

LAPORAN PRAKTIKUM
Algoritma Pemrograman

EVALUASI



Disusun oleh:

BAGUS IRSYAD KAMAL

109082500215

S1IF-13-02

PROGRAM STUDI S1 INFORMATIKA
FAKULTAS INFORMATIKA
TELKOM UNIVERSITY PURWOKERTO
2025

SOAL

1. SOAL 1

Source Code

```
package main

import "fmt"

func main(){

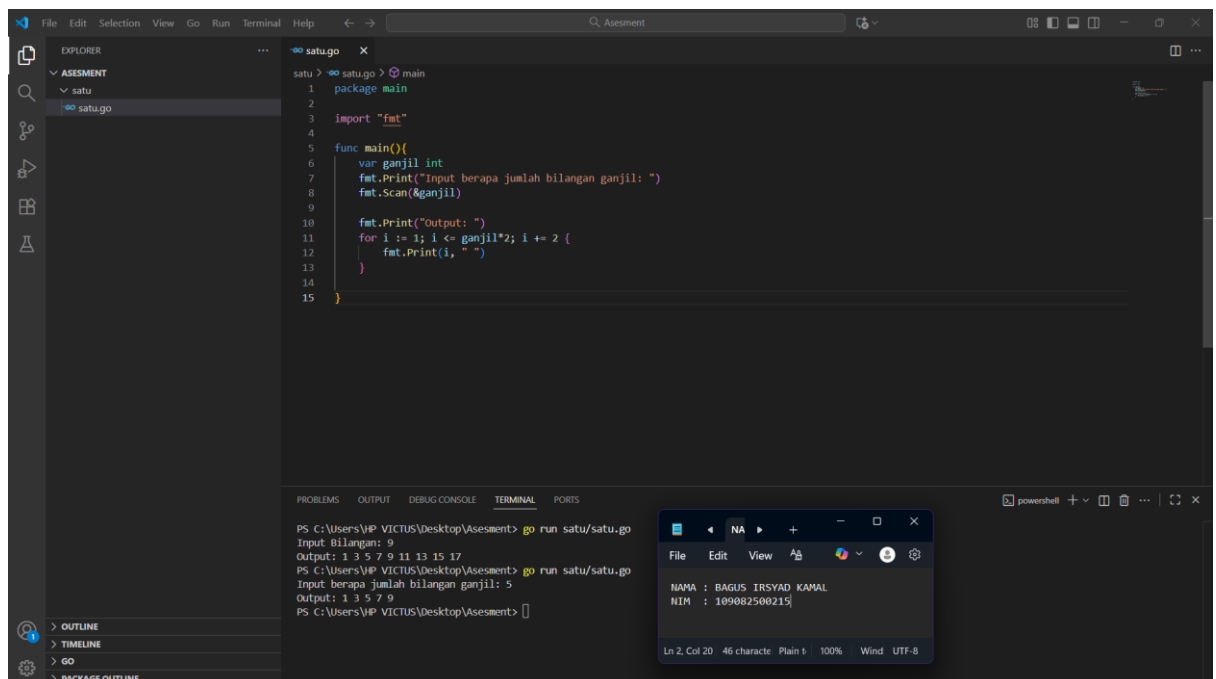
    var ganjil int

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

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

}
```

Screenshoot program



Deskripsi program

Program ini berfungsi untuk menampilkan deret bilangan ganjil sebanyak jumlah yang dimasukkan oleh pengguna. Pertama, program meminta pengguna untuk memasukkan berapa banyak bilangan ganjil yang ingin ditampilkan. Nilai tersebut disimpan dalam variabel bernama **ganjil**.

Setelah itu, program mencetak tulisan "Output:" di layar, lalu menjalankan perulangan mulai dari angka 1. Nilai awal 1 dipilih karena merupakan bilangan ganjil pertama. Setiap kali perulangan berjalan, nilainya akan bertambah 2, sehingga angka yang muncul selalu bilangan ganjil seperti 1, 3, 5, dan seterusnya.

Kondisi perulangannya diatur agar berhenti ketika nilai sudah mencapai dua kali lipat dari jumlah bilangan ganjil yang diinginkan. Dengan begitu, jumlah bilangan ganjil yang dicetak sesuai dengan input pengguna.

Sebagai contoh, jika pengguna memasukkan angka 5, maka program akan menampilkan deret **1 3 5 7 9**.

2. SOAL 2

Source Code

```
package main

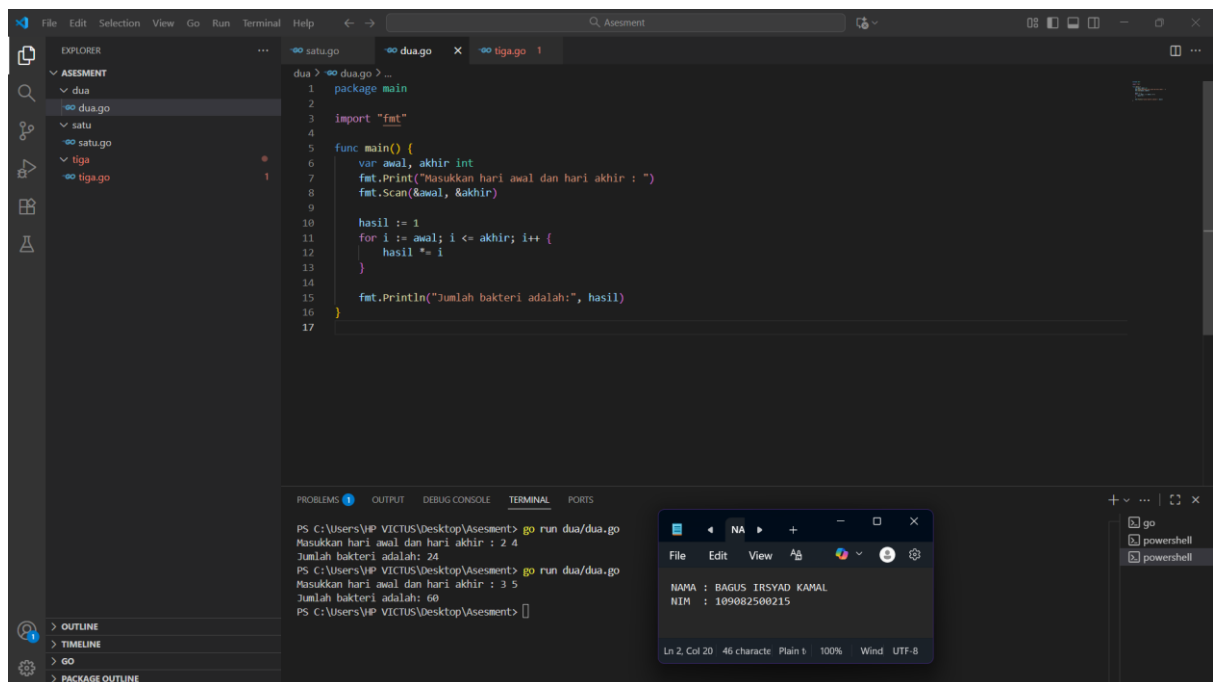
import "fmt"

func main() {
    var awal, akhir int
    fmt.Print("Masukkan hari awal dan hari akhir : ")
    fmt.Scan(&awal, &akhir)

    hasil := 1
    for i := awal; i <= akhir; i++ {
        hasil *= i
    }

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

Screenshoot program



```
1 package main
2
3 import "fmt"
4
5 func main() {
6     var awal, akhir int
7     fmt.Print("Masukkan hari awal dan hari akhir : ")
8     fmt.Scan(&awal, &akhir)
9
10    hasil := 1
11    for i := awal; i <= akhir; i++ {
12        hasil *= i
13    }
14
15    fmt.Println("Jumlah bakteri adalah:", hasil)
16 }
17
```

```
PS C:\Users\VIP VICTUS\Desktop\Asesment> go run dua/dua.go
Masukkan hari awal dan hari akhir : 2 4
Jumlah bakteri adalah: 24
PS C:\Users\VIP VICTUS\Desktop\Asesment> go run dua/dua.go
Masukkan hari awal dan hari akhir : 3 5
Jumlah bakteri adalah: 60
PS C:\Users\VIP VICTUS\Desktop\Asesment>
```

```
NAMA : BAGUS IRSYAD KAMAL
NIM : 109082500215
```

Deskripsi program

Program ini berfungsi untuk menghitung jumlah akhir bakteri yang berkembang dari hari ke-x sampai hari ke-y. Pertama, program meminta pengguna untuk memasukkan dua angka, yaitu hari awal (x) dan hari akhir (y). Nilai tersebut disimpan dalam variabel bernama awal dan akhir.

Setelah itu, program menyiapkan variabel hasil dengan nilai awal 1. Variabel ini digunakan untuk menyimpan hasil perkalian pertumbuhan bakteri. Selanjutnya, program menjalankan perulangan mulai dari hari ke-x hingga hari ke-y. Setiap kali perulangan berjalan, nilai hasil dikalikan dengan angka hari saat itu, karena pada setiap hari jumlah bakteri bertambah sesuai kelipatan harinya.

Perulangan akan berhenti ketika mencapai hari ke-y, sehingga semua nilai dari x hingga y telah dikalikan. Nilai akhir dari variabel hasil menunjukkan jumlah bakteri terakhir setelah proses pertumbuhan selesai.

Sebagai contoh, jika pengguna memasukkan 2 dan 4, maka program akan menghitung $2 \times 3 \times 4 = 24$. Jadi, jumlah bakteri terakhir setelah berkembang dari hari ke-2 hingga hari ke-4 adalah 24.

3. SOAL 3

Source Code

```
package main

import "fmt"

func main() {

    var keping int

    fmt.Print("Masukkan jumlah dalam satuan keping: ")

    fmt.Scan(&keping)

    peti := keping / 800

    sisa := keping % 800

    karung := sisa / 100

    sisa = sisa % 100

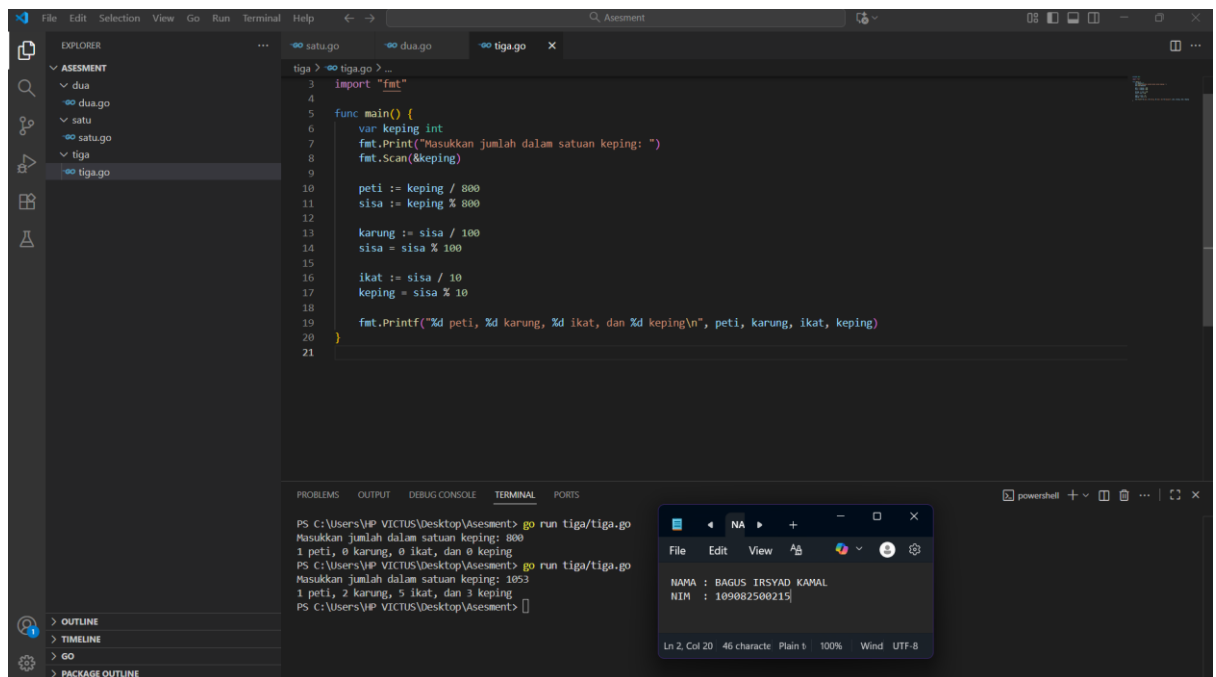
    ikat := sisa / 10

    keping = sisa % 10

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

}
```

Screenshoot program



The screenshot shows a Go IDE with the following components:

- EXPLORER:** A sidebar on the left showing a project structure with folders 'dua', 'dua.go', 'satu', 'satu.go', 'tiga', and 'tiga.go'.
- EDITOR:** The main window displays the source code of 'tiga.go'. The code is as follows:

```
1 package main
2
3 import "fmt"
4
5 func main() {
6     var keping int
7     fmt.Print("Masukkan jumlah dalam satuan keping: ")
8     fmt.Scan(&keping)
9
10    peti := keping / 800
11    sisa := keping % 800
12
13    karung := sisa / 100
14    sisa = sisa % 100
15
16    ikat := sisa / 10
17    keping = sisa % 10
18
19    fmt.Printf("%d peti, %d karung, %d ikat, dan %d keping\n", peti, karung, ikat, keping)
20 }
21
```
- TERMINAL:** A window at the bottom shows the execution of the program. The commands and output are:

```
PS C:\Users\VIP VICTUS\Desktop\Asesment> go run tiga/tiga.go
Masukkan jumlah dalam satuan keping: 800
1 peti, 0 karung, 0 ikat, dan 0 keping
PS C:\Users\VIP VICTUS\Desktop\Asesment> go run tiga/tiga.go
Masukkan jumlah dalam satuan keping: 1053
1 peti, 2 karung, 5 ikat, dan 3 keping
PS C:\Users\VIP VICTUS\Desktop\Asesment>
```
- OUTPUT WINDOW:** A small window on the right shows the output of the program, displaying the user's name and NIM:

```
NAMA : BAGUS IRSYAD KAMAL
NIM : 109082500215
```

Deskripsi program

Program ini berfungsi untuk mengubah sejumlah keping menjadi satuan yang lebih besar, yaitu peti, karung, ikat, dan keping. Pertama, program meminta pengguna untuk memasukkan jumlah keping yang dimiliki. Nilai tersebut disimpan dalam variabel bernama keping.

Setelah itu, program menghitung konversi satuan berdasarkan hubungan yang telah ditentukan, yaitu 1 peti sama dengan 8 karung, 1 karung sama dengan 10 ikat, dan 1 ikat sama dengan 10 keping. Dengan hubungan tersebut, dapat disimpulkan bahwa 1 peti setara dengan 800 keping.

Program kemudian menentukan berapa banyak peti dengan membagi jumlah keping dengan 800. Sisa keping yang belum dikonversi digunakan untuk menghitung karung dengan membaginya dengan 100. Selanjutnya, sisa dari hasil tersebut dihitung lagi untuk mencari jumlah ikat dengan pembagian 10, dan sisanya merupakan jumlah keping terakhir.

Hasil akhirnya ditampilkan dalam bentuk jumlah peti, karung, ikat, dan keping yang tersisa.

Sebagai contoh, jika pengguna memasukkan angka 1053, maka program akan menghasilkan keluaran 1 peti, 2 karung, 5 ikat, dan 3 keping.