

**LAPORAN PRAKTIKUM**  
**Algoritma Pemrograman**

**EVALUASI**



**Disusun oleh:**

**FERDINAND AXEL VALERIAN**

**109082500154**

**S1IF-13-07**

**PROGRAM STUDI S1 INFORMATIKA**  
**FAKULTAS INFORMATIKA**  
**TELKOM UNIVERSITY PURWOKERTO**  
**2025**

## SOAL

### 1. SOAL 1

#### Source Code

```
package main

import (
    "fmt"
)

func main() {
    var ketinggianTotal int = 0
    var inputTambah int

    for {
        switch {
        case ketinggianTotal == 0:
            fmt.Println("Masukkan ketinggian pohon wok (N > 8):")
        default:
            fmt.Printf("yah kurang wok, cuma %d (masih <= 8). tambahin lagi wok: ", ketinggianTotal)
        }
    }

    _, err := fmt.Scan(&inputTambah)

    if err != nil {
        fmt.Println("\nInput ngga valid. Pastiin inputnya cuma angka.")
        var discard string
    }
}
```

```
    fmt.Scanln(&discard)

    continue

}

kettinggianTotal += inputTambah

if ketinggianTotal > 8 {

    fmt.Printf("\nKetinggian totalnya %d yh. otw
bikin...\n\n", ketinggianTotal)

    break

}

for i := 1; i <= ketinggianTotal; i++ {

    for j := 0; j < ketinggianTotal-i; j++ {

        fmt.Print(" ")

    }

    for k := 0; k < 2*i-1; k++ {

        fmt.Print("*")

    }

    fmt.Println()

}

for i := 0; i < 3; i++ {

    for j := 0; j < ketinggianTotal-1; j++ {

        fmt.Print(" ")

    }

}
```

```

        }

        fmt.Println("|")

    }

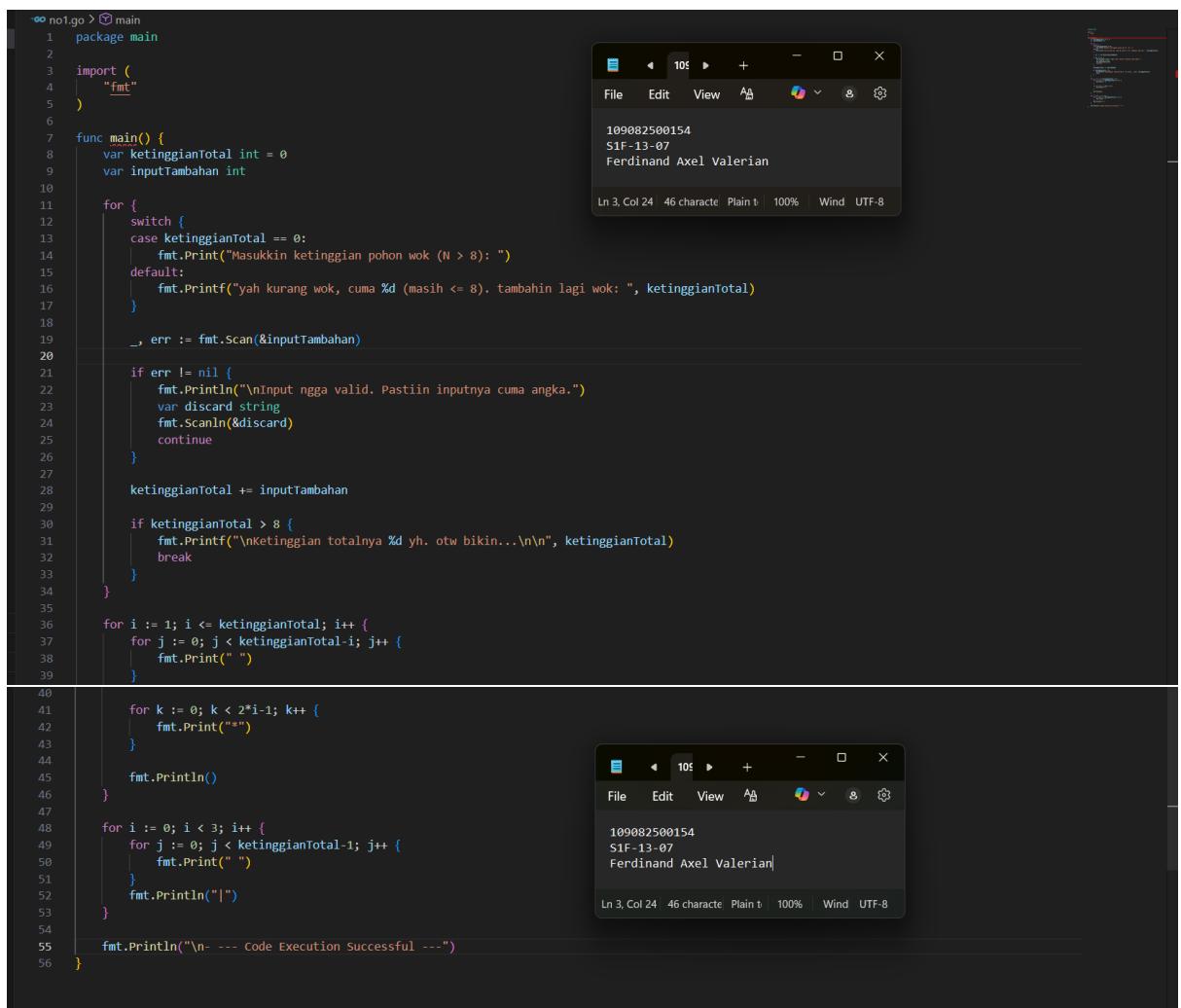
}

fmt.Println("\nCode Execution Successful ---")

}

```

## Screenshot program



The screenshot shows a Go code editor with a file named `not.go`. The code implements a command-line application for calculating the total height of trees based on their individual heights and printing a diamond pattern.

```

1  package main
2
3  import (
4      "fmt"
5  )
6
7  func main() {
8      var ketinggianTotal int = 0
9      var inputTambahkan int
10
11     for {
12         switch {
13             case ketinggianTotal == 0:
14                 fmt.Print("Masukkan ketinggian pohon wok (N > 8): ")
15             default:
16                 fmt.Printf("yah kurang wok, cuma %d (masih <= 8), tambahan lagi wok: ", ketinggianTotal)
17         }
18
19         _, err := fmt.Scan(&inputTambahkan)
20
21         if err != nil {
22             fmt.Println("\nInput ngga valid. Pastiin inputnya cuma angka.")
23             var discard string
24             fmt.Scanln(&discard)
25             continue
26         }
27
28         ketinggianTotal += inputTambahkan
29
30         if ketinggianTotal > 8 {
31             fmt.Printf("\nKetinggian totalnya %d yah. otw bikin...\\n\\n", ketinggianTotal)
32             break
33         }
34     }
35
36     for i := 1; i <= ketinggianTotal; i++ {
37         for j := 0; j < ketinggianTotal-i; j++ {
38             fmt.Print(" ")
39         }
40
41         for k := 0; k < 2*i-1; k++ {
42             fmt.Print("*")
43         }
44
45         fmt.Println()
46     }
47
48     for i := 0; i < 3; i++ {
49         for j := 0; j < ketinggianTotal-1; j++ {
50             fmt.Print(" ")
51         }
52         fmt.Println("|")
53     }
54
55     fmt.Println("\n--- Code Execution Successful ---")
56 }

```

The terminal window shows the execution of the program. It prompts for input, handles invalid input, calculates the total height, and prints a diamond pattern. The output includes the total height, a warning message, and the diamond pattern itself.

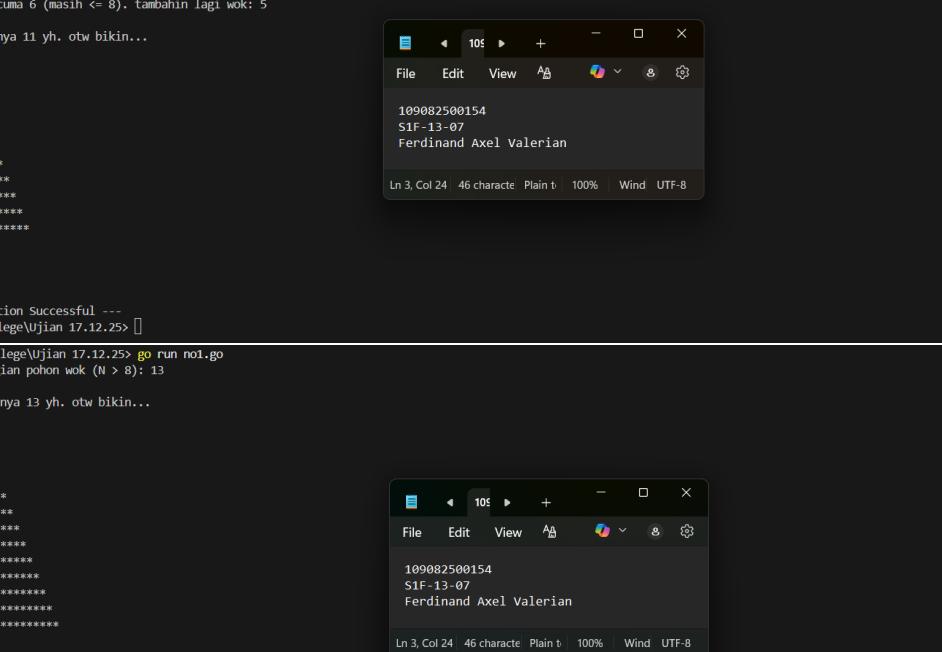
PROBLEMS 2 OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS D:\vscode\College\Ujian 17.12.25> go run no1.go  
Masukkan ketinggian pohon wok (N > 8): 6  
yah kurang wok, cuma 6 (masih <= 8). tambahin lagi wok: 5  
Ketinggian totalnya 11 yh. otw bikin...  
\*  
\*\*\*  
\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*  
|  
- - - Code Execution Successful ---

PS D:\vscode\College\Ujian 17.12.25> [ ]

PS D:\vscode\College\Ujian 17.12.25> go run no1.go  
Masukkan ketinggian pohon wok (N > 8): 13  
Ketinggian totalnya 13 yh. otw bikin...  
\*  
\*\*\*  
\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*  
\*\*\*\*\*  
|  
- - - Code Execution Successful ---

PS D:\vscode\College\Ujian 17.12.25> [ ]



## Deskripsi program

Si udin baru jadi ni bikinin programnya, otw pamer rek ke mas bahlil. Nah si bahlil kagum ni rek sama programnya, trus minta lah coba ketinggian 6, tapi si *switch-case* nya ngebaca kurang dari 8, jadinya ngeluarin teks kedua buat minta si bahlil nambahin angka. Si bahlil nyoba buat nambahin lagi 5, dan program bagian *if-then* yang menyatakan ketinggian total dari hasil ketinggian awal sama ketinggian tambahan(kalo misalkan ketinggian awal kurang, kalo ga kurang juga sama aja). Program akan menghitung menggunakan *for-loop* buat bikin pohon natalnya, pake “,”, “\*”, buat pohonnya dan “|” buat batangnya

## 2. SOAL 2

## Source Code

```
package main

import (
    "fmt"
)

func main() {
```

```
menu := []string{
    "Empal Gentong Biasa",
    "Empal Asem",
    "Sate Kambing Muda",
    "Nasi Lengko Pagonan",
}

harga := []int{
    22000,
    22000,
    40000,
    15000,
}

fmt.Println("==== MENU EMPAL GENTONG MAS FUAD ====")

for i := 0; i < len(menu); i++ {
    fmt.Printf("%d. %s - Rp%d\n", i+1, menu[i], harga[i])
}

fmt.Println("-----")

var pilihanMenu int

var jumlahBeli int

var totalHargaFinal int

for {
    fmt.Print("Pilih menu (1-4): ")
    _, err := fmt.Scan(&pilihanMenu)
```

```
switch {

    case err != nil:

        fmt.Println("\nInputnya ngga valid. Pastiinn
maneh masukkin angka.")

    case pilihanMenu >= 1 && pilihanMenu <= 4:

        goto LanjutJumlahBeli

    default:

        fmt.Println("\nInputnya ngga valid. Masukkan
nomor menu antara 1 sampai 4.")

    }

    var discard string

    fmt.Scanln(&discard)

    continue

}

LanjutJumlahBeli:

for {

    fmt.Print("Masukkan jumlah beli: ")

    _, err := fmt.Scan(&jumlahBeli)

    if err != nil || jumlahBeli <= 0 {

        fmt.Println("\nInputnya ngga valid. Jumlah beli
harus angka positif.")

        var discard string

        fmt.Scanln(&discard)

        continue

    }

    break
}
```

```
}

indexPilihan := pilihanMenu - 1

hargaSatuan := harga[indexPilihan]

totalHarga := hargaSatuan * jumlahBeli

totalHargaFinal = totalHarga

fmt.Println("\n==== STRUK PEMBAYARAN ===")

fmt.Printf("Menu      : %s\n", menu[indexPilihan])

fmt.Printf("Harga     : Rp %d\n", hargaSatuan)

fmt.Printf("Jumlah    : %d\n", jumlahBeli)

fmt.Printf("Subtotal: Rp %d\n", totalHarga)

fmt.Println("-----")

fmt.Printf("TOTAL BAYAR: Rp %d\n", totalHargaFinal)

fmt.Println("-----")

fmt.Println("\n--- Code Execution Successful ---")

}
```

### Screenshot program

```

1  no2.go > main
2
3  package main
4
5  import (
6      "fmt"
7  )
8
9  func main() {
10    menu := []string{
11        "Empal Gentong Biasa",
12        "Empal Asem",
13        "Sate Kambing Muda",
14        "Nasi Lengko Pagonan",
15    }
16    harga := []int{
17        22000,
18        22000,
19        40000,
20        15000,
21    }
22    fmt.Println("==== MENU EMPAL GENTONG MAS FUAD ===")
23    for i := 0; i < len(menu); i++ {
24        fmt.Printf("%d. %s - Rp%d\n", i+1, menu[i], harga[i])
25    }
26    fmt.Println("-----")
27
28    var pilihanMenu int
29    var jumlahBeli int
30    var totalHargaFinal int
31
32    for {
33        fmt.Print("Pilih menu (1-4): ")
34        _, err := fmt.Scan(&pilihanMenu)
35
36        switch {
37            case err != nil:
38                fmt.Println("\nInputnya ngga valid. Pastiinn maneh masukkin angka.")
39            case pilihanMenu >= 1 && pilihanMenu <= 4:
40                goto LanjutJumlahBeli
41
42            default:
43                fmt.Println("\nInputnya ngga valid. Masukkan nomor menu antara 1 sampai 4.")
44        }
45        var discard string
46        fmt.Scanln(&discard)
47        continue
48    }
49
50    LanjutJumlahBeli:
51    for {
52        fmt.Print("Masukkan jumlah beli: ")
53        _, err := fmt.Scan(&jumlahBeli)
54
55        if err != nil || jumlahBeli <= 0 {
56            fmt.Println("\nInputnya ngga valid. Jumlah beli harus angka positif.")
57            var discard string
58            fmt.Scanln(&discard)
59            continue
60        }
61        break
62    }
63
64    indexPilihan := pilihanMenu - 1
65    hargaSatuan := harga[indexPilihan]
66    totalHarga := hargaSatuan * jumlahBeli
67    totalHargaFinal = totalHarga
68
69    fmt.Println("==== STRUK PEMBAYARAN ===")
70    fmt.Printf("Menu [%d]: %s\n", menu[indexPilihan])
71    fmt.Printf("Harga [%d]: Rp %d\n", hargaSatuan)
72    fmt.Printf("Jumlah [%d]: %d\n", jumlahBeli)
73    fmt.Printf("Subtotal: Rp %d\n", totalHarga)
74    fmt.Println("-----")
75    fmt.Println("TOTAL BAYAR: Rp %d\n", totalHargaFinal)
76    fmt.Println("-----")
77
78    fmt.Println("\n--- Code Execution Successful ---")
}

```

The code is a Go program that prints a menu of four items (Empal Gentong Biasa, Empal Asem, Sate Kambing Muda, Nasi Lengko Pagonan) with their respective prices (Rp 22,000, Rp 22,000, Rp 40,000, Rp 15,000). It then prompts the user to select a menu item (1-4) and enter the quantity. The program calculates the total price and prints a receipt.

**Output 1:**

```

109082500154
S1F-13-07
Ferdinand Axel Valerian|
```

**Output 2:**

```

109082500154
S1F-13-07
Ferdinand Axel Valerian|
```

**Terminal Log:**

```

PS D:\vscode\College\Ujian 17.12.25> go run no2.go
==== MENU EMPAL GENTONG MAS FUAD ===
1. Empal Gentong Biasa - Rp22000
2. Empal Asem - Rp22000
3. Sate Kambing Muda - Rp40000
4. Nasi Lengko Pagonan - Rp15000
-----
Pilih menu (1-4): 2
Masukkan jumlah beli: 2

==== STRUK PEMBAYARAN ===
Menu : Empal Asem
Harga : Rp 22000
Jumlah : 2
Subtotal: Rp 44000
-----
TOTAL BAYAR: Rp 44000
-----

--- Code Execution Successful ---
PS D:\vscode\College\Ujian 17.12.25> 
```

### **Deskripsi program**

Sir resbob fan gogh laper ni wok, dia otw ke warung makan empal gentong mas fuad. Bes langsung aja dia order 2 porsi empal asem. program ngeloop sama *switch-case* buat si resbob milih menu pake angka, trus program menggunakan if then buat masukkin jumlah belinya, trus algoritmanya ngitung dan terakhir nyetak struknya buat ngasih info kalo sir resbob belinya makanan apa, berapa porsi dan total harga

### **3. SOAL 3**

#### **Source Code**

```
package main

import (
    "fmt"
    "os"
)

func main() {
    var a, b, c float64

    fmt.Println("==== Penentu Jenis Segitiga ====")

    for {
        fmt.Print("Masukkan tiga sisi (a b c): ")
        _, err := fmt.Scan(&a, &b, &c)

        if err != nil || a <= 0 || b <= 0 || c <= 0 {
            fmt.Println("\nInputnya ngga valid wok. Pastiin
semua sisi bilangan bulat positif.")

            var discard string
            fmt.Scanln(&discard)

            continue
        }

        break
    }
}
```

```
if a+b <= c || a+c <= b || b+c <= a {  
    fmt.Println("\nBukan segitiga")  
    os.Exit(0)  
}  
  
ia, ib, ic := int(a), int(b), int(c)  
  
isSikuSiku := (ia*ia + ib*ib == ic*ic) ||  
    (ia*ia + ic*ic == ib*ib) ||  
    (ib*ib + ic*ic == ia*ia)  
  
if ia == ib && ib == ic {  
    fmt.Println("\nSegitiga sama sisi")  
} else if isSikuSiku {  
    isSamaKaki := (ia == ib) || (ia == ic) || (ib ==  
        ic)  
  
    if isSamaKaki {  
        fmt.Println("\nSegitiga siku-siku, Segitiga sama  
        kaki")  
    } else {  
        fmt.Println("\nSegitiga siku-siku")  
    }  
} else if ia == ib || ia == ic || ib == ic {  
    fmt.Println("\nSegitiga sama kaki")  
  
} else {  
    fmt.Println("\nSegitiga sembarang")  
}
```

```

    fmt.Println("\n--- Code Execution Successful ---")

}

```

## Screenshot program

The screenshot shows a Go code editor with a file named no3.go. The code defines a main package with a main function. It prompts the user for three side lengths (a, b, c) and checks if they are valid positive integers. It then checks if the triangle is equilateral, isosceles, or scalene. Finally, it checks if the triangle is right-angled by comparing the square of each side with the sum of the other two sides.

```

no3.go > main
1 package main
2
3 import (
4     "fmt"
5     "os"
6 )
7
8 func main() {
9     var a, b, c float64
10
11     fmt.Println("==> Penentu Jenis Segitiga ==>")
12
13     for {
14         fmt.Print("Masukkan tiga sisi (a b c): ")
15         _, err := fmt.Scan(&a, &b, &c)
16
17         if err != nil || a <= 0 || b <= 0 || c <= 0 {
18             fmt.Println("\nInputnya ngga valid wok. Pastiin semua sisi bilangan bulat positif.")
19             var discard string
20             fmt.Scanln(&discard)
21             continue
22         }
23         break
24     }
25
26     if a+b <= c || a+c <= b || b+c <= a {
27         fmt.Println("\nBukan segitiga")
28         os.Exit(0)
29     }
30
31     ia, ib, ic := int(a), int(b), int(c)
32
33     isSikuSiku := (ia*ia + ib*ib == ic*ic) ||
34                 (ia*ia + ic*ic == ib*ib) ||
35                 (ib*ib + ic*ic == ia*ia)
36
37     if ia == ib && ib == ic {
38         fmt.Println("\nSegitiga sama sisi")
39     }
40     } else if isSikuSiku {
41         isSamaKaki := (ia == ib) || (ia == ic) || (ib == ic)
42
43         if isSamaKaki {
44             fmt.Println("\nSegitiga siku-siku, Segitiga sama kaki")
45         } else {
46             fmt.Println("\nSegitiga siku-siku")
47         }
48
49     } else if ia == ib || ia == ic || ib == ic {
50         fmt.Println("\nSegitiga sama kaki")
51     } else {
52         fmt.Println("\nSegitiga sembarang")
53     }
54
55     fmt.Println("\n--- Code Execution Successful ---")
56 }

```

The terminal window shows the command go run no3.go being run. It then prompts for three side lengths (a, b, c). When 3 3 3 is entered, it outputs "Segitiga sama sisi". When 5 5 8 is entered, it outputs "Segitiga sama kaki". When 3 4 5 is entered, it outputs "Segitiga siku-siku". When 4 5 6 is entered, it outputs "Segitiga sembarang". When 123 is entered, it outputs "Bukan segitiga".

```

PROBLEMS 4 OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS D:\vscode\College\Ujian 17.12.25> go run no3.go
==> Penentu Jenis Segitiga ==
Masukkan tiga sisi (a b c): 3 3 3

Segitiga sama sisi
● PS D:\vscode\College\Ujian 17.12.25> go run no3.go
==> Penentu Jenis Segitiga ==
Masukkan tiga sisi (a b c): 5 5 8

Segitiga sama kaki
● PS D:\vscode\College\Ujian 17.12.25> go run no3.go
==> Penentu Jenis Segitiga ==
Masukkan tiga sisi (a b c): 3 4 5

Segitiga siku-siku
● PS D:\vscode\College\Ujian 17.12.25> go run no3.go
==> Penentu Jenis Segitiga ==
Masukkan tiga sisi (a b c): 4 5 6

Segitiga sembarang
● PS D:\vscode\College\Ujian 17.12.25> go run no3.go
==> Penentu Jenis Segitiga ==
Masukkan tiga sisi (a b c): 123
1 2 3

Bukan segitiga
* PS D:\vscode\College\Ujian 17.12.25>

```

### **Deskripsi program**

Jadi program ini itu buat ngeklarifikasi jenis segitiga. For {} dipake sebagai *repeat-until*, minta user buat masukkan ketiga angka sisi, dan *if-then* bakal bekerja kalo user masukkan angka bilangan bulat bernilai negatif dan meminta user buat input ulang. Ketika user memasukkan angka yang tepat sesuai permintaan maka algoritma bakal menghitung ketiga sisi dan *else-if if-then* akan menyatakan termasuk jenis segitiga apakah itu, kecuali Panjang sisi 1 2 3 karena itu emang bukan segitiga