

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

**EVALUASI**



**Disusun oleh:**

**Johanson Leeroy**

**109082500017**

**S1IF-13-04**

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

## SOAL

### 1. SOAL 1

#### Source Code

```
package main

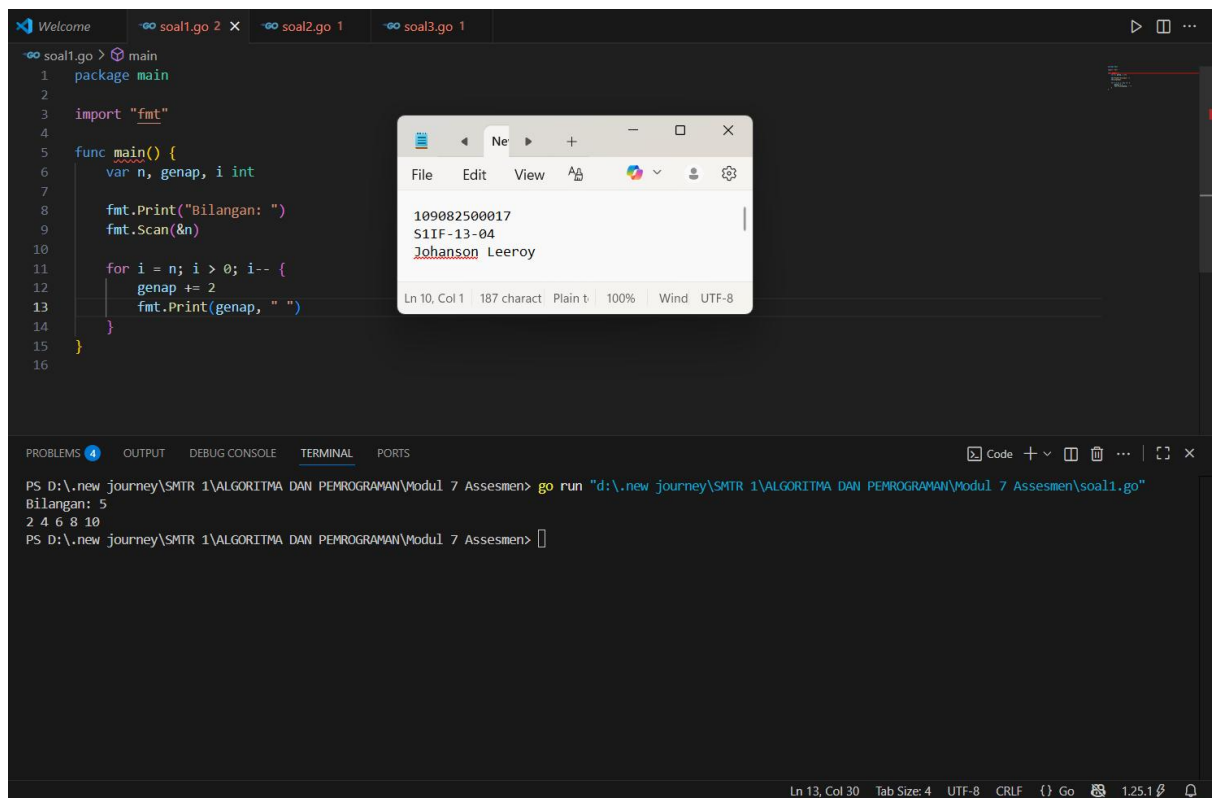
import "fmt"

func main() {
    var n, genap, i int

    fmt.Print("Bilangan: ")
    fmt.Scan(&n)

    for i = n; i > 0; i-- {
        genap += 2
        fmt.Print(genap, " ")
    }
}
```

## Screenshoot program



```
1 package main
2
3 import "fmt"
4
5 func main() {
6     var n, genap, i int
7
8     fmt.Print("Bilangan: ")
9     fmt.Scan(&n)
10
11     for i = n; i > 0; i-- {
12         genap += 2
13         fmt.Print(genap, " ")
14     }
15 }
```

PS D:\new journey\SMTR 1\ALGORITMA DAN PEMROGRAMAN\Modul 7 Assesmen> go run "d:\new journey\SMTR 1\ALGORITMA DAN PEMROGRAMAN\Modul 7 Assesmen\soal1.go"

Bilangan: 5

2 4 6 8 10

PS D:\new journey\SMTR 1\ALGORITMA DAN PEMROGRAMAN\Modul 7 Assesmen>

## Deskripsi program

1. Program diawali dengan package main, import "fmt", dan func main () {
2. Mendeklarasikan variabel n, genap, i dengan tipe data integer/int
3. User menginputkan bilangan bulat/integer ke dalam variabel n
4. Memulai for-loop dengan inisialisasi i = n ; kondisi i > 0 ; update i-- atau i = i - 1
5. Jika kondisi terpenuhi maka program akan melakukan operasi penjumlahan bilangan 2 ke dalam variabel genap
6. Selanjutnya program akan mengeluarkan output berupa deretan bilangan genap sebanyak n yang diawali dari angka 2
7. Jika kondisi sudah tidak lagi terpenuhi maka program akan berhenti

## 2. SOAL 2

### Source Code

```
package main

import "fmt"

func main() {
    var x, y, bakteri, i int

    fmt.Print("Hari pertama: ")

    fmt.Scan(&x)

    fmt.Print("hari terakhir: ")

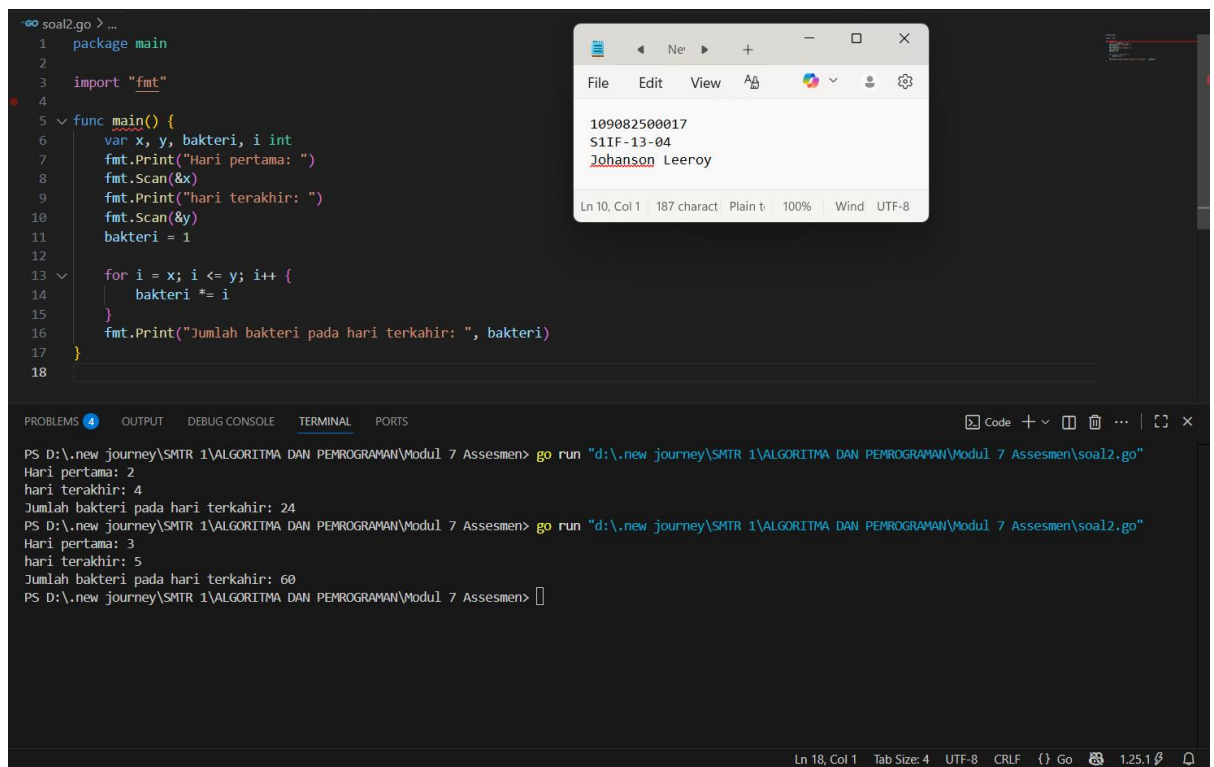
    fmt.Scan(&y)

    bakteri = 1

    for i = x; i <= y; i++ {
        bakteri *= i
    }

    fmt.Print("Jumlah bakteri pada hari terkahir: ",
    bakteri)
}
```

## Screenshoot program



The screenshot shows a Go program in a VS Code editor. The code defines a `main` function that takes two integers `x` and `y`, and a variable `bakteri` of type `int`. It prompts the user for "Hari pertama:" and "hari terakhir:". A `for` loop calculates the total number of bacteria by multiplying `bakteri` by `i` for each `i` from `x` to `y`. The final result is printed as "Jumlah bakteri pada hari terakhir: ". A small window in the foreground displays the output of the program.

```
1 package main
2
3 import "fmt"
4
5 func main() {
6     var x, y, bakteri, i int
7     fmt.Print("Hari pertama: ")
8     fmt.Scan(&x)
9     fmt.Print("hari terakhir: ")
10    fmt.Scan(&y)
11    bakteri = 1
12
13    for i = x; i <= y; i++ {
14        bakteri *= i
15    }
16    fmt.Print("Jumlah bakteri pada hari terakhir: ", bakteri)
17 }
18
```

Terminal output:

```
PS D:\new journey\SMTR 1\ALGORITMA DAN PEMROGRAMAN\Modul 7 Assesmen> go run "d:\new journey\SMTR 1\ALGORITMA DAN PEMROGRAMAN\Modul 7 Assesmen\soal2.go"
Hari pertama: 2
hari terakhir: 4
Jumlah bakteri pada hari terakhir: 24
PS D:\new journey\SMTR 1\ALGORITMA DAN PEMROGRAMAN\Modul 7 Assesmen> go run "d:\new journey\SMTR 1\ALGORITMA DAN PEMROGRAMAN\Modul 7 Assesmen\soal2.go"
Hari pertama: 3
hari terakhir: 5
Jumlah bakteri pada hari terakhir: 60
PS D:\new journey\SMTR 1\ALGORITMA DAN PEMROGRAMAN\Modul 7 Assesmen>
```

## Deskripsi program

1. Program diawali dengan `package main`, `import "fmt"`, dan `func main () {`
2. Mendeklarasikan variabel `x`, `y`, `bakteri`, `i` dengan tipe data integer/`int`
3. User menginputkan bilangan bulat/integer ke dalam variabel `x` dan `y` yang melambangkan hari pertama dan hari terakhir
4. Mendeklarasikan variabel `bakteri = 1` agar hasil perkalian tidak = 0
5. Memulai `for`-loop dengan inisialisasi `i = x` ; kondisi `i <= y` ; update `i++` atau `i = i + 1`
6. Jika kondisi terpenuhi maka variabel `bakteri` akan mengalikan dirinya dengan variabel `i`, terus seperti itu sampai kondisi tidak lagi terpenuhi
7. Jika kondisi sudah tidak lagi terpenuhi maka program akan mengeluarkan output berupa jumlah bakteri pada hari terakhir-y

### 3. SOAL 3

#### Source Code

```
package main

import "fmt"

func main() {

    var peti, karung, ikat, keping int

    fmt.Print("Keping: ")

    fmt.Scan(&keping)

    peti = keping / 800

    karung = keping % 800 / 80

    ikat = keping % 800 % 80 / 8

    keping = keping % 800 % 80 % 8

    fmt.Println(peti, "peti,", karung, "karung,", ikat,
"ikat dan", keping, "keping")

}
```

#### Screenshoot program

The screenshot shows a Go program in a file named `soal3.go` and its execution in a terminal. The program defines a `main` function that takes an integer `keping` as input and calculates the number of `peti`, `karung`, and `ikat` based on the input. The terminal shows two runs: one with `800` and one with `1053`.

```
1 package main
2
3 import "fmt"
4
5 func main() {
6     var peti, karung, ikat, keping int
7     fmt.Print("Keping: ")
8     fmt.Scan(&keping)
9
10    peti = keping / 800
11    karung = keping % 800 / 80
12    ikat = keping % 800 % 80 / 8
13    keping = keping % 800 % 80 % 8
14
15    fmt.Println(peti, "peti,", karung, "karung,", ikat, "ikat dan", keping, "keping")
16 }
17
```

Terminal output:

```
PS D:\new journey\SMTR 1\ALGORITMA DAN PEMROGRAMAN\Modul 7 Assesmen> go run "d:\new journey\SMTR 1\ALGORITMA DAN PEMROGRAMAN\Modul 7 Assesmen\soal3.go"
Keping: 800
1 peti, 0 karung, 0 ikat dan 0 keping
PS D:\new journey\SMTR 1\ALGORITMA DAN PEMROGRAMAN\Modul 7 Assesmen> go run "d:\new journey\SMTR 1\ALGORITMA DAN PEMROGRAMAN\Modul 7 Assesmen\soal3.go"
Keping: 1053
1 peti, 3 karung, 1 ikat dan 5 keping
PS D:\new journey\SMTR 1\ALGORITMA DAN PEMROGRAMAN\Modul 7 Assesmen>
```

## Deskripsi program

1. Program diawali dengan `package main`, `import "fmt"`, dan `func main () {`
2. Mendeklarasikan variabel `peti`, `karung`, `ikat`, `keping` dengan tipe data integer
3. Menginstruksikan user untuk menginputkan angka yang akan dimasukan ke dalam variabel `keping`
4. Melakukan perhitungan konversi mata uang keping ke `peti`, `karung`, dan `ikat` serta sisa dari keping itu sendiri dengan menggunakan rumus:
  - a) `peti = keping / 800`
  - b) `karung = keping % 800 / 80`
  - c) `ikat = keping % 800 % 80 / 8`
  - d) `Keping = keping % 800 % 80 % 8`
5. Output berupa konversi mata uang keping ke `peti`, `karung`, `ikat`, beserta sisanya