

LAPORAN PRAKTIKUM
Algoritma Pemrograman

EVALUASI



Disusun oleh:

YEDIJA JOHANAN SIREGAR

109082500075

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 ganjil int

    fmt.Print("Input bilangan ganjil: ")
    fmt.Scan(&ganjil)

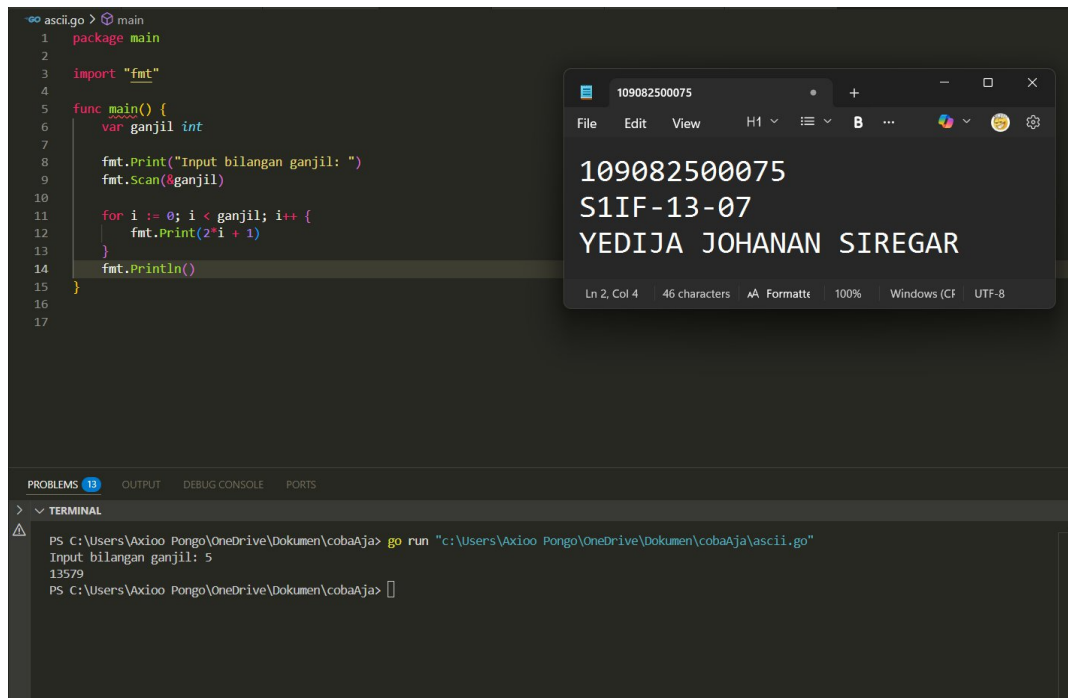
    for i := 0; i < ganjil; i++ {
        fmt.Print(2*i + 1)
    }

    fmt.Println()
}
```

Screenshoot program

//tambahkan tangkapan layar dari program (boleh lebih dari 1 jika diperlukan)

CONTOH TANGKAPAN LAYAR: (GUNAKAN NOTEPAD)



The screenshot shows a Go program in an IDE. The code defines a `main` function that prompts the user for an odd number, then uses a `for` loop to print the first 10 odd numbers. The terminal output shows the program running successfully, with the input `5` and the output of the first 10 odd numbers.

```
1 package main
2
3 import "fmt"
4
5 func main() {
6     var ganjil int
7
8     fmt.Print("Input bilangan ganjil: ")
9     fmt.Scan(&ganjil)
10
11     for i := 0; i < ganjil; i++ {
12         fmt.Print(2*i + 1)
13     }
14     fmt.Println()
15 }
16
17
```

Terminal Output:

```
PS C:\Users\Axioo Pongo\OneDrive\Dokumen\cobaAja> go run "c:\Users\Axioo Pongo\OneDrive\Dokumen\cobaAja\ascii.go"
Input bilangan ganjil: 5
13579
PS C:\Users\Axioo Pongo\OneDrive\Dokumen\cobaAja>
```

Deskripsi program

Kode diatas ini meminta program untuk mengidentifikasi sebuah bilangan yang di inputkan oleh user merupakan bilangan ganjil secara berurutan, dengan kita memanggil sebuah for loop untuk membuat bilangan tersebut melanjutkan bilangan ganjil yang di inputkan.

2. SOAL 2

Source Code

```
package main

import "fmt"

func main() {

    var x, y int

    fmt.Print("Masukkan hari x dan y: ")

    fmt.Scan(&x, &y)
```

```
        if x > y {  
            fmt.Println("Hari x harus kurang dari atau sama  
dengan y")  
        }  
  
        bakteri := 1  
        for i := x; i <= y; i++ {  
            if i >= x {  
                bakteri *= i  
            }  
        }  
  
        fmt.Printf("Jumlah bakteri awal adalah: %d\n",  
bakteri)  
    }
```

Screenshoot program

```
1 package main
2
3 import "fmt"
4
5 func main() {
6     var x, y int
7     fmt.Print("Masukkan hari x dan y: ")
8     fmt.Scan(&x, &y)
9
10    if x > y {
11        fmt.Println("Hari x harus kurang dari atau sama dengan y")
12    }
13
14    bakteri := 1
15    for i := x; i <= y; i++ {
16        if i >= x {
17            bakteri *= i
18        }
19    }
20
21    fmt.Printf("Jumlah bakteri awal adalah: %d\n", bakteri)
22 }
23 }
```

109082500075

109082500075
S1IF-13-07
YEDIJA JOHANAN SIREGAR

Ln 2, Col 4 46 characters AA Formatt 100% Windows (CF

PROBLEMS 15 OUTPUT DEBUG CONSOLE PORTS

TERMINAL

PS C:\Users\Axioo Pongo\OneDrive\Dokumen\cobaAja> go run ./ascii.go
Masukkan hari x dan y: 2 4
Jumlah bakteri awal adalah: 24
PS C:\Users\Axioo Pongo\OneDrive\Dokumen\cobaAja> go run ./ascii.go
Masukkan hari x dan y: 3 5
Jumlah bakteri awal adalah: 60
PS C:\Users\Axioo Pongo\OneDrive\Dokumen\cobaAja>

Deskripsi program

Program ini diminta untuk menghitung sebuah jumlah bakteri terakhir dari hari x sampai y, dengan cara kita pakai yang if $x > y$, untuk menunjukkan bahwa variabel x itu harus lebih kecil/kurang dari, dan juga kita memakai perulangan untuk membuat logika dalam sebuah pemrograman ini, contohnya jika kita input 2 dan 4, maka logika nya adalah $2 \cdot 3 \cdot 4$, maka output yang dihasilkan adalah 24.

3. SOAL 3

Source Code

```
package main

import "fmt"

func main() {

    var uang int

    fmt.Print("Masukkan jumlah uang: ")

    fmt.Scan(&uang)
```

```
    peti := uang / 800

    sisa := uang % 800

    karung := sisa / 100

    sisa = sisa % 100

    ikat := sisa / 10

    keping := sisa % 10

    fmt.Printf("%d %d %d %d\n", peti, karung, ikat,
keping)

    fmt.Println("Hasil dalam bentuk peti: ", peti)

    fmt.Println("Hasil dalam bentuk karung: ", karung)

    fmt.Println("Hasil dalam bentuk ikat: ", ikat)

    fmt.Println("Hasil dalam bentuk keping: ", keping)

}
```

Screenshoot program

The image shows a Go program in an IDE (VS Code) and its execution output in a terminal window. The program is a simple calculator that takes an input amount of money and calculates how many 'peti' (boxes), 'karung' (bags), 'ikat' (bundles), and 'keping' (coins) can be made from it, based on fixed ratios.

```
1 package main
2
3 import "fmt"
4
5 func main() {
6     var uang int
7     fmt.Print("Masukkan jumlah uang: ")
8     fmt.Scan(&uang)
9
10    peti := uang / 800
11    sisa := uang % 800
12
13    karung := sisa / 100
14    sisa = sisa % 100
15
16    ikat := sisa / 10
17    keping := sisa % 10
18
19    fmt.Printf("%d %d %d\n", peti, karung, ikat, keping)
20    fmt.Println("Hasil dalam bentuk peti: ", peti)
21    fmt.Println("Hasil dalam bentuk karung: ", karung)
22    fmt.Println("Hasil dalam bentuk ikat: ", ikat)
23    fmt.Println("Hasil dalam bentuk keping: ", keping)
24 }
```

The terminal output shows the program running successfully for two different inputs:

```
PS C:\Users\Aldoo Pongo\OneDrive\Documents\cobatja> go run ./ascii.go
Masukkan jumlah uang: 800
1 0 0 0
Hasil dalam bentuk peti: 1
Hasil dalam bentuk karung: 0
Hasil dalam bentuk peti: 1
Hasil dalam bentuk peti: 1
Hasil dalam bentuk peti: 1
Hasil dalam bentuk peti: 1
Hasil dalam bentuk peti: 1
Hasil dalam bentuk karung: 0
Hasil dalam bentuk ikat: 0
Hasil dalam bentuk keping: 0
PS C:\Users\Aldoo Pongo\OneDrive\Documents\cobatja> go run ./ascii.go
Masukkan jumlah uang: 1053
1 2 5 3
Hasil dalam bentuk peti: 1
Hasil dalam bentuk karung: 2
Hasil dalam bentuk ikat: 5
Hasil dalam bentuk keping: 3
PS C:\Users\Aldoo Pongo\OneDrive\Documents\cobatja>
```

The output window shows the final result of the program:

```
109082500075
S1IF-13-07
YEDIJA JOHANAN SIREGAR
```

Deskripsi program

Kode diatas ini diminta untuk mengetahui berapa banyak peti, karung, ikat, dan keping yang setara dengan barang yang pedagang itu punya, contohnya jika pedagang tersebut punya 1 peti maka terdapat 8 karung di dalamnya, dan cara program bisa mengetahui berapa banyak peti, karung, ikat dan keping tersebut, dengan cara kita memunculkan perhitungan satu peti setara dengan 8 karung, satu karung berisi 10 ikat, dan satu ikat berisi 10 keping. Jika dikalikan, maka 1 peti sama dengan $8 \times 10 \times 10 = 800$ keping. Oleh karena itu, untuk mengetahui berapa banyak peti yang bisa dibentuk dari sejumlah keping yang dimiliki, program membagi jumlah keping dengan 800 ($\text{peti} := \text{uang} / 800$).