(Ahnaf []int, target int) int {
27     count := 0
28     for _, nilai := range Ahnaf {
29         if nilai == target {
30             count++
31         }
32     }
33     return count
34 }
35

```



```

func tampilkanIndeks(Ahnaf []int) {
    fmt.Println("Nilai indeks ganjil: ")
    for i := 1; i < len(Ahnaf); i += 2 {
        fmt.Println(Ahnaf[i], " ")
    }
    fmt.Println()

    fmt.Println("Nilai indeks genap: ")
    for i := 0; i < len(Ahnaf); i += 2 {
        fmt.Println(Ahnaf[i], " ")
    }
    fmt.Println()
}

func main() {
    var jumlahElemen int
    fmt.Println("Masukkan jumlah elemen: ")
    fmt.Scan(&jumlahElemen)

    Ahnaf := make([]int, jumlahElemen)
    for i := 0; i < jumlahElemen; i++ {
        fmt.Println("Masukkan elemen ke-:", i+1)
        fmt.Scan(&Ahnaf[i])
    }

    fmt.Println("Isi array:", Ahnaf)
    tampilkanIndeks(Ahnaf)

    fmt.Printf("Rata-rata: %.2f\n", hitungRataRata(Ahnaf))
    fmt.Printf("Simpangan baku (tanpa akar): %.2f\n", hitungSimpanganBaku(Ahnaf))

    var angkaUntukFrekuensi int
    fmt.Println("Masukkan angka untuk dihitung frekuensinya: ")
    fmt.Scan(&angkaUntukFrekuensi)
    fmt.Printf("Frekuensi angka %d: %d\n", angkaUntukFrekuensi, hitungFrekuensi(Ahnaf, angkaUntukFrekuensi))
}

```

Output code

```

PS C:\ARIEL AHNAF KUSUMA\103112400050_Modul 6> go run "c:\ARIEL AHNAF KUSUMA\103112400050_Modul 6\unguided2.go"
Masukkan jumlah elemen: 5
Masukkan elemen ke-1: 1
Masukkan elemen ke-2: 2
Masukkan elemen ke-3: 3
Masukkan elemen ke-4: 4
Masukkan elemen ke-5: 5

Isi array: [1 2 3 4 5]
Nilai indeks ganjil: 2 4
Nilai indeks genap: 1 3 5
Rata-rata: 3.00
Simpangan baku (tanpa akar): 2.00
Masukkan angka untuk dihitung frekuensinya: 2
Frekuensi angka 2: 1
PS C:\ARIEL AHNAF KUSUMA\103112400050_Modul 6>

```

Deskripsi code

Program diatas bertujuan untuk mengolah lalu menganalisis sebuah Array yang berisikan angka dengan beberapa operasi statistic dasar.

Beberapa fungsi utama program termasuk hitungRataRata elemen elemen array, hitungSimpanganBaku untuk menghitung simpangan baku, atau varians, dari array, dan hitugFrekuensi untuk menghitung berapa kali sebuah angka. muncul dalam array. tampilkanIndeks juga dapat menampilkan elemen yang ada pada indeks ganjil dan genap array.

Unguided 3

Source code

```
unguided3.go > ...
1  package main
2
3  //1033112400050
4  import "fmt"
5
6  func main() {
7      var klubA, klubB string
8      var skorA, skorB int
9      var hasil []string
10     fmt.Print("Klub A: ")
11     fmt.Scanln(&klubA)
12     fmt.Print("Klub B: ")
13     fmt.Scanln(&klubB)
14     pertandingan := 1
15     for {
16         fmt.Printf("Pertandingan %d : ", pertandingan)
17         fmt.Scan(&skorA, &skorB)
18         if skorA < 0 || skorB < 0 {
19             break
20         }
21         if skorA > skorB {
22             hasil = append(hasil, klubA)
23         } else if skorB > skorA {
24             hasil = append(hasil, klubB)
25         } else {
26             hasil = append(hasil, "Draw")
27         }
28         pertandingan++
29     }
30     for i, v := range hasil {
31         if v == "Draw" {
32             fmt.Printf("Hasil %d : Draw\n", i+1)
33         } else {
34             fmt.Printf("Hasil %d : %s\n", i+1, v)
35         }
36     }
37     fmt.Println("pertandingan selesai.")
38 }
```

Output code

```
PS C:\ARIEL AHNAF KUSUMA\103112400050_Modul 6> go run "c:\A
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
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.
PS C:\ARIEL AHNAF KUSUMA\103112400050_Modul 6>
```

Deskripsi code

Program diatas berfungsi untuk mencatat lalu menampilkan hasil dari pertandingan antar dua klub sepak bola, saat skor dimasukan program akan membandingkan nilai klub A dan klub B

Unguided 4

Source code

```

1 // 103112400050
2 package main
3
4 import (
5     "fmt"
6 )
7
8 const NMAX int = 127
9
10 type tabel [NMAX]rune
11
12 func isiArray(t *tabel, n *int) {
13     var karakter rune
14     *n = 0
15     fmt.Print("Teks : ")
16     for {
17         fmt.Scanf("%c", &karakter)
18         if karakter == '.' || *n >= NMAX {
19             break
20         }
21         if karakter != ' ' && karakter != '\n' && karakter != '\r' {
22             t[*n] = karakter
23             *n++
24         }
25     }
26 }
27
28 func cetakArray(t tabel, n int) {
29     for i := 0; i < n; i++ {
30         fmt.Printf("%c ", t[i])
31     }
32     fmt.Println()
33 }
34
35 func balikanArray(t *tabel, n int) {
36     for i := 0; i < n/2; i++ {
37         temp := t[i]
38         t[i] = t[n-1-i]
39         t[n-1-i] = temp
40     }
41 }
42
43 func palindrom(t tabel, n int) bool {
44     var salin tabel
45     for i := 0; i < n; i++ {
46         salin[i] = t[i]
47     }
48     balikanArray(&salin, n)
49     for i := 0; i < n; i++ {
50         if t[i] != salin[i] {
51             return false
52         }
53     }
54     return true
55 }
56
57 func main() {
58     var teks tabel
59     var jumlah int
60     isiArray(&teks, &jumlah)
61     fmt.Print("Teks : ")
62     cetakArray(teks, jumlah)
63     if palindrom(teks, jumlah) {
64         fmt.Println("Palindrom ? true")
65     } else {
66         fmt.Println("Palindrom ? false")
67     }
68 }

```

Output code

```
PS C:\ARIEL AHNAF KUSUMA\103112400050_Modul 6> go run "c:\ARIEL AHNAF KUSUMA\103112400050_Modul 6\unguided4.go"
Teks : K A T A K.
Teks : K A T A K
Palindrom ? true
PS C:\ARIEL AHNAF KUSUMA\103112400050_Modul 6> go run "c:\ARIEL AHNAF KUSUMA\103112400050_Modul 6\unguided4.go"
Teks : S E N A N G.
Teks : S E N A N G
Palindrom ? false
PS C:\ARIEL AHNAF KUSUMA\103112400050_Modul 6> █
```

Deskripsi code

Program diatas berfungsi untuk memanfaatkan array bertipe rune untuk menentukan apakah sebuah teks merupakan palindrom.

IV. KESIMPULAN

Hasil dari laporan praktikum ini adalah bahwa pemrograman dengan menggunakan Struct dan Array sangat bermanfaat untuk mengelola data yang terstruktur dan kompleks. Struktur memungkinkan kita untuk

menggabungkan data yang memiliki hubungan khusus dalam satu unit, seperti yang ditunjukkan oleh contoh aplikasi struk belanja yang menggunakan Struct untuk mendefinisikan item dan struk belanja. Di sisi lain, Array memungkinkan penyimpanan data dalam bentuk koleksi dengan ukuran tetap, sedangkan Slice dan Map menawarkan fleksibilitas yang lebih besar, seperti yang ditunjukkan oleh aplikasi pengelolaan data nilai siswa yang menggunakan Array dan Map dua dimensi. Praktik ini menunjukkan bahwa pemahaman tentang penggunaan Struct dan Array dalam bahasa pemrograman Go sangat penting untuk mengelola data dengan efisiensi, melakukan analisis, dan mempermudah implementasi berbagai jenis algoritma dalam pemrograman.

V. REFERENSI

Dasar pemrograman golang A.24. Struct – Dasar Pemrograman Golang by
Noval Agung

Dasar pemrograman golang A.15. Array – Dasar pemrograman Golang by
Noval Agung

Modul 6 Algoritma Pemrograman 2 Telkom University