

**LAPORAN PRAKTIKUM
ALGORITMA DAN PEMROGRAMAN 2**

**MODUL 7
STRUCK & ARRAY**



Disusun Oleh :

PRIESTY AMEILIANA MAULIDAH / 2311102175

IF-11-05

Dosen Pengampu :

ARIF AMRULLOH, S.KOM.,M.KOM

PROGRAM STUDI S1 TEKNIK INFORMATIKA

FAKULTAS INFORMATIKA

TELKOM UNIVERSITY PURWOKERTO

2024

1.) DASAR TEORI

1. tipe bentuk

Tipe bentuk memungkinkan pemrograman untuk mendefinisikan suatu tipe data baru pada suatu bahasa pemrograman.

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 diubah dengan nama alias "bilangan".

Struct atau record

Structure memungkinkan pemrograman untuk mengelompokkan beberapa data atau nilai yang memiliki relasi atau keterkaitan tertentu menjadi suatu kesatuan.

2. Array

Array merupakan ukuran (jumlah elemen) yang tetap (statis) selama eksekusi program, sehingga jumlah elemen array menjadi bagian dari deklarasi variabel dengan tipe array.

Slice(array dinamik)

Array dalam go juga dapat mempunyai ukuran yang dinamik.

Map

Tipe array lain, sebuah array dinamik. indeksnya (disini kunci) tidak harus berbentuk integer.

2.) GUIDED

Soal Studi Case

1)

Sourcecode

```
modul 7 > -60 guided 1.go > ...
1 package main
2 import "fmt"
3 type waktu struct {
4     jam, menit, detik int
5 }
6
7 func main(){
8     var wParkir, wPulang, durasi waktu // awal = 2 45 14
9     var dParkir, dPulang, lParkir int // pulang = 4 30 12
10
11     fmt.Scan(&wParkir.jam, &wParkir.menit, &wParkir.detik)
12     fmt.Scan(&wPulang.jam, &wPulang.menit, &wPulang.detik)
13
14     dParkir = wParkir.detik + wParkir.menit*60 + wParkir.jam*3600 // konversimke detik
15     dPulang = wPulang.detik + wPulang.menit*60 + wPulang.jam*3600 // detik
16     lParkir = dPulang - dParkir // detik dari pulang-datang
17
18     durasi.jam = lParkir / 3600
19     durasi.menit = lParkir % 3600 / 60
20     durasi.detik = lParkir % 3600 % 60 // 17
21     fmt.Printf("Lama parkir: %d jam %d menit %d detik", durasi.jam, durasi.menit, durasi.detik)
22 }
```

Screenshoot Output

```
PROBLEMS 77 OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS D:\1-Priesty AM\alpro 3 semester 3> go run "d:\1-Priesty AM\alpro 3 semester 3\modul 7\guided 1.go"
2 45 14
4 30 12
Lama parkir: 1 jam 44 menit 58 detik
PS D:\1-Priesty AM\alpro 3 semester 3> |
```

Deskripsi Program

Program menghitung lama waktu parkir

- Input : waktu masuk dan keluar (jam menit detik)
- Proses : konversi ke detik, hitung selisih, konversi balik ke format jam-menit-detik
- Output : menampilkan durasi parkir

Hasil output :

- Masuk : 2 45 14
- Keluar : 4 30 12
- Program hitung : selisih waktu dalam format jam : menit : detik

GUIDED

Soal Studi Case

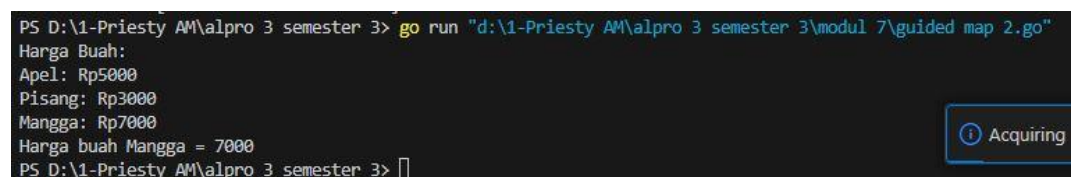
2.

Sourcecode



```
modul 7 > guided map 2.go > ...
1 package main
2 import (
3     "fmt"
4 )
5
6 func main() {
7     // Membuat map dengan nama buah sebagai kunci dan harga sebagai nilai
8     hargaBuah := map[string]int{
9         "Apel": 5000,
10        "Pisang": 3000,
11        "Mangga": 7000,
12    }
13
14    // Menampilkan harga dari setiap buah
15    fmt.Println("Harga Buah:")
16    for buah, harga := range hargaBuah {
17        fmt.Printf("%s: Rp%d\n", buah, harga)
18    }
19
20    fmt.Print("Harga buah Mangga = ", hargaBuah["Mangga"])
21 }
```

Screenshoot Output



```
PS D:\1-Priesty AM\alpro 3 semester 3> go run "d:\1-Priesty AM\alpro 3 semester 3\modul 7\guided map 2.go"
Harga Buah:
Apel: Rp5000
Pisang: Rp3000
Mangga: Rp7000
Harga buah Mangga = 7000
PS D:\1-Priesty AM\alpro 3 semester 3>
```

Deskripsi Program

Program mengelola daftar harga buah

- Data : menyimpan nama dan harga buah dalam map
- Proses : menampilkan semua harga buah dengan perulangan
- Output : menampilkan daftar lengkap dan harga spesifik buah mangga

Data harga buah :

- Apel = Rp5.000
- Pisang = Rp3.000

- Mangga = Rp7.000

GUIDED

Soal Studi Case

3.

Sourcecode

```
modul 7 > go guided slice 2.go > main
1  package main
2
3  import (
4      "fmt"
5  )
6
7  // Fungsi untuk mengecek apakah nama sudah ada di dalam slice
8  func sudahAda(daftarTeman []string, nama string) bool {
9      for _, teman := range daftarTeman {
10         if teman == nama {
11             return true
12         }
13     }
14     return false
15 }
16
17 func main() {
18     // Slice awal untuk daftar teman dengan beberapa data
19     daftarTeman := []string{"Andi", "Budi", "Cici"}
20
21     // Nama-nama baru yang ingin ditambahkan
22     namaBaru := []string{"Dewi", "Budi", "Eka"}
23
24     // Menambahkan nama baru hanya jika belum ada di daftar
25     for _, nama := range namaBaru {
26         if !sudahAda(daftarTeman, nama) {
27             daftarTeman = append(daftarTeman, nama)
28         } else {
29             fmt.Println("Nama", nama, "sudah ada dalam daftar.")
30         }
31     }
32
33     // Menampilkan daftar teman akhir
34     fmt.Println("Daftar Teman:", daftarTeman)
35 }
36
```

Screenshoot Output

```
PS D:\1-Priesty AM\alpro 3 semester 3> go run "d:\1-Priesty AM\alpro 3 semester 3\modul 7\guided slice 2.go"
Nama Budi sudah ada dalam daftar.
Daftar Teman: [Andi Budi Cici Dewi Eka]
PS D:\1-Priesty AM\alpro 3 semester 3> 
```

Deskripsi Program

Program mengelola daftar nama

- Input : daftar nama awal (andi, budi, cici) dan nama baru (dewi, budi, eka)
- Proses : mengecek setiap nama baru, hanya menambahkan yang belum ada
- Output : pesan jika nama duplikasi dan daftar final tanpa duplkasi

Cara kerja : program memeriksa setiap nama baru, menambahkannya jika unik, dan memberikan peingatan jika nama sudah ada dalam daftar

Contoh : budi tidak ditambahkan karena sudah ada, dewi dan eka ditambahkan karena baru

3.) UNGUIDED

Modul 7

Soal Studi Case

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 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 di dalam lingkaran 1", "Titik di dalam lingkaran 2", atau "Titik di luar lingkaran 1 dan 2".

Contoh

No	Masukan	Keluaran
1	1 1 5 8 8 4 2 2	Titik di dalam lingkaran 1
2	1 2 3 4 5 6 7 8	Titik di dalam lingkaran 2
3	5 10 15 -15 4 20 0 0	Titik di dalam lingkaran 1 dan 2
4	1 1 5 8 8 4 15 20	Titik di luar lingkaran 1 dan 2

Sourcecode

```

modul 7 > unguided array 1.go > main
1  package main
2
3  import (
4      "fmt"
5      "math"
6  )
7
8  type Titik struct {
9      x, y int
10 }
11
12 type Lingkaran struct {
13     pusat Titik
14     radius int
15 }
16
17 func jarak(p, q Titik) float64 {
18     return math.Sqrt(float64((p.x-q.x)*(p.x-q.x) + (p.y-q.y)*(p.y-q.y)))
19 }
20
21 func didalam(c Lingkaran, p Titik) bool {
22     return jarak(c.pusat, p) < float64(c.radius)
23 }
24
25 func main() {
26     var l1, l2 Lingkaran
27     var p Titik
28
29     fmt.Scan(&l1.pusat.x, &l1.pusat.y, &l1.radius)
30     fmt.Scan(&l2.pusat.x, &l2.pusat.y, &l2.radius)
31     fmt.Scan(&p.x, &p.y)
32
33     inLingkaran1 := didalam(l1, p)
34     inLingkaran2 := didalam(l2, p)
35
36     if inLingkaran1 && inLingkaran2 {
37         fmt.Println("Titik di dalam lingkaran 1 dan 2")
38     } else if inLingkaran1 {
39         fmt.Println("Titik di dalam lingkaran 1")
40     } else if inLingkaran2 {
41         fmt.Println("Titik di dalam lingkaran 2")
42     } else {
43         fmt.Println("Titik di luar lingkaran 1 dan 2")
44     }
45 }

```

Screenshoot Output


```

Titik di luar lingkaran 1 dan 2
PS D:\1-Priesty AM\alpro 3 semester 3> go run "d:\1-Priesty AM\alpro 3 semester 3\modul 7\unguided 1.go"
3\modul 7\unguided 1.go"
1 1 5
8 8 4
2 2
Titik di dalam lingkaran 1
PS D:\1-Priesty AM\alpro 3 semester 3> go run "d:\1-Priesty AM\alpro 3 semester 3\modul 7\unguided 1.go"
1 2 3
4 5 6
7 8
Titik di dalam lingkaran 2
PS D:\1-Priesty AM\alpro 3 semester 3> go run "d:\1-Priesty AM\alpro 3 semester 3\modul 7\unguided 1.go"
5 10 15
-15 4 20
0 0
Titik di dalam lingkaran 1 dan 2
PS D:\1-Priesty AM\alpro 3 semester 3> go run "d:\1-Priesty AM\alpro 3 semester 3\modul 7\unguided 1.go"
1 1 5
8 8 4
15 20
Titik di luar lingkaran 1 dan 2
PS D:\1-Priesty AM\alpro 3 semester 3>

```

Deskripsi Program

Input program :

- Lingkaran 1 -> pusat (x,y) & radius
- Lingkaran 2-> pusat(x,y)& radius
- Titik yang dicek -> koordinat(x,y)

Program menggunakan rumus Pythagoras untuk menghitung jarak titik ke pusat lingkaran :

1. Input: L1(1,1,5), L2(8,8,4), T(2,2) Output: "Titik di dalam lingkaran 1"
2. Input: L1(1,2,3), L2(4,5,6), T(7,8) Output: "Titik di dalam lingkaran 2"
3. Input: L1(5,10,15), L2(-15,4,20), T(0,0) Output: "Titik di dalam lingkaran 1 dan 2"
4. Input: L1(1,1,5), L2(8,8,4), T(15,20) Output: "Titik di luar lingkaran 1 dan 2"

UNGUIDED

Soal Studi Case

2.) 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 indek ke-0 adalah genap).
- d. Menampilkan elemen-elemen array dengan indeks kelipatan bilangan x. x bisa diperoleh dari masukan pengguna.
- 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

```
modul 7 > -go unguided array 2.go > main
1 package main
2
3 import (
4     "fmt"
5     "math"
6 )
7
8 func main() {
9     var N int
10    fmt.Print("Masukkan jumlah elemen array: ")
11    fmt.Scan(&N)
12
13    arr := make([]int, N)
14    for i := 0; i < N; i++ {
15        fmt.Printf("Masukkan elemen ke-%d: ", i)
16        fmt.Scan(&arr[i])
17    }
18
19    fmt.Println("Isi keseluruhan array:", arr)
20
21    fmt.Print("Elemen dengan indeks ganjil: ")
22    for i := 1; i < N; i += 2 {
23        fmt.Print(arr[i], " ")
24    }
25    fmt.Println()
26
27    fmt.Print("Elemen dengan indeks genap: ")
28    for i := 0; i < N; i += 2 {
29        fmt.Print(arr[i], " ")
30    }
31    fmt.Println()
32
33    var x int
```

Screenshoot Output

```
PROBLEMS 77 OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS D:\1-Priesty AM\alpro 3 semester 3> go run "d:\1-Priesty AM\alpro 3 semester 3\modul 7\unguided array 2.go"
Masukkan jumlah elemen array: 20
Masukkan elemen ke-0: 1
Masukkan elemen ke-1: 2
Masukkan elemen ke-2: 3
Masukkan elemen ke-3: 4
Masukkan elemen ke-4: 5
Masukkan elemen ke-5: 6
Masukkan elemen ke-6: 7
Masukkan elemen ke-7: 8
Masukkan elemen ke-8: 9
Masukkan elemen ke-9: 10
Masukkan elemen ke-10: 11
Masukkan elemen ke-11: 12
Masukkan elemen ke-12: 13
Masukkan elemen ke-13: 14
Masukkan elemen ke-14: 15
Masukkan elemen ke-15: 16
Masukkan elemen ke-16: 17
Masukkan elemen ke-17: 18
Masukkan elemen ke-18: 19
Masukkan elemen ke-19: 20
Isi keseluruhan array: [1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20]
Elemen dengan indeks ganjil: 2 4 6 8 10 12 14 16 18 20
Elemen dengan indeks genap: 1 3 5 7 9 11 13 15 17 19
Masukkan bilangan x untuk indeks kelipatan: 13
Elemen dengan indeks kelipatan 13: 1 14
Masukkan indeks yang ingin dihapus: 18
Isi array setelah penghapusan: [1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 20]
Rata-rata: 10.052631578947368
Standar deviasi: 5.56751559646648
Masukkan bilangan untuk frekuensi: 10
Frekuensi dari bilangan 10: 1
```

Deskripsi Program

1. Input :

- User memasukkan jumlah elemen (N) dan nilai-nilai array

2. Proses :

- Program menampilkan elemen berdasarkan indeks (ganjil/genap/kelipatan)
- Dapat menghapus elemen pada indeks tertentu
- Menghitung statistik dasar (rata-rata, standar deviasi)
- Mencari frekuensi kemunculan angka

3. Cara kerja

Input N -> Buat & isi array -> Tampilkan elemen berdasar indeks ->
Hapus elemen -> Hitung statistik -> Hitung frekuensi

4. Output

- Isi array (sebelum & sesudah penghapusan)
- Hasil filter indeks (ganjil/genap/ kelipatan)
- Hasil penghitung statistik

- Jumlah kemunculan angka tertentu

UNGUIDED

Soal Studi Case

3. Buatlah program yang mengimplementasikan rekursif untuk menampilkan faktor bilangan dari suatu N, atau bilangan yang apa saja yang habis membagi N. Masukan terdiri dari sebuah bilangan bulat positif N. Keluaran terdiri dari barisan bilangan yang menjadi faktor dari N (terurut dari 1 hingga N ya). Contoh masukan dan keluaran:

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

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

Screenshoot Output

```
PROBLEMS 77 OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS D:\1-Priesty AM\alpro 3 semester 3> go run "d:\1-Priesty AM\alpro 3 semester 3\modul 7\unguided array 3.go"
Klub A: mu
Klub B: inter
Pertandingan 1: 2 0
Hasil 1: mu
Pertandingan 2: 1 2
Hasil 2: inter
Pertandingan 3: 2 2
Hasil 3: Draw
Pertandingan 4: 0 1
Hasil 4: inter
Pertandingan 5: 3 2
Hasil 5: mu
Pertandingan 6: 1 0
Hasil 6: mu
Pertandingan 7: 5 2
Hasil 7: mu
Pertandingan 8: 2 3
Hasil 8: inter
Pertandingan 9: -1 2
Pertandingan selesai
PS D:\1-Priesty AM\alpro 3 semester 3> |
```

Deskripsi Program

1. Fungsi utama

- Mencatat hasil perbandingan 2 klub
- Menyimpan riwayat pertandingan
- Berhenti saat skor negatif diinput

2. Cara kerja

Input nama klub -> Input skor -> Tentukan pemenang -> Simpan hasil -> Ulangi sampai skor negatif -> Tampilkan total pertandingan

3. Aturan skor :

- Skor tim A > tim B = tim A menang
- Skor tim B > tim A = tim B menang
- Skor sama = draw (seri)

4. MU vs Inter :

- 8 perbandingan total
- MU : 4 kemenangan
- Inter : 3 kemenangan
- Draw : 1 kemenangan
- Program berhenti di perbandingan 9 (input -1)

UNGUIDED

Soal Studi Case

4. Sebuah array digunakan untuk menampung sekumpulan karakter, Anda diminta untuk membuat sebuah subprogram untuk melakukan membalikkan urutan isi array dan memeriksa apakah membentuk palindrom.

Sourcecode

```
modul 7 > -GO unguided array 4.go > main
1 package main
2
3 import (
4     "fmt"
5     "strings"
6 )
7
8 func reverseArray(arr []rune) []rune {
9     n := len(arr)
10    for i := 0; i < n/2; i++ {
11        arr[i], arr[n-i-1] = arr[n-i-1], arr[i]
12    }
13    return arr
14 }
15
16 func palindrom(arr []rune) bool {
17     reversed := reverseArray(append([]rune{}, arr...))
18     return string(arr) == string(reversed)
19 }
20
21 func main() {
22     var input string
23     fmt.Print("Teks: ")
24     fmt.Scanln(&input)
25
26     arr := []rune(strings.ReplaceAll(input, " ", ""))
27     reversed := reverseArray(append([]rune{}, arr...))
28
29     fmt.Printf("Reverse teks: %s\n", string(reversed))
30     fmt.Printf("Palindrom? %t\n", palindrom(arr))
31 }
```

Screenshoot Output

```

PS D:\1-Priesty AM\alpro 3 semester 3> go run "d:\1-Priesty AM\alpro 3 semester 3\modul 7\unguided array 4.go"
Teks: senang
Reverse teks: gnanes
Palindrom? false
PS D:\1-Priesty AM\alpro 3 semester 3> go run "d:\1-Priesty AM\alpro 3 semester 3\modul 7\unguided array 4.go"
Teks: katak
Reverse teks: katak
Palindrom? true
PS D:\1-Priesty AM\alpro 3 semester 3> go run "d:\1-Priesty AM\alpro 3 semester 3\modul 7\unguided array 4.go"
Teks: katak
Reverse teks: katak
Palindrom? true
PS D:\1-Priesty AM\alpro 3 semester 3> go run "d:\1-Priesty AM\alpro 3 semester 3\modul 7\unguided array 4.go"
Teks: senang
Reverse teks: gnanes
Palindrom? false

```

Deskripsi Program

1. Inti program :

- Mengecek palindrom
- Membalik urutan karakter
- Menghapus spasi

2. cara kerja

Input "senang" -> Balik jadi "gnanes" -> Bandingkan -> Bukan palindrom (false)

Input "katak" -> Balik jadi "katak" -> Bandingkan -> Adalah palindrom (true)

3. fungsi utama :

- reverseArray : membalik urutan karakter
- palindrom : membandingkan teks asli & terbalik
- replaceAll : menghasilkan spasi

4. output :

Input: "senang" - Reverse: gnanes - Palindrom? false (karena berbeda)

Input: "katak" - Reverse: katak - Palindrom? true (karena sama)