

LAPORAN PRAKTIKUM
Algoritma Pemrograman

MODUL 10

ELSE-IF



Disusun oleh:

Johanson Leeroy

109082500017

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 == true {
        fmt.Println("Bisa membuat KTP")
    } else {
        fmt.Println("Tidak bisa membuat KTP")
    }
}
```

Screenshot program

The screenshot shows a code editor interface with a Go file named 'Guided1.go'. The code defines a package 'main' with a function 'main()' that reads an integer 'Usia' and a boolean 'KK' from standard input, then prints 'Bisa membuat KTP' if both are true, or 'Tidak bisa membuat KTP' otherwise. Below the editor is a terminal window showing the execution of the program with two different inputs: one where 'Usia' is 17 and 'KK' is true, resulting in the output 'Bisa membuat KTP', and another where 'Usia' is 15 and 'KK' is false, resulting in the output 'Tidak bisa membuat KTP'.

```

1 package main
2
3 import "fmt"
4
5 func main() {
6     var Usia int
7     var KK bool
8
9     fmt.Scan(&Usia, &KK)
10
11    if Usia >= 17 && KK == true {
12        fmt.Println("Bisa membuat KTP")
13    } else {
14        fmt.Println("Tidak bisa membuat KTP")
15    }
16}
17

```

PROBLEMS 10 OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS D:\.new journey\SMTR 1\ALGORITMA DAN PEMROGRAMAN\LAPRAK Modul 10> go run "d:\.new journey\SMTR 1\ALGORITMA DAN PEMROGRAMAN\LAPRAK Modul 10\Guided1.go"

17
true
Bisa membuat KTP

PS D:\.new journey\SMTR 1\ALGORITMA DAN PEMROGRAMAN\LAPRAK Modul 10> go run "d:\.new journey\SMTR 1\ALGORITMA DAN PEMROGRAMAN\LAPRAK Modul 10\Guided1.go"

20
false
Tidak bisa membuat KTP

PS D:\.new journey\SMTR 1\ALGORITMA DAN PEMROGRAMAN\LAPRAK Modul 10> go run "d:\.new journey\SMTR 1\ALGORITMA DAN PEMROGRAMAN\LAPRAK Modul 10\Guided1.go"

15
true
Tidak bisa membuat KTP

PS D:\.new journey\SMTR 1\ALGORITMA DAN PEMROGRAMAN\LAPRAK Modul 10>

Deskripsi program

1. Program diawali dengan package main, import “fmt”, dan func main () {
2. Mendeklarasikan variabel Usia dengan tipe data integer/int
3. Mendeklarasikan variabel KK dengan tipe data Boolean
4. User menginputkan bilangan bulat/integer ke dalam variabel Usia dan nilai kebenaran True/False ke dalam variabel KK
5. Membuat struktur kontrol if-then dengan kondisi Usia \geq 17 && KK == true
6. Jika kondisi bernilai true maka program akan memberi output “Bisa membuat KTP” dan Jika kondisi bernilai false maka program akan mengeksekusi aksi di blok else yaitu berupa output “Tidak bisa membuat KTP”

2. Guided 2

Source Code

```

package main

import "fmt"

func main() {

```

```
var huruf string

fmt.Scan(&huruf)

if huruf == "A" || huruf == "a" || huruf == "I" || huruf
== "i" || huruf == "u" || huruf == "E" ||
huruf == "e" || huruf == "O" || huruf == "o" {
    fmt.Println("Vokal")
} else if (huruf >= "A" && huruf <= "Z") || (huruf >=
"a" && huruf <= "z") {
    fmt.Println("Konsonan")
} else {
    fmt.Println("bukan huruf")
}

}
```

Screenshot program

```

  Guided2.go > main
  1 package main
  2
  3 import "fmt"
  4
  5 func main() {
  6     var huruf string
  7
  8     fmt.Scan(&huruf)
  9
 10    if huruf == "A" || huruf == "a" || huruf == "I" || huruf == "i" || huruf == "U" || huruf == "u" || huruf == "E" ||
 11        huruf == "e" || huruf == "O" || huruf == "o" {
 12        fmt.Println("Vokal")
 13    } else if (huruf >= "A" && huruf <= "Z") || (huruf >= "a" && huruf <= "z") {
 14        fmt.Println("Konsonan")
 15    } else {
 16        fmt.Println("bukan huruf")
 17    }
 18
 19 }
 20

```

PROBLEMS 10 OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS D:\.new_journey\SMTR 1\ALGORITMA DAN PEMROGRAMAN\LAPRAK Modul 10> go run "d:\.new_journey\SMTR 1\ALGORITMA DAN PEMROGRAMAN\LAPRAK Modul 10\Guided2.go"
A
Vokal
PS D:\.new_journey\SMTR 1\ALGORITMA DAN PEMROGRAMAN\LAPRAK Modul 10> go run "d:\.new_journey\SMTR 1\ALGORITMA DAN PEMROGRAMAN\LAPRAK Modul 10\Guided2.go"
f
Konsonan
PS D:\.new_journey\SMTR 1\ALGORITMA DAN PEMROGRAMAN\LAPRAK Modul 10> go run "d:\.new_journey\SMTR 1\ALGORITMA DAN PEMROGRAMAN\LAPRAK Modul 10\Guided2.go"
1
bukan huruf
PS D:\.new_journey\SMTR 1\ALGORITMA DAN PEMROGRAMAN\LAPRAK Modul 10> go run "d:\.new_journey\SMTR 1\ALGORITMA DAN PEMROGRAMAN\LAPRAK Modul 10\Guided2.go"
\$

bukan huruf

Deskripsi program

1. Program diawali dengan package main, import “fmt”, dan func main () {
2. Mendeklarasikan variabel huruf dengan tipe data string
3. User menginputkan string kedalam variabel huruf
4. Membuat struktur kontrol if-then dengan kondisi huruf == "A" || huruf == "a" || huruf == "I" || huruf == "i" || huruf == "U" || huruf == "u" || huruf == "E" || huruf == "e" || huruf == "O" || huruf == "o"
5. Jika kondisi bernilai true maka program akan mengeluarkan output “vokal” dan jika kondisi bernilai false maka program akan lanjut ke struktur kontrol selanjutnya
6. Membuat struktur kontrol else-if dengan kondisi (huruf >= "A" && huruf <= "Z") || (huruf >= "a" && huruf <= "z")
7. Jika kondisi bernilai true maka program akan mengeluarkan output “Konsonan” dan jika kondisi tidak terpenuhi maka program akan mengeluarkan output “bukan huruf”

3. Guided 3

Source Code

```
package main
```

```
import "fmt"
```

```
func main() {
```

```
var bilangan int

var d1, d2, d3, d4 int

fmt.Scan(&bilangan)

d1 = bilangan / 1000

d2 = bilangan / 100 % 10

d3 = bilangan / 10 % 10

d4 = bilangan % 10


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

    fmt.Println("Digit pada bilangan ", bilangan, " terurut membesar")

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

    fmt.Println("Digit pada bilangan ", bilangan, " terurut mengecil")

} else {

    fmt.Println("Digit pada bilangan ", bilangan, " tidak terurut")

}

}
```

Screenshot program

```

--o Guided3.go 1
--o Guided3.go > main
1 package main
2
3 import "fmt"
4
5 func main() {
6     var bilangan int
7     var d1, d2, d3, d4 int
8     fmt.Scan(&bilangan)
9     d1 = bilangan / 1000
10    d2 = bilangan / 100 % 10
11    d3 = bilangan / 10 % 10
12    d4 = bilangan % 10
13
14    if d1 < d2 && d2 < d3 && d3 < d4 {
15        fmt.Println("Digit pada bilangan ", bilangan, " terurut membesar")
16    } else if d1 > d2 && d2 > d3 && d3 > d4 {
17        fmt.Println("Digit pada bilangan ", bilangan, " terurut mengecil")
18    } else {
19        fmt.Println("Digit pada bilangan ", bilangan, " tidak terurut")
20    }
21 }

```

PROBLEMS 10 OUTPUT DEBUG CONSOLE TERMINAL PORTS

```

2489
Digit pada bilangan 2489 terurut membesar
PS D:\new journey\SMTR 1\ALGORITMA DAN PEMROGRAMAN\LAPRAK Modul 10> go run "d:\new journey\SMTR 1\ALGORITMA DAN PEMROGRAMAN\LAPRAK Modul 10\Guided3.go"
2861
Digit pada bilangan 2861 tidak terurut
PS D:\new journey\SMTR 1\ALGORITMA DAN PEMROGRAMAN\LAPRAK Modul 10> go run "d:\new journey\SMTR 1\ALGORITMA DAN PEMROGRAMAN\LAPRAK Modul 10\Guided3.go"
9651
Digit pada bilangan 9651 terurut mengecil
PS D:\new journey\SMTR 1\ALGORITMA DAN PEMROGRAMAN\LAPRAK Modul 10>

```

Ln 15, Col 73 Tab Size: 4 UTF-8 CRLF ⌂ Go 1.25.1 ⌂

Deskripsi program

1. Program diawali dengan package main, import “fmt”, dan func main () {
2. Mendeklarasikan variabel bilangan, d1, d2, d3, d4 dengan tipe data integer/int
3. User menginputkan bilangan bulat/integer ke dalam variabel bilangan
4. Melakuakn perhitungan untuk mendapatkan setiap digit dari bilangan
5. Membuat struktur kontrol if-then dengan kondisi $d1 < d2 \&\& d2 < d3 \&\& d3 < d4$
6. Jika kondisi bernilai true maka program akan menyatakan bahwa bilangan merupakan bilangan yang terurut membesar dan jika kondisi bernilai false maka program berlanjut ke struktur kontrol berikutnya
7. Membuat struktur kontrol else-if dengan kondisi $d1 > d2 \&\& d2 > d3 \&\& d3 > d4$
8. Jika kondisi bernilai true maka program akan menyatakan bahwa bilangan merupakan bilangan yang terurut mengecil dan jika kondisi bernilai false maka program akan menyatakan bahwa bilangan merupakan bilangan yang tidak terurut

TUGAS

1. Tugas 1

Source code

```
package main

import "fmt"

func main() {
    var berat, kg, sisa, hargaTambahan, hargaAkhir int
    fmt.Scan(&berat)

    kg = berat / 1000
    sisa = berat % 1000
    fmt.Println("Detail berat:", kg, "kg +", sisa, "gr")
    hargaAkhir = kg * 10000

    if sisa >= 500 && kg <= 10 {
        hargaTambahan = sisa * 5
        fmt.Println("Detail biaya: Rp.", hargaAkhir, "+"
                    Rp., hargaTambahan)
        hargaAkhir = hargaAkhir + hargaTambahan
        fmt.Print("Total biaya: ", hargaAkhir)
    } else if sisa < 500 && kg <= 10 {
        hargaTambahan = sisa * 15
        fmt.Println("Detail biaya: Rp.", hargaAkhir, "+"
                    Rp., hargaTambahan)
        hargaAkhir = hargaAkhir + hargaTambahan
    }
}
```

```

        fmt.Println("Total biaya: ", hargaAkhir)

    } else if kg > 10 {

        fmt.Println("Detail biaya: Rp.", hargaAkhir, "+"
Rp., hargaTambahan)

        fmt.Println("Total biaya: ", hargaAkhir)

    }

}

```

Screenshot program

The screenshot shows a code editor with the file `Tugas1.go` open. The code implements a function to calculate the total cost based on weight and a surcharge rule. The terminal window shows the execution of the program and its output.

```

109082500017
S11F-13-04
Johanson Leeroy

```

PROBLEMS 10 OUTPUT DEBUG CONSOLE TERMINAL PORTS

```

PS D:\new_journey\SMTR 1\ALGORITMA DAN PEMROGRAMAN\LAPRAK Modul 10> go run "d:\new_journey\SMTR 1\ALGORITMA DAN PEMROGRAMAN\LAPRAK Modul 10\Tugas1.go"
8500
Detail berat: 8 kg + 500 gr
PS D:\new_journey\SMTR 1\ALGORITMA DAN PEMROGRAMAN\LAPRAK Modul 10> go run "d:\new_journey\SMTR 1\ALGORITMA DAN PEMROGRAMAN\LAPRAK Modul 10\Tugas1.go"
9250
Detail berat: 9 kg + 250 gr
Detail biaya: Rp. 90000 + Rp. 3750
Total biaya: 93750
PS D:\new_journey\SMTR 1\ALGORITMA DAN PEMROGRAMAN\LAPRAK Modul 10> go run "d:\new_journey\SMTR 1\ALGORITMA DAN PEMROGRAMAN\LAPRAK Modul 10\Tugas1.go"
11750
Detail berat: 11 kg + 750 gr
Detail biaya: Rp. 110000 + Rp. 0
Total biaya: 110000

```

Deskripsi program

1. Program diawali dengan package main, import “fmt”, dan func main () {
2. Mendeklarasikan variabel berat, kg, sisa, hargaTambahan, hargaAkhir dengan tipe data int/integer
3. User menginputkan bilangan bulat/integer ke dalam variabel berat
4. Melakukan perhitungan untuk menentukan berat dalam Kg dan sisanya
5. Membuat struktur kontrol if-then dengan kondisi sisa $\geq 500 \ \&\amp; \ kg \leq 10$

6. Jika kondisi bernilai true, maka hargaTambahan dihitung sebagai sisa 5. Program mencetak rincian biaya awal dan tambahan, lalu hargaAkhir diperbarui dengan menjumlahkan hargaAkhir dan hargaTambahan sebelum mencetak total biaya.
7. Membuat struktur kontrol else-if dengan kondisi sisa < 500 && kg <= 10
8. Jika kondisi bernilai true, maka hargaTambahan dihitung sebagai sisa * 15. Program mencetak rincian biaya awal dan tambahan, lalu hargaAkhir diperbarui dengan menjumlahkan hargaAkhir dan hargaTambahan sebelum mencetak total biaya.
9. Membuat struktur kontrol else-if yang terakhir dengan kondisi kg > 10
10. Jika kondisi bernilai true, program mencetak rincian biaya awal dan biaya tambahan (yang bernilai 0), serta total biaya yang sama dengan harga awal (hargaAkhir).

2. Tugas 2

1. Keluaran dari program tersebut berupa error karena kesalahan pada penulisan variabel
2. Ada banyak perbaikan yang harus diperbaiki, yaitu kondisi pada setiap struktur kontrol if-then dan juga penulisan variabel

Source code (Perbaikan Kode)

```
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"
    }
    if nam > 72.5 && nam <= 80 {
        nmk = "AB"
    }
}
```

```
if nam > 65 && nam <= 72.5 {  
    nmk = "B"  
}  
  
if nam > 57.5 && nam <= 65 {  
    nmk = "BC"  
}  
  
if nam > 50 && nam <= 57.5 {  
    nmk = "C"  
}  
  
if nam > 40 && nam <= 50 {  
    nmk = "D"  
}  
  
if nam <= 40 {  
    nmk = "E"  
}  
  
fmt.Println("Nilai mata kuliah: ", nmk)  
}
```

Screenshot program

```

1  package main
2
3  import "fmt"
4
5  func main() {
6      var nam float64
7      var nmk string
8      fmt.Println("Nilai akhir mata kuliah: ")
9      fmt.Scan(&nam)
10     if nam > 80 {
11         nmk = "A"
12     }
13     if nam > 72.5 && nam <= 80 {
14         nmk = "AB"
15     }
16     if nam > 65 && nam <= 72.5 {
17         nmk = "B"
18     }
19     if nam > 57.5 && nam <= 65 {
20         nmk = "BC"
21     }
22     if nam > 50 && nam <= 57.5 {
23         nmk = "C"
24     }
25     if nam > 40 && nam <= 50 {
26         nmk = "D"
27     }
28     if nam <= 40 {
29         nmk = "E"
30     }
31     fmt.Println("Nilai mata kuliah: ", nmk)
32 }

```

PROBLEMS 19 OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS D:\new_journey\SMTR 1\ALGORITMA DAN PEMROGRAMAN\LAPRAK Modul 10> go run "d:\new_journey\SMTR 1\ALGORITMA DAN PEMROGRAMAN\LAPRAK Modul 10\Tugas2.go"
PS D:\new_journey\SMTR 1\ALGORITMA DAN PEMROGRAMAN\LAPRAK Modul 10> go run "d:\new_journey\SMTR 1\ALGORITMA DAN PEMROGRAMAN\LAPRAK Modul 10\Tugas2.go"
Nilai akhir mata kuliah: 70.6
Nilai mata kuliah: B
PS D:\new_journey\SMTR 1\ALGORITMA DAN PEMROGRAMAN\LAPRAK Modul 10> go run "d:\new_journey\SMTR 1\ALGORITMA DAN PEMROGRAMAN\LAPRAK Modul 10\Tugas2.go"
Nilai akhir mata kuliah: 49.5
Nilai mata kuliah: D
PS D:\new_journey\SMTR 1\ALGORITMA DAN PEMROGRAMAN\LAPRAK Modul 10>

Deskripsi program

1. Program diawali dengan package main, import “fmt”, dan func main () {
2. Mendeklarasikan variabel nam dengan tipe data float64
3. Mendeklarasikan variabel nmk dengan tipe data string
4. User menginputkan bilangan rill ke dalam variabel nam
5. Membuat struktur kontrol if-then dengan kondisi nam > 80. Jika true maka nmk = “A”
6. Membuat struktur kontrol if-then dengan kondisi nam > 72.5 && nam <= 80. Jika true maka nmk = “AB”
7. Membuat struktur kontrol if-then dengan kondisi nam > 65 && nam <= 72.5. Jika true maka nmk = “B”
8. Membuat struktur kontrol if-then dengan kondisi nam > 57.5 && nam <= 65. Jika true maka nmk = “BC”
9. Membuat struktur kontrol if-then dengan kondisi nam > 50 && nam <= 57.5. Jika true maka nmk = “C”
10. Membuat struktur kontrol if-then dengan kondisi nam > 40 && nam <= 50. Jika true maka nmk = “D”
11. Membuat struktur kontrol if-then dengan kondisi nam <= 40. Jika true maka nmk = “E”
12. Output berupa string akhir pada variabel nmk

3. Tugas 3

Source code

```
package main

import "fmt"

func main() {
    var bil, i, sisa, faktor int
    var prima bool
    fmt.Scan(&bil)

    for i = 1; i <= bil; i++ {
        sisa = bil % i
        if sisa == 0 {
            faktor = i
            fmt.Println(faktor, " ")
        }
    }
    fmt.Println()

    if bil%2 == 0 || bil%3 == 0 {
        prima = false
    } else {
        prima = true
    }
    if bil <= 3 && bil > 0 {
        prima = true
    }
    fmt.Println("Prima: ", prima)
}
```

Screenshot program

The screenshot shows a code editor interface with a terminal window below it. The code editor displays a Go program named Tugas3.go. The terminal window shows the execution of the program and its output.

```
1 package main
2
3 import "fmt"
4
5 func main() {
6     var bil, i, sisa, faktor int
7     var prima bool
8     fmt.Scan(&bil)
9
10    for i = 1; i <= bil; i++ {
11        sisa = bil % i
12        if sisa == 0 {
13            faktor = i
14            fmt.Print(faktor, " ")
15        }
16    }
17    fmt.Println()
18
19    if bil%2 == 0 || bil%3 == 0 {
20        prima = false
21    } else {
22        prima = true
23    }
24    if bil <= 3 && bil > 0 {
25        prima = true
26    }
27    fmt.Println("Prima: ", prima)
28 }
```

Terminal Output:

```
PS D:\.new_journey\SMTR 1\ALGORITMA DAN PEMROGRAMAN\LAPRAK Modul 10> go run "d:\.new_journey\SMTR 1\ALGORITMA DAN PEMROGRAMAN\LAPRAK Modul 10\Tugas3.go"
12
1 2 3 4 6 12
Prima: false
PS D:\.new_journey\SMTR 1\ALGORITMA DAN PEMROGRAMAN\LAPRAK Modul 10> go run "d:\.new_journey\SMTR 1\ALGORITMA DAN PEMROGRAMAN\LAPRAK Modul 10\Tugas3.go"
7
1 7
Prima: true
PS D:\.new_journey\SMTR 1\ALGORITMA DAN PEMROGRAMAN\LAPRAK Modul 10> []
```

Deskripsi program

1. Program diawali dengan package main, import “fmt”, dan func main () {
2. Mendeklarasikan variabel bil, i, sisa, faktor dengan tipe data integer/int
3. Mendeklarasikan variabel prima dengan tipe data boolean
4. User menginputkan bilangan bulat/integer ke dalam variabel bil
5. Membuat struktur perulangan for-loop dengan inisialisasi i = 1, kondisi i <= bil dan update i++. Jika kondisi masih terpenuhi program akan melakukan: perhitungan sisa dengan sisa = bil % i lalu membuat struktur kontrol if-then dengan kondisi sisa == 0. Jika kondisi terpenuhi program akan menyatakan i dalam perhitungan sebagai faktor dari bil
6. Membuat struktur kontrol if-then pertama dengan kondisi bil%2 == 0 || bil%3 == 0
7. Jika kondisi bernilai true maka program akan memberi nilai kebenaran false pada variabel prima selain itu maka prima = true
8. Membuat struktur kontrol if=then kedua dengan kondisi bil<= 3 && bil> 0
9. Jika kondisi bernilai true maka program akan memberi nilai kebenaran true pada variabel prima
10. Keseluruhan output berupa deretan faktor dari bil dan juga identifikasi bil merupakan bilangan prima atau bukan