

LAPORAN PRAKTIKUM
Algoritma Pemrograman

EVALUASI



Disusun oleh:

Didi Hermawanto

109082500088

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 n int

    fmt.Print("Masukkan jumlah bilangan genap: ")

    fmt.Scanln(&n)

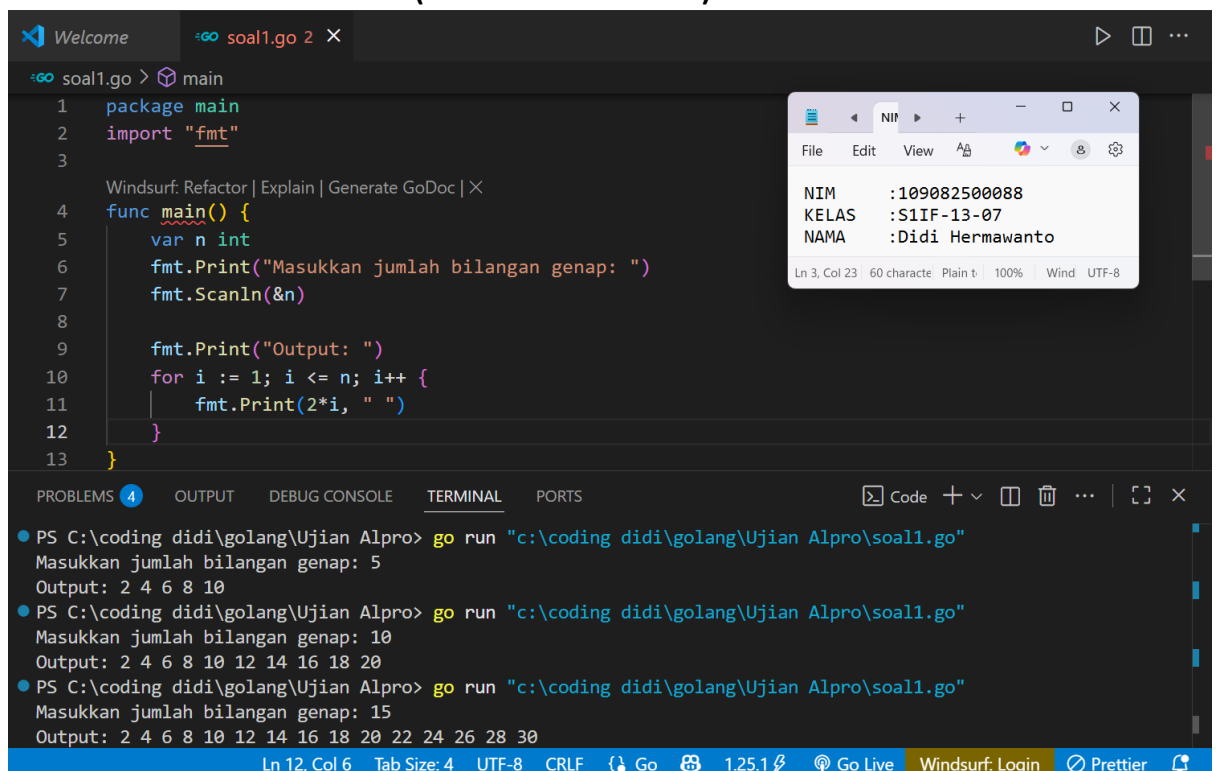
    fmt.Print("Output: ")

    for i := 1; i <= n; i++ {
        fmt.Print(2*i, " ")
    }
}
```

Screenshoot program

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

CONTOH TANGKAPAN LAYAR: (GUNAKAN NOTEPAD)



Deskripsi program

Program ini digunakan untuk menampilkan deret bilangan genap sebanyak jumlah yang dimasukkan oleh pengguna. Di awal program, variabel *n* dibuat untuk menyimpan angka yang diketik oleh pengguna melalui perintah `fmt.Scanln(&n)`. Setelah pengguna memasukkan nilai, program akan menuliskan kata "Output:" sebagai penanda hasil. Lalu program menjalankan perulangan `for i := 1; i <= n; i++` yang berarti proses akan diulang dari angka 1 sampai angka yang dimasukkan pengguna. Setiap kali perulangan berjalan, nilai *i* dikalikan dengan 2 agar menghasilkan bilangan genap, kemudian hasilnya langsung ditampilkan di layar dengan jarak spasi di antara tiap angka. Jadi, kalau pengguna memasukkan angka 5, maka program akan mencetak deret "2 4 6 8 10". Secara sederhana, alurnya dimulai dari membaca input pengguna, lalu menghitung bilangan genap satu per satu menggunakan rumus $2*i$, dan mencetak hasilnya secara berurutan sampai jumlahnya sesuai dengan input.

2. SOAL 2

Source Code

```
package main

import "fmt"

func main() {
    var x, y int
    fmt.Print("Masukkan dua bilangan (x y): ")
    fmt.Scan(&x, &y)

    hasil := 1
    for i := x; i <= y; i++ {
        hasil *= i
    }

    fmt.Println("Jumlah bakteri terakhir:", hasil)
}
```

Screenshoot program

```
1 package main
2 import "fmt"
3
4 func main() {
5     var x, y int
6     fmt.Print("Masukkan dua bilangan (x y): ")
7     fmt.Scan(&x, &y)
8
9     hasil := 1
10    for i := x; i <= y; i++ {
11        hasil *= i
12    }
13}
```

PROBLEMS 4 OUTPUT DEBUG CONSOLE TERMINAL PORTS

Masukkan dua bilangan (x y): 2 4
Jumlah bakteri terakhir: 24

PS C:\coding didi\golang\Ujian Alpro> go run "c:\coding didi\golang\Ujian Alpro\soal2.go"

Masukkan dua bilangan (x y): 3 5
Jumlah bakteri terakhir: 60

PS C:\coding didi\golang\Ujian Alpro> go run "c:\coding didi\golang\Ujian Alpro\soal2.go"

Masukkan dua bilangan (x y): 4 6
Jumlah bakteri terakhir: 120

PS C:\coding didi\golang\Ujian Alpro>

Ln 16, Col 1 Tab Size: 4 UTF-8 CRLF Go 1.25.1 Go Live Windsurf: Login Prettier

Deskripsi program

Program ini digunakan untuk menghitung hasil perkalian dari dua bilangan yang dimasukkan oleh pengguna. Di awal, program meminta pengguna mengetik dua angka melalui pesan “Masukkan dua bilangan (x y):” dan menyimpannya ke dalam variabel x dan y. Setelah itu, variabel hasil diisi dengan angka 1 sebagai nilai awal. Program kemudian menjalankan perulangan dari nilai x sampai y, di mana setiap angka dalam rentang tersebut dikalikan satu per satu ke dalam variabel hasil. Jadi, hasil akan terus bertambah nilainya karena setiap kali perulangan berjalan, nilainya dikalikan dengan angka berikutnya. Setelah proses selesai, program menampilkan hasil akhirnya dengan tulisan “Jumlah bakteri terakhir:”. Alurnya sederhana: masukkan dua angka, kalikan semua angka dari batas pertama sampai batas kedua, lalu tampilkan hasil perkaliannya di layar.

3. SOAL 3

Source Code

```
package main

import "fmt"

func main() {
    var keping int
```

```
    fmt.Print("Masukkan jumlah keping: ")
    fmt.Scan(&keping)

    kepingPerIkat := 8
    ikatPerKarung := 10
    karungPerPeti := 10

    kepingPerKarung := kepingPerIkat * ikatPerKarung
    kepingPerPeti := kepingPerKarung * karungPerPeti

    peti := keping / kepingPerPeti
    sisa := keping % kepingPerPeti

    karung := sisa / kepingPerKarung
    sisa %= kepingPerKarung

    ikat := sisa / kepingPerIkat
    sisa %= kepingPerIkat

    fmt.Printf("%d peti, %d karung, %d ikat, dan %d keping\n", peti, karung, ikat, sisa)
}
```

Screenshoot program

```
13 kepingPerKarung := kepingPerIkut * ikatPerKarung
14 kepingPerPeti := kepingPerKarung * karungPerPeti
15
16 peti := keping / kepingPerPeti
17 sisa := keping % kepingPerPeti
18
19 karung := sisa / kepingPerKarung
20 sisa %= kepingPerKarung
21
22 ikat := sisa / kepingPerIkut
23 sisa %= kepingPerIkut
24
25 fmt.Printf("%d peti, %d karung, %d ikat, dan %d keping\n", peti, karung, ikat, sisa)
```

```
PS C:\coding didi\golang\Ujian Alpro> go run "c:\coding didi\golang\Ujian Alpro\soal3.go"
Masukkan jumlah keping: 800
1 peti, 0 karung, 0 ikat, dan 0 keping
PS C:\coding didi\golang\Ujian Alpro> go run "c:\coding didi\golang\Ujian Alpro\soal3.go"
Masukkan jumlah keping: 1053
1 peti, 3 karung, 1 ikat, dan 5 keping
PS C:\coding didi\golang\Ujian Alpro> go run "c:\coding didi\golang\Ujian Alpro\soal3.go"
Masukkan jumlah keping: 1250
1 peti, 5 karung, 6 ikat, dan 2 keping
```

Deskripsi program

Program ini berfungsi untuk mengubah jumlah keping menjadi satuan yang lebih besar, yaitu ikat, karung, dan peti. Pertama, pengguna memasukkan jumlah keping, lalu program menghitung berapa banyak peti, karung, dan ikat yang bisa dibentuk berdasarkan ketentuan: 1 ikat = 8 keping, 1 karung = 10 ikat, dan 1 peti = 10 karung. Program membagi jumlah keping dari satuan terbesar ke terkecil, lalu menampilkan hasilnya dalam bentuk “peti, karung, ikat, dan keping”. Jadi, pengguna bisa langsung tahu hasil konversinya dari jumlah keping yang dimasukkan.

Nilai keping yang dimasukkan pengguna disimpan ke variabel keping. Kemudian program menentukan nilai konversi antar satuan seperti kepingPerIkut, ikatPerKarung, dan karungPerPeti. Setelah itu, dihitung total keping dalam satu karung dan satu peti menggunakan perkalian antar satuan. Program kemudian membagi jumlah keping yang diinput dengan jumlah keping per peti untuk mengetahui berapa banyak peti. Sisa dari pembagian tersebut digunakan untuk menghitung jumlah karung, kemudian ikat, dan terakhir keping yang tersisa. Semua hasil tersebut kemudian ditampilkan ke layar.