

LAPORAN PRAKTIKUM
Algoritma Pemrograman

MODUL 10

ELSE-IF



Disusun oleh:

Raga Prasetyo

109082500028

S1IF-13-04

PROGRAM STUDI S1 INFORMATIKA

FAKULTAS INFORMATIKA

TELKOM UNIVERSITY PURWOKERTO

2025

LATIHAN KELAS – GUIDED

1. Guided 1

Source Code

```
package main

import "fmt"

func main() {

    var usia int

    var kk bool

    fmt.Scan(&usia, &kk)

    if usia >= 17 && kk {

        fmt.Println("bisa membuat KTP")

    }else{

        fmt.Println("belum bisa membuat KTP")

    }

}
```

Screenshoot program

```
alpro_10.go > guided_1.go
1  package main
2  import "fmt"
3
4  func main() {
5      var usia int
6      var kk bool
7
8      fmt.Scan(&usia, &kk)
9
10     if usia >= 17 && kk {
11         fmt.Println("bisa membuat KTP")
12     }else{
13         fmt.Println("belum bisa membuat KTP")
14     }
15 }
16
17
18
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

Open file in editor (ctrl + click)

```
PS C:\Users\HP\.vscode> go run "c:\Users\HP\.vscode\alpro_10.go\guided_1.go"
17 true
bisa membuat KTP
PS C:\Users\HP\.vscode> go run "c:\Users\HP\.vscode\alpro_10.go\guided_1.go"
20 false
belum bisa membuat KTP
PS C:\Users\HP\.vscode> go run "c:\Users\HP\.vscode\alpro_10.go\guided_1.go"
15 true
belum bisa membuat KTP
PS C:\Users\HP\.vscode>
```

Deskripsi program

Program Go di atas dibuat untuk menentukan apakah seseorang sudah memenuhi syarat untuk membuat KTP (Kartu Tanda Penduduk). Program meminta dua input dari pengguna: usia (tipe integer) dan kk (tipe boolean), yang menunjukkan apakah orang tersebut memiliki Kartu Keluarga (KK). Setelah menerima input, program melakukan pengecekan menggunakan pernyataan if. Syarat seseorang dapat membuat KTP adalah usia minimal 17 tahun dan memiliki KK. Jika kedua kondisi tersebut terpenuhi, program akan menampilkan pesan "bisa membuat KTP". Jika salah satu atau kedua syarat tidak terpenuhi, program akan menampilkan "belum bisa membuat KTP". Program ini membantu menilai kelayakan pembuatan KTP secara sederhana berdasarkan dua parameter utama: usia dan kepemilikan KK.

2. Guided 2

Source Code

```
package main

import "fmt"

func main() {

    var x rune

    var huruf, vKecil, vBesar bool

    fmt.Scanf("%c", &x)

    huruf = (x >= 'a' && x <= 'z') || (x >= 'A' && x <=
'Z')

    vKecil = x == 'a' || x == 'i' || x == 'u' || x == 'e'
|| x == 'o'

    vBesar = x == 'A' || x == 'I' || x == 'U' || x == 'E'
|| x == 'O'

    if huruf && (vKecil || vBesar) {

        fmt.Println("vokal")

    }else if huruf && !(vKecil || vBesar){

        fmt.Println("konsonan")


    }else{

        fmt.Println("bukan huruf")

    }

}
```

Screenshoot program

```
alpro_10.go >  guided_2.go
1  package main
2  import "fmt"
3
4  func main() {
5      var x rune
6      var huruf, vKecil, vBesar bool
7
8      fmt.Scanf("%c", &x)
9
10     huruf = (x >= 'a' && x <= 'z') || (x >= 'A' && x <= 'Z')
11     vKecil = x == 'a' || x == 'i' || x == 'u' || x == 'e' || x == 'o'
12     vBesar = x == 'A' || x == 'I' || x == 'U' || x == 'E' || x == 'O'
13     if huruf && (vKecil || vBesar) {
14         fmt.Println("vokal")
15     } else if huruf && !(vKecil || vBesar) {
16         fmt.Println("konsonan")
17     } else {
18         fmt.Println("bukan huruf")
19     }
20 }
21
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\HP\.vscode> go run "c:\Users\HP\.vscode\alpro_10.go\guided_2.go"
A
vokal
PS C:\Users\HP\.vscode> go run "c:\Users\HP\.vscode\alpro_10.go\guided_2.go"
f
konsonan
PS C:\Users\HP\.vscode> go run "c:\Users\HP\.vscode\alpro_10.go\guided_2.go"
1
bukan huruf
PS C:\Users\HP\.vscode> go run "c:\Users\HP\.vscode\alpro_10.go\guided_2.go"
$
bukan huruf
PS C:\Users\HP\.vscode> |
```

Deskripsi program

Program Go ini berfungsi untuk mengidentifikasi jenis sebuah karakter yang diinputkan oleh pengguna—apakah karakter tersebut merupakan huruf vokal, huruf konsonan, atau bukan huruf. Program membaca satu karakter menggunakan `fmt.Scanf("%c", &x)` kemudian melakukan beberapa pemeriksaan logis. Pertama, program mengecek apakah karakter tersebut merupakan huruf dengan memeriksa apakah karakter berada dalam rentang 'a' hingga 'z' atau 'A' hingga 'Z'. Selanjutnya, karakter dibandingkan dengan daftar huruf vokal baik dalam bentuk kecil (a, i, u, e, o) maupun besar (A, I, U, E, O). Jika karakter adalah huruf dan termasuk dalam kategori vokal, program mencetak "vokal". Jika karakter adalah huruf tetapi bukan vokal, program mencetak "konsonan". Jika karakter tidak termasuk huruf alfabet, program mencetak "bukan huruf". Program

ini bermanfaat untuk memahami pengolahan karakter dan penggunaan operator logika dalam Go.

3. Guided 3

Source Code

```
package main

import "fmt"

func main() {

    var bilangan, d1, d2, d3, d4 int

    var teks string

    fmt.Print("Bilangan: ")

    fmt.Scan(&bilangan)

    d4 = bilangan % 10

    d3 = (bilangan % 100) / 10

    d2 = (bilangan % 1000) / 10

    d1 = bilangan / 1000

    if d1 < d2 && d2 < d3 && d3 < d4 {

        teks = "terurut membesar"

    }else if d1 > d2 && d2 > d3 && d3 > d4{

        teks = "terurut mengecil"

    }else{

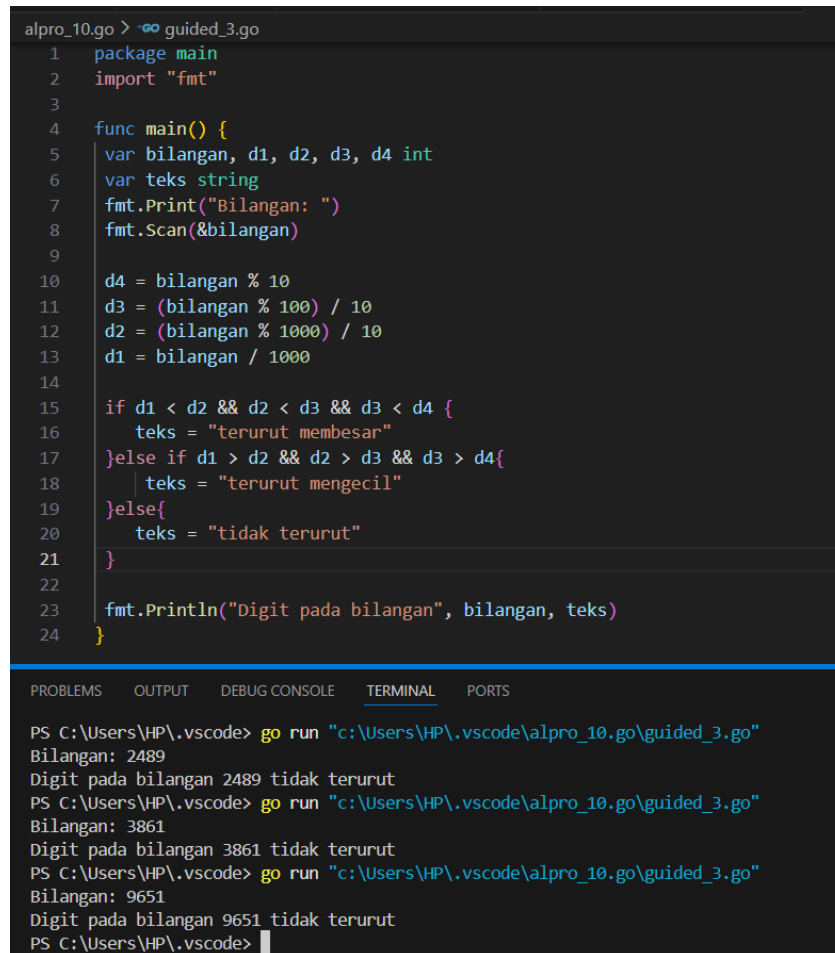
        teks = "tidak terurut"

    }

}
```

```
        fmt.Println("Digit pada bilangan", bilangan, teks)
    }
}
```

Screenshoot program



```
alpro_10.go > guided_3.go
1  package main
2  import "fmt"
3
4  func main() {
5      var bilangan, d1, d2, d3, d4 int
6      var teks string
7      fmt.Print("Bilangan: ")
8      fmt.Scan(&bilangan)
9
10     d4 = bilangan % 10
11     d3 = (bilangan % 100) / 10
12     d2 = (bilangan % 1000) / 10
13     d1 = bilangan / 1000
14
15     if d1 < d2 && d2 < d3 && d3 < d4 {
16         teks = "terurut membesar"
17     } else if d1 > d2 && d2 > d3 && d3 > d4 {
18         teks = "terurut mengecil"
19     } else {
20         teks = "tidak terurut"
21     }
22
23     fmt.Println("Digit pada bilangan", bilangan, teks)
24 }
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\HP\.vscode> go run "c:\Users\HP\.vscode\alpro_10.go\guided_3.go"
Bilangan: 2489
Digit pada bilangan 2489 tidak terurut
PS C:\Users\HP\.vscode> go run "c:\Users\HP\.vscode\alpro_10.go\guided_3.go"
Bilangan: 3861
Digit pada bilangan 3861 tidak terurut
PS C:\Users\HP\.vscode> go run "c:\Users\HP\.vscode\alpro_10.go\guided_3.go"
Bilangan: 9651
Digit pada bilangan 9651 tidak terurut
PS C:\Users\HP\.vscode> 
```

Deskripsi program

Program Go ini berfungsi untuk menentukan apakah empat digit dari sebuah bilangan bulat tersusun secara membesar, mengecil, atau tidak terurut. Setelah pengguna memasukkan sebuah bilangan empat digit, program memisahkan setiap digitnya menggunakan operasi modulo (%) dan pembagian. Digit satuan disimpan sebagai d4, puluhan sebagai d3, ratusan sebagai d2, dan ribuan sebagai d1. Selanjutnya, program membandingkan urutan digit-digit tersebut. Jika $d1 < d2 < d3 < d4$, maka digit dianggap terurut membesar. Jika $d1 > d2 > d3 > d4$, maka dianggap terurut mengecil. Jika tidak memenuhi kedua kondisi tersebut, maka digit dinyatakan tidak terurut. Hasil analisis

kemudian ditampilkan dalam bentuk kalimat yang menyatakan kondisi urutan digit pada bilangan yang dimasukkan. Program ini melatih pemahaman tentang operasi aritmetika sederhana dan logika perbandingan dalam pemrograman.

TUGAS

1. Tugas 1

Source code

```
package main

import "fmt"

func main() {

    var berat, kg, sisa int

    var biayaUtama, biayaTambahan, totalBiaya int

    fmt.Print("Berat parsel (gram): ")

    fmt.Scan(&berat)

    kg = berat / 1000

    sisa = berat % 1000

    biayaUtama = kg * 10000

    // Semua kondisi menggunakan else-if

    if kg > 10 {

        // Jika lebih dari 10 kg, sisa digratiskan
```



```
        biayaTambahan = 0

    } else if sisa >= 500 {

        // Sisa >= 500

        biayaTambahan = sisa * 5

    } else {

        // Sisa < 500

        biayaTambahan = sisa * 15

    }

    totalBiaya = biayaUtama + biayaTambahan

    fmt.Printf("Detail berat: %d kg + %d gr\n", kg, sisa)

    fmt.Printf("Detail biaya: Rp. %d + Rp. %d\n",
biayaUtama, biayaTambahan)

    fmt.Printf("Total biaya: Rp. %d\n", totalBiaya)

}
```

Screenshoot program

```
alpro_10.go > go soal_1.go
1 package main
2 import "fmt"
3
4 func main() {
5     var berat, kg, sisa int
6     var biayaUtama, biayaTambahan, totalBiaya int
7
8     fmt.Print("Berat parsel (gram): ")
9     fmt.Scan(&berat)
10
11     kg = berat / 1000
12     sisa = berat % 1000
13
14     biayaUtama = kg * 10000
15
16     if kg > 10 {
17         biayaTambahan = 0
18     } else if sisa >= 500 {
19         biayaTambahan = sisa * 5
20     } else {
21         biayaTambahan = sisa * 15
22     }
23
24     totalBiaya = biayaUtama + biayaTambahan
25
26     fmt.Printf("Detail berat: %d kg + %d gr\n", kg, sisa)
27     fmt.Printf("Detail biaya: Rp. %d + Rp. %d\n", biayaUtama, biayaTambahan)
28     fmt.Printf("Total biaya: Rp. %d\n", totalBiaya)
29 }
```

PROBLEMS OUTPUT DEBUG CONSOLE **TERMINAL** PORTS

```
PS C:\Users\HP\.vscode> go run "c:\Users\HP\.vscode\alpro_10.go\soal_1.go"
Berat parsel (gram): 8500
Detail berat: 8 kg + 500 gr
Detail biaya: Rp. 80000 + Rp. 2500
Total biaya: Rp. 82500
PS C:\Users\HP\.vscode> go run "c:\Users\HP\.vscode\alpro_10.go\soal_1.go"
Berat parsel (gram): 9250
Detail berat: 9 kg + 250 gr
Detail biaya: Rp. 90000 + Rp. 3750
Total biaya: Rp. 93750
PS C:\Users\HP\.vscode> go run "c:\Users\HP\.vscode\alpro_10.go\soal_1.go"
Berat parsel (gram): 11750
Detail berat: 11 kg + 750 gr
Detail biaya: Rp. 110000 + Rp. 0
Total biaya: Rp. 110000
```

Deskripsi program

Program *BiayaPos* dibuat untuk menghitung biaya pengiriman parsel berdasarkan berat barang yang dimasukkan oleh pengguna dalam satuan gram. Program terlebih dahulu mengubah berat tersebut menjadi dua bagian, yaitu jumlah kilogram (kg) dan sisa berat dalam gram (gr). Biaya dasar pengiriman dihitung dari jumlah kilogram dengan tarif Rp 10.000 per kg. Selanjutnya, program menentukan biaya tambahan menggunakan percabangan else-if. Jika total berat mencapai lebih dari 10 kg, maka sisa berat digratiskan dan tidak dikenai biaya tambahan. Jika tidak lebih dari 10 kg dan sisa berat mencapai minimal 500 gram, maka biaya tambahan dikenakan sebesar Rp 5 per gram. Namun jika sisa berat kurang dari 500 gram, biaya tambahan dihitung sebesar Rp 15 per gram. Setelah semua biaya dihitung, program menampilkan detail berat, rincian biaya, dan total biaya pengiriman sesuai format yang diberikan dalam soal.

2. Tugas 2

Source code

```
package main

import "fmt"

func main() {

    var nam float64

    var nmk string

    fmt.Print("Nilai akhir mata kuliah: ")

    fmt.Scan(&nam)

    if nam > 80 {

        nmk = "A"

    } else if nam > 72.5 {

        nmk = "AB"

    } else if nam > 65 {

        nmk = "B"

    } else if nam > 57.5 {

        nmk = "BC"

    } else if nam > 50 {

        nmk = "C"

    } else if nam > 40 {

        nmk = "D"

    } else {

        nmk = "E"

    }

}
```

```
        fmt.Println("Nilai mata kuliah:", nmk)
    }
}
```

Screenshoot program

```
alpro_10.go >  soal_3.go
1  package main
2  import "fmt"
3
4  func main() {
5      var nam float64
6      var nmk string
7
8      fmt.Print("Nilai akhir mata kuliah: ")
9      fmt.Scan(&nam)
10
11     if nam > 80 {
12         nmk = "A"
13     } else if nam > 72.5 {
14         nmk = "AB"
15     } else if nam > 65 {
16         nmk = "B"
17     } else if nam > 57.5 {
18         nmk = "BC"
19     } else if nam > 50 {
20         nmk = "C"
21     } else if nam > 40 {
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
PS C:\Users\HP\.vscode> go run "c:\Users\HP\.vscode\alpro_10.go\soal_3.go"
Nilai akhir mata kuliah: 93.5
Nilai mata kuliah: A
PS C:\Users\HP\.vscode> go run "c:\Users\HP\.vscode\alpro_10.go\soal_3.go"
Nilai akhir mata kuliah: 70.6
Nilai mata kuliah: B
PS C:\Users\HP\.vscode> go run "c:\Users\HP\.vscode\alpro_10.go\soal_3.go"
Nilai akhir mata kuliah: 49.5
Nilai mata kuliah: D
PS C:\Users\HP\.vscode> 
```

Deskripsi program

Program ini dibuat untuk menentukan nilai mata kuliah (NMK) dalam bentuk huruf berdasarkan nilai akhir mata kuliah (NAM) yang dimasukkan oleh pengguna. Penilaian huruf dibagi ke dalam beberapa kategori sesuai dengan tabel rentang nilai yang telah ditentukan pada soal, yaitu mulai dari A sampai E.

Program membaca input berupa nilai angka (bilangan desimal), kemudian melakukan proses pengecekan melalui serangkaian pernyataan kondisi if – else if – else untuk menentukan kategori nilai yang sesuai.

Setiap kondisi mewakili satu interval nilai berdasarkan standar konversi NAM ke NMK. Ketika nilai numerik memenuhi salah satu syarat interval, program menetapkan nilai huruf yang sesuai dan menghentikan proses pengecekan lebih lanjut.

Setelah kategori nilai ditentukan, program menampilkan hasil akhir berupa nilai mata kuliah dalam bentuk huruf (A, AB, B, BC, C, D, atau E). Dengan demikian, program berfungsi sebagai alat bantu otomatis untuk mengonversi nilai angka menjadi nilai huruf secara konsisten, cepat, dan sesuai dengan ketentuan penilaian yang berlaku.

3. Tugas 3

Source code

```
package main
import "fmt"

func main() {
    var b int

    fmt.Print("Bilangan: ")
    fmt.Scan(&b)

    fmt.Print("Faktor: ")

    jumlahFaktor := 0
    for i := 1; i <= b; i++ {
        if b%i == 0 {
            fmt.Print(i, " ")
            jumlahFaktor++
        }
    }

    fmt.Println()
```

```

    var prima bool

    if jumlahFaktor == 2 {
        prima = true
    } else if jumlahFaktor != 2 {
        prima = false
    }

    fmt.Println("Prima:", prima)
}

```

Screenshoot program

The screenshot shows a Go program in a VS Code editor. The source code defines a function `main()` that prompts the user for a number (`Bilangan`) and its factors (`Faktor`). It then checks if the number is prime (`Prima`) based on the count of factors. The terminal shows the program being run twice: first with input 12 (factors: 1 2 3 4 6 12, not prime) and then with input 7 (factors: 1 7, prime).

```

alpro_10.go > soal_3.go
1  package main
2  import "fmt"
3
4  func main() {
5      var b int
6
7      fmt.Print("Bilangan: ")
8      fmt.Scan(&b)
9
10     fmt.Print("Faktor: ")
11
12     jumlahFaktor := 0
13     for i := 1; i <= b; i++ {
14         if b%i == 0 {
15             fmt.Print(i, " ")
16             jumlahFaktor++
17         }
18     }
19
20     fmt.Println()
21
22     var prima bool
23

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```

Prima: false
PS C:\Users\HP\.vscode> go run "c:\Users\HP\.vscode\alpro_10.go\soal_3.go"
Bilangan: 12
PS C:\Users\HP\.vscode> go run "c:\Users\HP\.vscode\alpro_10.go\soal_3.go"
PS C:\Users\HP\.vscode> go run "c:\Users\HP\.vscode\alpro_10.go\soal_3.go"
Bilangan: 12
Faktor: 1 2 3 4 6 12
Prima: false
PS C:\Users\HP\.vscode> go run "c:\Users\HP\.vscode\alpro_10.go\soal_3.go"
Bilangan: 7
Faktor: 1 7
Prima: true
PS C:\Users\HP\.vscode>

```

Deskripsi program

Program ini dibuat untuk menerima sebuah bilangan bulat $b > 1$, kemudian menentukan dan menampilkan semua faktor dari bilangan tersebut serta memeriksa apakah bilangan tersebut merupakan bilangan prima. Program bekerja dalam dua tahap utama. Pada tahap pertama, program melakukan proses pencarian faktor dengan menggunakan perulangan dari angka 1 hingga b . Setiap nilai yang dapat membagi b tanpa sisa ($b \% i == 0$) dianggap sebagai faktor dan ditampilkan. Selain itu, jumlah faktor dihitung untuk digunakan pada tahap berikutnya. Pada tahap kedua, program menentukan apakah bilangan tersebut merupakan bilangan prima. Suatu bilangan dikatakan prima apabila memiliki tepat dua faktor, yaitu 1 dan bilangan itu sendiri. Penentuan sifat prima dilakukan menggunakan struktur percabangan `if – else if`, sehingga keputusan diambil berdasarkan jumlah faktor yang ditemukan sebelumnya. Akhirnya, program menampilkan daftar faktor bilangan dan status apakah bilangan tersebut termasuk bilangan prima atau bukan. Dengan demikian, program berfungsi sebagai alat untuk menganalisis bilangan bulat dengan menampilkan faktor-faktornya serta menentukan keprimaannya secara efektif dan sistematis.