

**LAPORAN PRAKTIKUM ALGORITMA
DAN PEMROGRAMAN 1**

MODUL 10

ELSE-IF



Disusun oleh:

Anindya Rahadita Yumnaa

109082500138

S1IF-13-07

Asisten Praktikum

Adithana dharma putra

Apri pandu wicaksono

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")

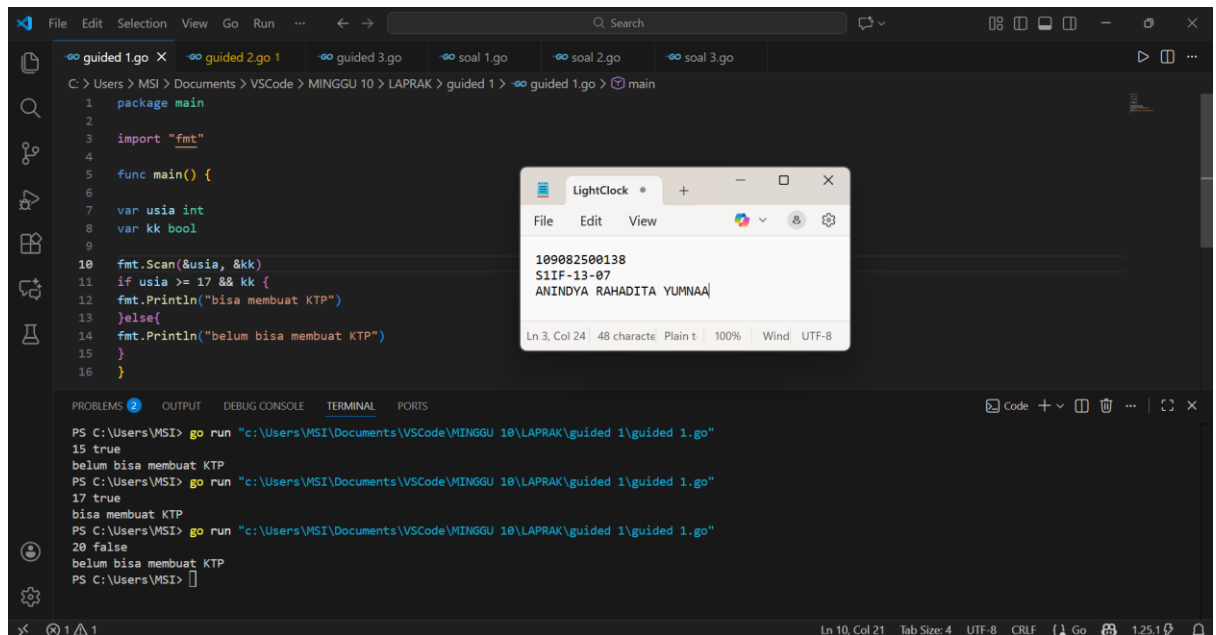
    }else{

        fmt.Println("belum bisa membuat KTP")

    }

}
```

Screenshoot program



Deskripsi program

Program sederhana ini dibuat untuk **mengecek** apakah seseorang sudah **memenuhi syarat untuk mengurus KTP**. Cara kerjanya, program akan meminta kita memasukkan **usia** dan **status Kartu Keluarga (KK)**, kemudian menginputkan **ya** atau **tidak** untuk KK-nya. Setelah kedua informasi itu dimasukkan, program langsung mengecek: **apakah usia sudah 17 tahun atau lebih, dan apakah benar-benar sudah punya Kartu Keluarga?** Hanya jika kedua hal itu benar, program akan memberi tahu bahwa **"bisa membuat KTP"**. Jika salah satu syarat belum terpenuhi, misalnya baru berumur 16 tahun, maka hasilnya adalah **"belum bisa membuat KTP"**.

2. Guided 2

Source Code

```
package main

import "fmt"

func main() {

    var x rune

    var huruf, vKecil, vBesar boo

    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{

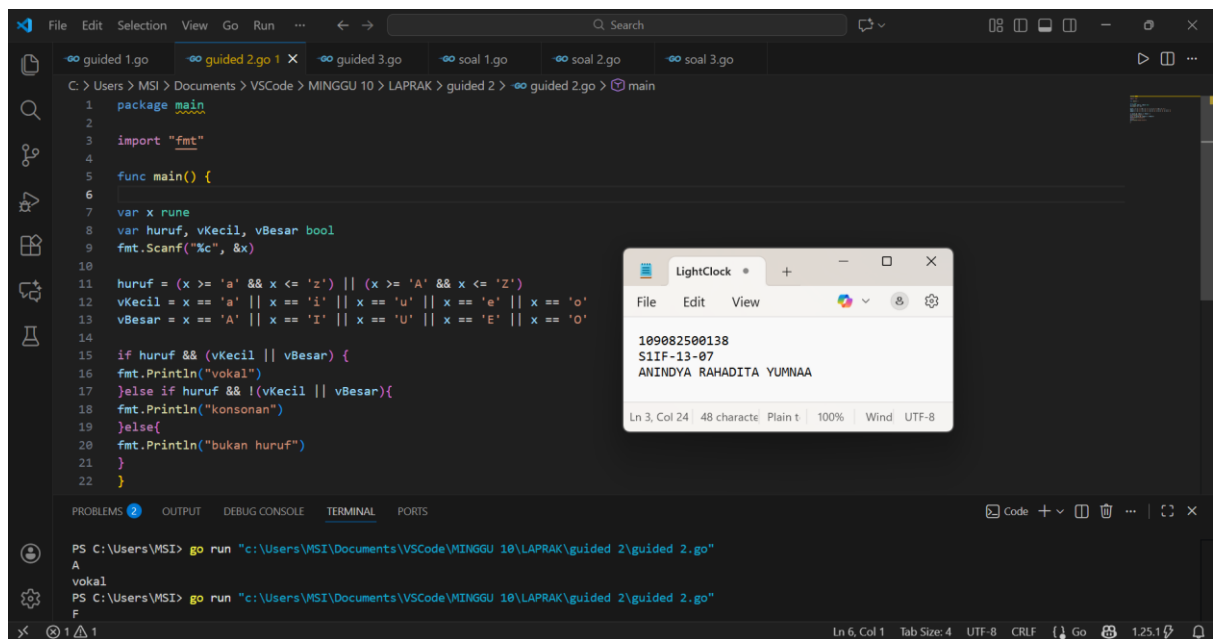
fmt.Println("bukan huruf")

}

}

```

Screenshoot program



```
1 package main
2
3 import "fmt"
4
5 func main() {
6
7     var x rune
8     var huruf, vKecil, vBesar bool
9     fmt.Scanf("%c", &x)
10
11     huruf = (x >= 'a' && x <= 'z') || (x >= 'A' && x <= 'Z')
12     vKecil = x == 'a' || x == 'i' || x == 'u' || x == 'e' || x == 'o'
13     vBesar = x == 'A' || x == 'I' || x == 'U' || x == 'E' || x == 'O'
14
15     if huruf {
16         if vKecil || vBesar {
17             fmt.Println("vokal")
18         } else {
19             fmt.Println("konsonan")
20         }
21     } else {
22         fmt.Println("bukan huruf")
23     }
24 }
```

PS C:\Users\MSI> go run "c:\Users\MSI\Documents\VSCode\MINGGU 10\LAPRAK\guided 2\guided 2.go"

A

vokal

PS C:\Users\MSI> go run "c:\Users\MSI\Documents\VSCode\MINGGU 10\LAPRAK\guided 2\guided 2.go"

F

konsonan

PS C:\Users\MSI> go run "c:\Users\MSI\Documents\VSCode\MINGGU 10\LAPRAK\guided 2\guided 2.go"

1

bukan huruf

PS C:\Users\MSI> go run "c:\Users\MSI\Documents\VSCode\MINGGU 10\LAPRAK\guided 2\guided 2.go"

\$

bukan huruf

PS C:\Users\MSI>

Deskripsi program

Program ini untuk **menganalisis satu karakter** yang akan dimasukkan, bisa berupa huruf, angka, atau simbol, kemudian memutuskan apakah inputan tersebut termasuk kelompok yang mana. Pertama, program akan mengecek, "Apakah ini **huruf** secara umum (A-Z atau a-z)?" Jika ya, program masuk ke tahap berikutnya. Di tahap kedua, program mengecek, "Apakah huruf ini termasuk **vokal** (A, I, U, E, O)?" Jika benar, hasilnya langsung dicap "**vokal**". Namun, jika di tahap kedua ternyata inputan adalah huruf tapi **bukan vokal**, otomatis inputan tersebut dicap sebagai "**konsonan**". Terakhir, jika dari awal karakter itu bahkan **bukan huruf sama sekali** (misalnya, '\$'), maka program akan mencapnya sebagai "**bukan huruf**". Singkatnya, ini adalah sistem otomatis untuk mengklasifikasikan input tunggal menjadi vokal, konsonan, atau lainnya.

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) / 100
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

The screenshot shows the Visual Studio Code editor with a Go file named `guided 3.go`. The code defines a `main` function that takes a number and a string, processes the number into four digits (d1, d2, d3, d4), and then checks if the digits are in ascending or descending order. A `LightClock` window is open, displaying the input number `109082500138`, the date `S11F-13-07`, and the name `ANINDYA RAHADITA YUMNAA`.

```
1 package main
2
3 import "fmt"
4
5 func main() {
6
7     var bilangan, d1, d2, d3, d4 int
8     var teks string
9
10    fmt.Print("Bilangan: ")
11    fmt.Scan(&bilangan)
12
13    d4 = bilangan % 10
14    d3 = (bilangan % 100) / 10
15    d2 = (bilangan % 1000) / 100
16    d1 = bilangan / 1000
17
18    if d1 < d2 && d2 < d3 && d3 < d4 {
19        teks = "terurut membesar"
20    } else if d1 > d2 && d2 > d3 && d3 > d4 {
21        teks = "terurut mengecil"
22    } else {
23        teks = "tidak terurut"
24    }
25    fmt.Println("Digit pada bilangan", bilangan, teks)
26 }
```

The screenshot shows the same VS Code editor with the `guided 3.go` file. The `TERMINAL` panel at the bottom displays the output of the program for three different inputs: 2489, 3861, and 9651. The output shows the digit breakdown and the resulting comparison result for each input.

```
PS C:\Users\MSI> go run "c:\Users\MSI\Documents\VSCode\MINGGU 10\LAPRAK\guided 3\guided 3.go"
Bilangan: 2489
Digit pada bilangan 2489 terurut membesar
PS C:\Users\MSI> go run "c:\Users\MSI\Documents\VSCode\MINGGU 10\LAPRAK\guided 3\guided 3.go"
Bilangan: 3861
Digit pada bilangan 3861 tidak terurut
PS C:\Users\MSI> go run "c:\Users\MSI\Documents\VSCode\MINGGU 10\LAPRAK\guided 3\guided 3.go"
Bilangan: 9651
Digit pada bilangan 9651 terurut mengecil
PS C:\Users\MSI>
```

Deskripsi program

Program ini berfungsi untuk **menganalisis susunan digit pada sebuah angka** yang dimasukkan oleh pengguna, untuk mengetahui apakah angka tersebut terurut. Pertama, program akan meminta untuk memasukkan sebuah angka bulat (yang diasumsikan empat digit). Kemudian, program akan memecah angka tersebut menjadi digit-digit penyusunnya, yaitu d1 (ribuan), d2 (ratusan), d3 (puluhan), dan d4 (satuan), menggunakan perhitungan sisa bagi (%) dan bagi (/). Setelah semua digit didapatkan, program akan **memeriksa urutannya**: jika setiap digit berikutnya **selalu lebih besar** dari digit sebelumnya (misalnya 1234), maka hasilnya adalah **"terurut membesar"**. Jika setiap digit berikutnya **selalu lebih kecil** (misalnya 4321), maka hasilnya adalah **"terurut mengecil"**. Selain kedua kondisi tersebut, program akan menyatakan bahwa

angka itu **"tidak terurut"**. Hasil akhirnya akan ditampilkan, memberitahu bahwa status keterurutan digit pada angka yang telah dimasukkan.

TUGAS

1. Tugas 1

Source code

```
package main

import "fmt"

func main() {

    var berat, kg, sisa int

    var biayaKg, biayaSisa, total int

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

    fmt.Scan(&berat)

    kg = berat / 1000

    sisa = berat % 1000

    fmt.Println("Detail berat:", kg, "kg +", sisa, "gr")

    biayaKg = kg * 10000

    if sisa == 0 {

        biayaSisa = 0

    } else if sisa >= 500 {

        biayaSisa = sisa * 5

    } else {
```



```

        biayaSisa = sisa * 15

    }

    fmt.Println("Detail biaya: Rp.", biayaKg, "+ Rp.",
biayaSisa)

    total = biayaKg + biayaSisa

    fmt.Println("Total biaya: Rp.", total)

}

```

Screenshoot program

The screenshot shows a Go program in VS Code. The program calculates the total cost based on weight (kg) and a remaining amount (sisa). It includes a conditional logic for 'sisa' values. A 'LightClock' window is overlaid on the code, displaying a timestamp and user information.

```

1 package main
2
3 import "fmt"
4
5 func main() {
6
7     var berat, kg, sisa int
8     var biayaKg, biayaSisa, total int
9
10    fmt.Println("Berat parcel (gram): ")
11    fmt.Scan(&berat)
12    kg = berat / 1000
13    sisa = berat % 1000
14    fmt.Println("Detail berat:", kg, "kg +", sisa, "gr")
15    biayaKg = kg * 10000
16
17    if sisa == 0 {
18        biayaSisa = 0
19    } else if sisa >= 500 {
20        biayaSisa = sisa * 5
21    } else {
22        biayaSisa = sisa * 15
23    }
24    fmt.Println("Detail biaya: Rp.", biayaKg, "+ Rp.", biayaSisa)
25
26    total = biayaKg + biayaSisa
27    fmt.Println("Total biaya: Rp.", total)
28 }

```

LightClock

File Edit View

109082500138
S1IF-13-07
ANINDYA RAHADITA YUMNAA

Ln 3, Col 24 48 character Plain t 100% Wind UTF-8

Ln 13, Col 24 Spaces: 4 UTF-8 CRLF Go 1.25.1

```
1 package main
2
3 import "fmt"
4
5 func main() {
6
7     var berat, kg, sisa int
8     var biayaKg, biayaSisa, total int
9
10    fmt.Print("Berat parcel (gram): ")
11    fmt.Scan(&berat)
```

```
PS C:\Users\MSI> go run "c:\Users\MSI\Documents\VSCode\MINGGU 10\LAPRAK\soal 1\soal 1.go"
Berat parcel (gram): 8500
Detail berat: 8 kg + 500 gr
Detail biaya: Rp. 80000 + Rp. 2500
Total biaya: Rp. 82500
PS C:\Users\MSI> go run "c:\Users\MSI\Documents\VSCode\MINGGU 10\LAPRAK\soal 1\soal 1.go"
Berat parcel (gram): 9250
Detail berat: 9 kg + 250 gr
Detail biaya: Rp. 90000 + Rp. 3750
Total biaya: Rp. 93750
PS C:\Users\MSI> go run "c:\Users\MSI\Documents\VSCode\MINGGU 10\LAPRAK\soal 1\soal 1.go"
Berat parcel (gram): 11750
Detail berat: 11 kg + 750 gr
Detail biaya: Rp. 110000 + Rp. 3750
Total biaya: Rp. 113750
PS C:\Users\MSI>
```

Deskripsi program

Program ini berfungsi sebagai **penghitung (kalkulator) otomatis untuk menghitung biaya pengiriman paket** berdasarkan beratnya dalam satuan gram. Pertama, program akan meminta untuk memasukkan **berat total parcel** yang akan dikirim (dalam gram). Begitu data berat diterima, program langsung **memecah berat tersebut** menjadi dua bagian: berat penuh dalam **kilogram (kg)** dan sisanya dalam **gram (gr)**. **Biaya Kilogram** dihitung dengan mengalikan berat penuh dalam kg dengan tarif dasar Rp10.000 per kg.

Biaya Sisa Gram dihitung dengan aturan berlapis:

- Jika sisa gramnya **nol** (beratnya pas per kg), biaya sisa gramnya gratis (nol).
- Jika sisa gramnya **500 gram atau lebih**, biaya dihitung Rp5 per gram.
- Jika sisa gramnya **kurang dari 500 gram**, biayanya lebih mahal, yaitu Rp15 per gram.

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 {
    nam = "A"
}

if nam > 72.5 {
    nam = "AB"
}

if nam > 65 {
    nam = "B"
}

if nam > 57.5 {
    nam = "BC"
}

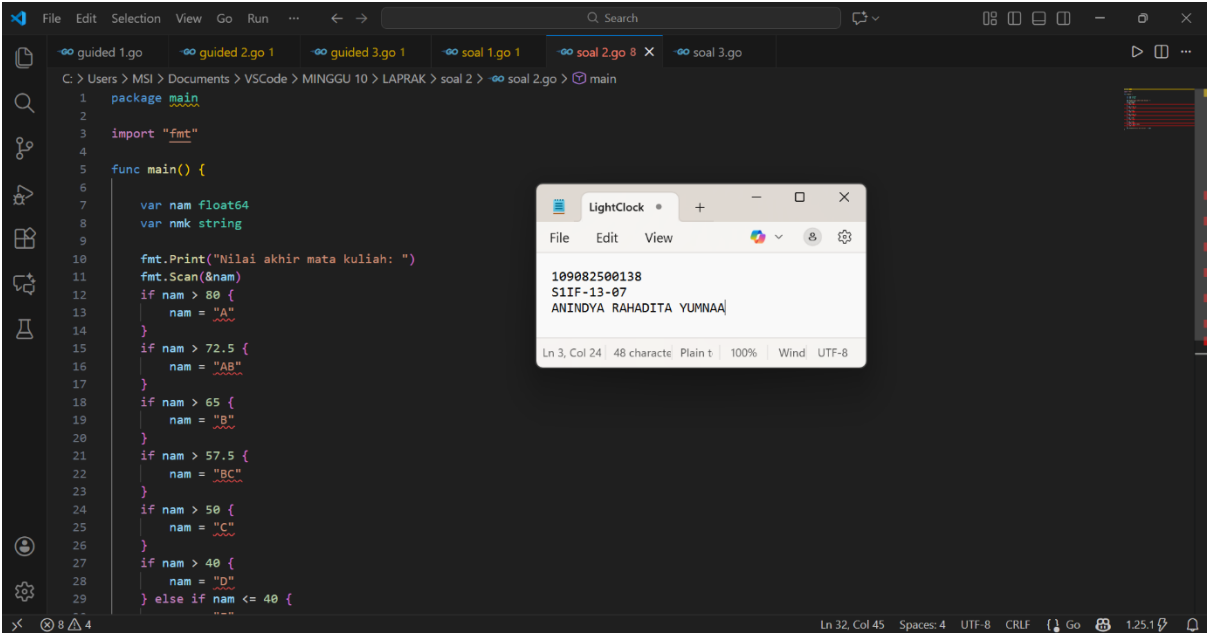
if nam > 50 {
    nam = "C"
}

if nam > 40 {
    nam = "D"
} else if nam <= 40 {
    nam = "E"
}

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

```
}
```

Screenshoot program

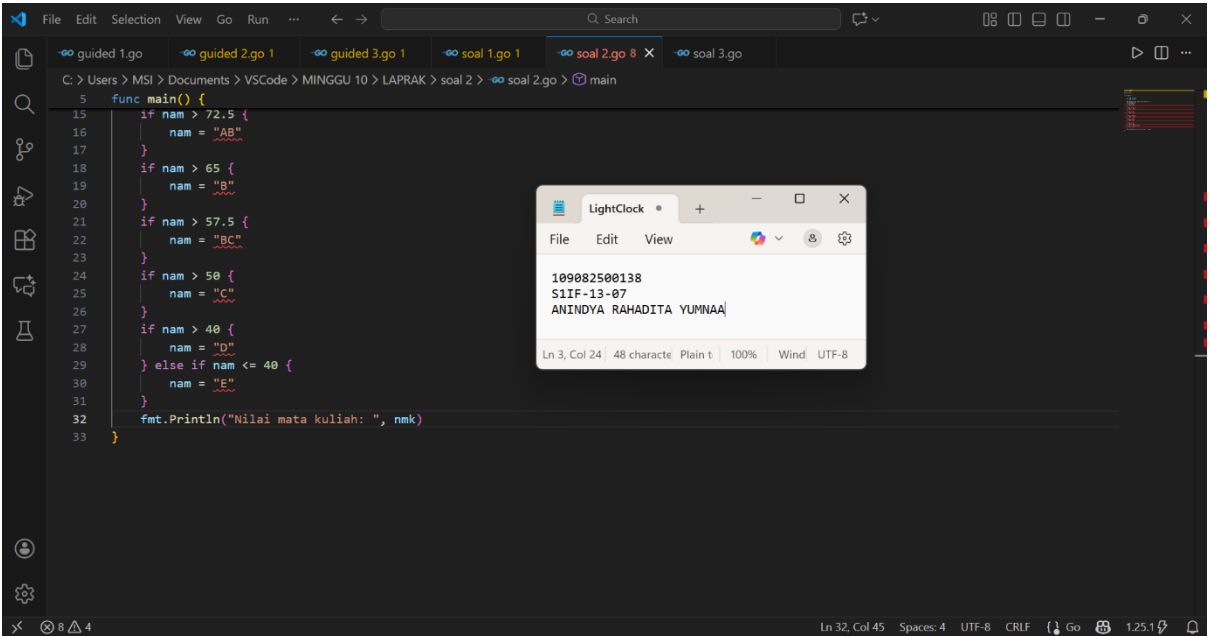


The screenshot shows the VS Code editor with a Go file named `soal 2.go`. The code defines a `main` function that reads a name and a score, then prints the final score. A terminal window titled `LightClock` displays the output of the program.

```
1 package main
2
3 import "fmt"
4
5 func main() {
6
7     var nam float64
8     var nmk string
9
10    fmt.Print("Nilai akhir mata kuliah: ")
11    fmt.Scan(&nam)
12    if nam > 80 {
13        nam = "A"
14    }
15    if nam > 72.5 {
16        nam = "AB"
17    }
18    if nam > 65 {
19        nam = "B"
20    }
21    if nam > 57.5 {
22        nam = "BC"
23    }
24    if nam > 50 {
25        nam = "C"
26    }
27    if nam > 40 {
28        nam = "D"
29    } else if nam <= 40 {
30        nam = "E"
31    }
32    fmt.Println("Nilai akhir mata kuliah: ", nmk)
33 }
```

Terminal Output:

```
109082500138
S11F-13-07
ANINDYA RAHADITA YUMNAA
```

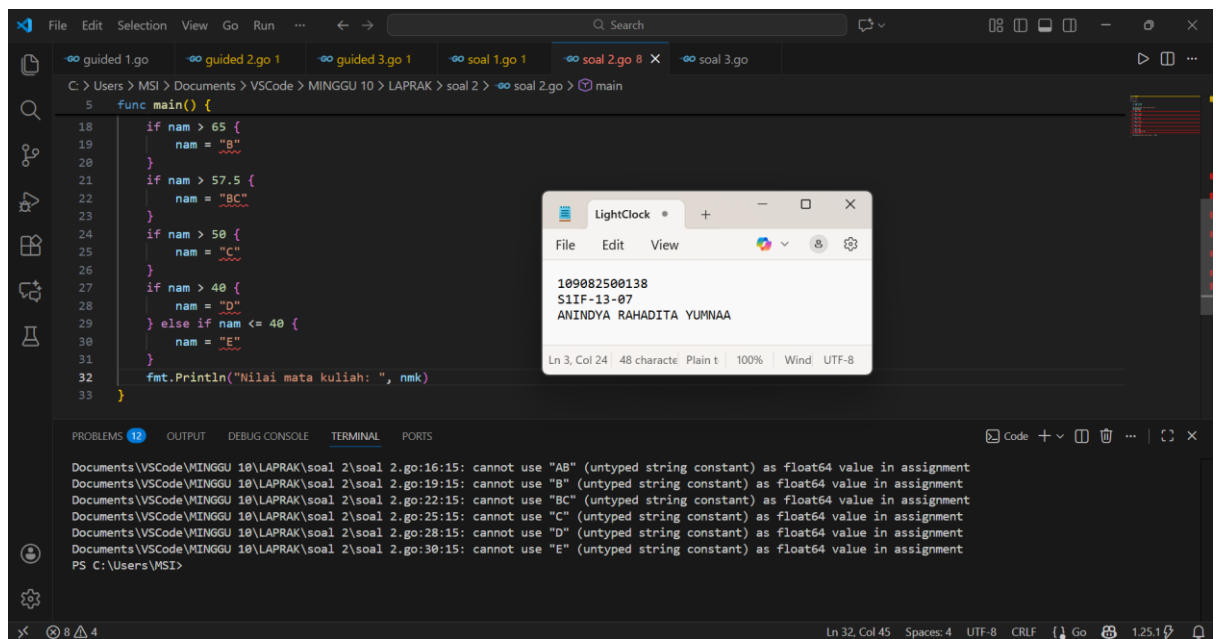


The screenshot shows the same VS Code editor with the same Go file `soal 2.go`. The code is identical to the first screenshot. The terminal window titled `LightClock` displays the output of the program.

```
5 func main() {
15    if nam > 72.5 {
16        nam = "AB"
17    }
18    if nam > 65 {
19        nam = "B"
20    }
21    if nam > 57.5 {
22        nam = "BC"
23    }
24    if nam > 50 {
25        nam = "C"
26    }
27    if nam > 40 {
28        nam = "D"
29    } else if nam <= 40 {
30        nam = "E"
31    }
32    fmt.Println("Nilai akhir mata kuliah: ", nmk)
33 }
```

Terminal Output:

```
109082500138
S11F-13-07
ANINDYA RAHADITA YUMNAA
```



- Jika **nam** diberikan adalah 80.1, apa keluaran dari program tersebut? Apakah eksekusi program tersebut sesuai spesifikasi soal?
: Keluaran dari program tersebut tidak dapat berjalan sama sekali, karena program Go tidak akan menjalankan variabel `float64` yang diisi dengan string. Kemudian untuk eksekusi program tersebut tidak memenuhi spesifikasi, yang dimana untuk `float64` diisi dengan string "A", "AB", "B", "BC", "C", "D", "E".
- Apa saja kesalahan dari program tersebut? Mengapa demikian? Jelaskan alur program seharusnya!
: Beberapa kesalahan dari program tersebut adalah:
 - 1) Tipe data yang salah (**float64** namun diisi dengan string)
 - 2) Penempatan inputan variabel yang salah. Seharusnya **nam**, bukan **nmk**
 - 3) Rantai kondisi tidak terhubung. Hanya dua baris terakhir yang menggunakan **else if**, sedangkan kondisi di atasnya menggunakan **if** terpisah. **else if** seharusnya terhubung dengan seluruh rantai kondisi, bukan hanya dua baris terakhir.
 - 4) **if** terpisah, bukan **if – else if – else** yang saling menyambung. Akibatnya, semua kondisi yang benar akan dieksekusi, bukan hanya satu. Seperti pada kondisi, jika menginputkan 80.1 hasil akhirnya adalah "D", bukan "A". Karena semua kondisi dijalankan terus sampai yang paling bawah.

Untuk alur program seharusnya:

1. Program akan meminta untuk menginput nama mahasiswa
2. Program meminta input nilai mahasiswa dalam bentuk angka (`float`), disini `float64`

3. Program membandingkan nilai tersebut menggunakan **rantai kondisi yang terhubung** (if – else if – else)
4. Hanya **satu kondisi** yang dieksekusi, yaitu kondisi pertama yang benar
5. Program menampilkan kategori nilai berdasarkan rentang nilai tersebut

c. Perbaiki program tersebut! Ujilah dengan masukan: 93.5; 70.6; dan 49.5. Seharusnya keluaran yang diperoleh adalah 'A', 'B', dan 'D'.

: Berikut perbaikan dari program tersebut, beserta output yang diperoleh

The screenshot shows a VS Code editor with a Go file named `soal 2.go`. The code implements a program that takes a score and a name as input and outputs a grade category based on the score. The code is as follows:

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

Overlaid on the editor is a terminal window titled "LightClock" showing the following input and output:

```
109082500138
S1IF-13-07
ANINDYA RAHADITA YUMNAA
```

This screenshot shows the same VS Code editor with the same Go code. The terminal window at the bottom shows the output of running the program three times with different inputs:

```
PS C:\Users\MSI> go run "c:\Users\MSI\Documents\VSCode\MINGGU 10\LAPRAK\soal 2\soal 2.go"
Nilai akhir mata kuliah: 93.5
Nilai mata kuliah: A
PS C:\Users\MSI> go run "c:\Users\MSI\Documents\VSCode\MINGGU 10\LAPRAK\soal 2\soal 2.go"
Nilai akhir mata kuliah: 70.6
Nilai mata kuliah: B
PS C:\Users\MSI> go run "c:\Users\MSI\Documents\VSCode\MINGGU 10\LAPRAK\soal 2\soal 2.go"
Nilai akhir mata kuliah: 49.5
Nilai mata kuliah: D
PS C:\Users\MSI>
```

Deskripsi program

Program ini berfungsi sebagai **konverter nilai otomatis** untuk mengubah nilai akhir angka menjadi **nilai huruf (seperti A, B, C, dst.)**. Program akan meminta untuk memasukkan **Nilai Akhir Mata Kuliah** berupa angka (berkemungkinan juga memiliki koma). Setelah angka dimasukkan, program langsung **menguji nilai** tersebut berdasarkan serangkaian aturan secara berurutan: jika nilainya di atas 80 dapat "A"; jika di atas 72.5 dapat "AB"; di atas 65 dapat "B", dan seterusnya, dengan batas terendah di atas 40 masih dapat "D". Jika nilai tersebut bahkan tidak mencapai batas 40, maka nilai mutu yang diberikan adalah "E". Di akhir proses, program akan menampilkan **nilai huruf** yang sesuai dengan nilai angka yang telah dimasukkan.

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 {
        prima = false
    }
}
```

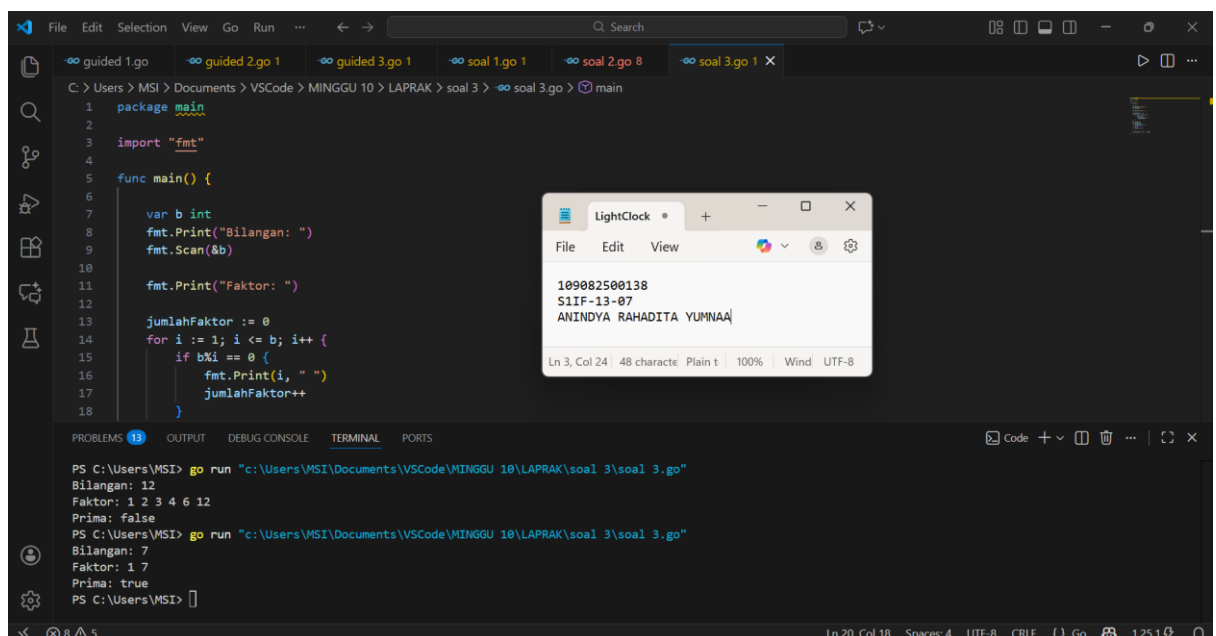
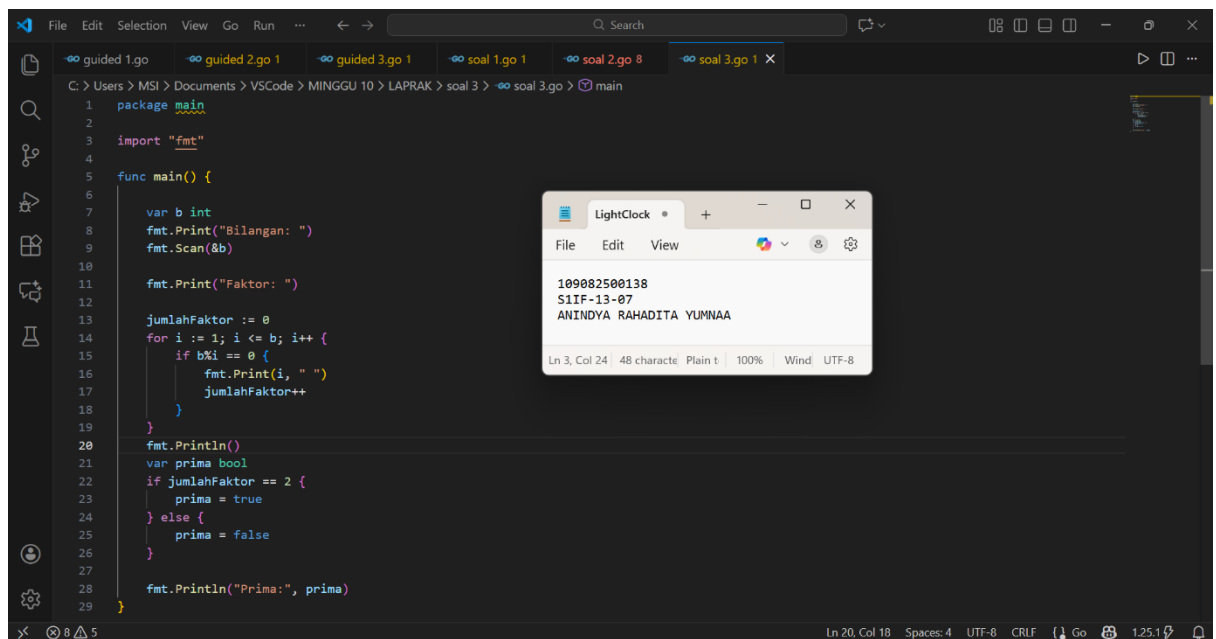
```

    }

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

```

Screenshoot program



Deskripsi program

Program dirancang untuk **mencari semua faktor pembagi dari sebuah bilangan** yang dimasukkan, dan sekaligus **menentukan apakah bilangan tersebut adalah bilangan prima**. Cara kerjanya dimulai dengan meminta untuk memasukkan sebuah angka

bulat. Setelah angka diinput, program akan memeriksa setiap angka mulai dari 1 hingga bilangan itu sendiri. Setiap kali sebuah angka ditemukan bisa **membagi habis** bilangan tanpa sisa (yaitu faktornya), program akan mencetak angka faktor tersebut dan mencatatnya ke dalam sebuah penghitung (jumlahFaktor). Setelah semua angka selesai diperiksa, program akan **menentukan status prima**: jika total faktor yang ditemukan **hanya ada dua** (yaitu 1 dan bilangan itu sendiri), maka bilangan tersebut dinyatakan sebagai "**Prima: true**"; jika jumlah faktornya lebih atau kurang dari dua, maka hasilnya "**Prima: false**".