

**LAPORAN PRAKTIKUM ALGORITMA
DAN PEMROGRAMAN 1**

**MODUL 5 & 6
FOR-LOOP**



Disusun oleh:

JEREMY CHRISTIAN SIHOMBING

109082500178

S1IF-13-07

Asisten Praktikum

Adithana dharma putra

Apri pandu wicaksono

PROGRAM STUDI S1 INFORMATIKA

FAKULTAS INFORMATIKA

TELKOM UNIVERSITY PURWOKERTO

2025

LATIHAN KELAS – GUIDED

1. Guided 1 Source Code

```
package main

import "fmt"

func main() {

    var a, b int

    fmt.Print("Masukkan nilai a: ")

    fmt.Scan(&a)

    fmt.Print("Masukkan nilai b: ")

    fmt.Scan(&b)

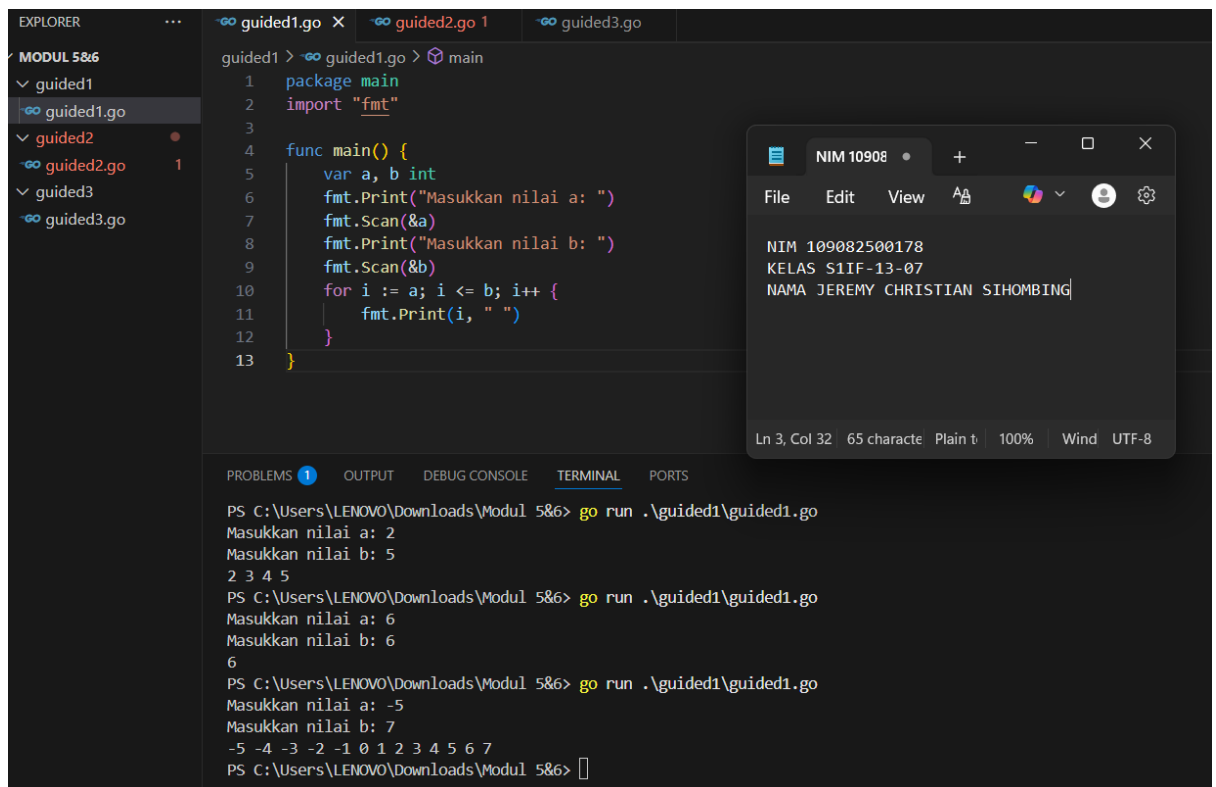
    for i := a; i <= b; i++ {

        fmt.Print(i, " ")

    }

}
```

Screenshoot program



```
guided1 > -go guided1.go > main
1 package main
2 import "fmt"
3
4 func main() {
5     var a, b int
6     fmt.Print("Masukkan nilai a: ")
7     fmt.Scan(&a)
8     fmt.Print("Masukkan nilai b: ")
9     fmt.Scan(&b)
10    for i := a; i <= b; i++ {
11        fmt.Print(i, " ")
12    }
13 }
```

File Edit View A 100% Wind UTF-8

NIM 109082500178
KELAS S1IF-13-07
NAMA JEREMY CHRISTIAN SIHOMBING

Ln 3, Col 32 65 character Plain t 100% Wind UTF-8

PROBLEMS 1 OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS C:\Users\LENOVO\Downloads\Modul 5&6> go run .\guided1\guided1.go
Masukkan nilai a: 2
Masukkan nilai b: 5
2 3 4 5
PS C:\Users\LENOVO\Downloads\Modul 5&6> go run .\guided1\guided1.go
Masukkan nilai a: 6
Masukkan nilai b: 6
6
PS C:\Users\LENOVO\Downloads\Modul 5&6> go run .\guided1\guided1.go
Masukkan nilai a: -5
Masukkan nilai b: 7
-5 -4 -3 -2 -1 0 1 2 3 4 5 6 7
PS C:\Users\LENOVO\Downloads\Modul 5&6>

Deskripsi program

Program ini berfungsi untuk menampilkan deretan bilangan dari nilai awal hingga nilai akhir sesuai dengan input pengguna. Di awal, program meminta dua buah nilai bertipe bilangan bulat, yaitu a sebagai nilai awal dan b sebagai nilai akhir. Kedua nilai ini dibaca menggunakan perintah `fmt.Scan`.

Setelah itu, program menggunakan perulangan `for` untuk mencetak semua bilangan yang dimulai dari a sampai b. Setiap kali perulangan berjalan, nilai i akan bertambah satu, dan program akan menampilkan angka tersebut menggunakan `fmt.Print`. Hasil akhirnya berupa deretan angka yang ditampilkan secara berurutan dengan spasi di antaranya, sesuai dengan rentang nilai yang dimasukkan oleh pengguna.

2. Guided 2

Source Code

```
package main

import "fmt"

func main() {

    var j, alas, tinggi, n int

    var luas float64

    fmt.Print("Masukkan jumlah data: ")

    fmt.Scan(&n)

    for j = 1; j <=n; j+=1 {

        fmt.Print("Masukkan alas dan tinggi: ")

        fmt.Scan(&alas, &tinggi)

        luas = 0.5 * float64(alas * tinggi)

        fmt.Println("Luas segitiga: ", luas)

    }

}
```

Screenshoot program

```
guided2 > go guided2.go > main
1 package main
2 import "fmt"
3
4 func main() {
5     var j, alas, tinggi, n int
6     var luas float64
7     fmt.Print("Masukkan jumlah data: ")
8     fmt.Scan(&n)
9     for j = 1; j <=n; j+=1 {
10         fmt.Print("Masukkan alas dan tinggi: ")
11         fmt.Scan(&alas, &tinggi)
12         luas = 0.5 * float64(alas * tinggi)
13         fmt.Println("Luas segitiga: ", luas)
14     }
15 }
```

NIM 109082500178
KELAS S1IF-13-07
NAMA JEREMY CHRISTIAN SIHOMBING

PS C:\Users\LENOVO\Downloads\Modul 5&6> go run .\guided2\guided2.go
Masukkan jumlah data: 5
Masukkan alas dan tinggi: 11 2
Luas segitiga: 11
Masukkan alas dan tinggi: 32 14
Luas segitiga: 224
Masukkan alas dan tinggi: 6 2
Luas segitiga: 6
Masukkan alas dan tinggi: 15 15
Luas segitiga: 112.5
Masukkan alas dan tinggi: 20 35
Luas segitiga: 350
PS C:\Users\LENOVO\Downloads\Modul 5&6>

Deskripsi program

Program ini digunakan untuk menghitung luas beberapa segitiga berdasarkan jumlah data yang dimasukkan oleh pengguna.

Pertama, program mendeklarasikan beberapa variabel: `j`, `alas`, `tinggi`, dan `n` bertipe integer, serta `luas` bertipe `float64` agar hasil perhitungannya bisa menampilkan nilai desimal. Program kemudian meminta pengguna memasukkan jumlah data segitiga yang ingin dihitung melalui `fmt.Scan(&n)`.

Selanjutnya, program menjalankan perulangan `for` sebanyak `n` kali. Pada setiap putaran, pengguna diminta memasukkan nilai alas dan tinggi segitiga. Nilai tersebut kemudian dikonversi menjadi tipe `float64` agar bisa dikalikan dengan `0.5` untuk menghitung luas segitiga menggunakan rumus `luas = 0.5 * alas * tinggi`.

Setelah perhitungan selesai, hasil luas segitiga langsung ditampilkan di layar dengan perintah `fmt.Println`.

3. Guided 3

Source Code

```
package main

import "fmt"

func main() {

    var a, b int

    var hasil = 0

    fmt.Print("Input A: ")

    fmt.Scan(&a)

    fmt.Print("Input B: ")

    fmt.Scan(&b)

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

        hasil += b

    }

    fmt.Println("Hasil:", hasil)

}
```

Screenshoot program

```
EXPLORER    ...    guided1.go    guided2.go    tugas1.go 1    guided3.go X
MODUL 5&6
└─ guided1
   └─ guided1.go
└─ guided2
   └─ guided2.go
└─ guided3
   └─ guided3.go
└─ tugas1
   └─ tugas1.go 1

guided3 > ~\guided3.go > main
1  package main
2  import "fmt"
3
4  func main() {
5      var a, b int
6      var hasil = 0
7      fmt.Print("Input A: ")
8      fmt.Scan(&a)
9      fmt.Print("Input B: ")
10     fmt.Scan(&b)
11     for i := 1; i <= a; i++ {
12         hasil += b
13     }
14     fmt.Println("Hasil:", hasil)
15 }
16

NIM 10908
File Edit View A 100% Wind UTF-8
NIM 109082500178
KELAS SIIF-13-07
NAMA JEREMY CHRISTIAN SIHOMBING
Ln 3, Col 32 65 character Plain t 100% Wind UTF-8

PROBLEMS 1 OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\Users\LENOVO\Downloads\Modul 5&6> go run .\guided3\guided3.go
Input A: 2
Input B: 100
Hasil: 200
PS C:\Users\LENOVO\Downloads\Modul 5&6> go run .\guided3\guided3.go
Input A: 7
Input B: 6
Hasil: 42
PS C:\Users\LENOVO\Downloads\Modul 5&6>
```

Deskripsi program

Program ini digunakan untuk menghitung hasil perkalian dua bilangan bulat tanpa menggunakan operator perkalian (*), melainkan melalui proses penjumlahan berulang.

Pertama, program mendeklarasikan dua variabel bertipe integer yaitu a dan b sebagai input dari pengguna, serta variabel hasil yang diberi nilai awal 0 untuk menampung hasil perhitungan. Selanjutnya, program meminta pengguna memasukkan nilai untuk a dan b menggunakan `fmt.Scan(&a)` dan `fmt.Scan(&b)`.

Setelah nilai dimasukkan, program menjalankan perulangan `for` dari 1 hingga a. Di dalam setiap iterasi, nilai b akan ditambahkan ke variabel hasil. Artinya, penjumlahan b dilakukan sebanyak a kali, sehingga secara matematis sama dengan operasi $a * b$.

Terakhir, program menampilkan hasil perhitungannya melalui perintah `fmt.Println("Hasil:", hasil)`.

TUGAS

1. Tugas 1

Source code

```
package main

import "fmt"

func main() {

    var n, hasil int

    fmt.Print("Masukkan bilangan: ")

    fmt.Scan(&n)

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

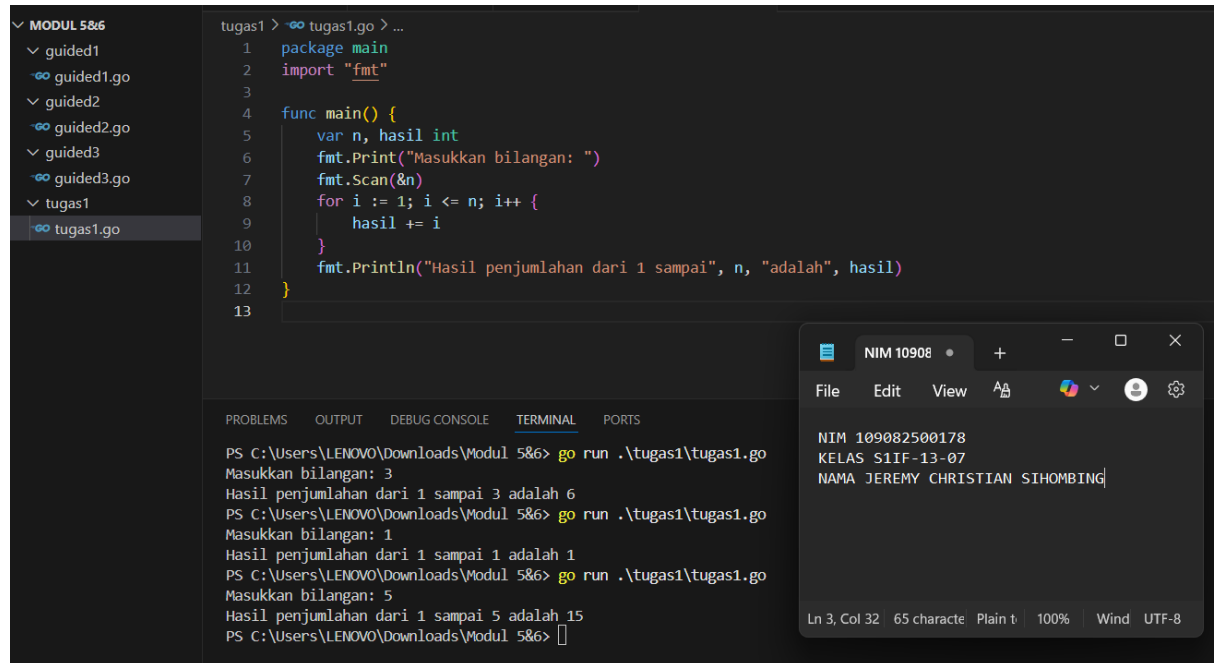
        hasil += i

    }

    fmt.Println("Hasil penjumlahan dari 1 sampai", n,
"adalah", hasil)

}
```


Screenshoot program



The screenshot displays a Go IDE interface. On the left, a file explorer shows a project structure with files named `guided1`, `guided1.go`, `guided2`, `guided2.go`, `guided3`, `guided3.go`, `tugas1`, and `tugas1.go`. The main editor window shows the source code for `tugas1.go`:

```
1 package main
2 import "fmt"
3
4 func main() {
5     var n, hasil int
6     fmt.Print("Masukkan bilangan: ")
7     fmt.Scan(&n)
8     for i := 1; i <= n; i++ {
9         hasil += i
10    }
11    fmt.Println("Hasil penjumlahan dari 1 sampai", n, "adalah", hasil)
12 }
13
```

Below the editor is a terminal window showing the execution of the program. It contains the following output:

```
PS C:\Users\LENOVO\Downloads\Modul 5&6> go run .\tugas1\tugas1.go
Masukkan bilangan: 3
Hasil penjumlahan dari 1 sampai 3 adalah 6
PS C:\Users\LENOVO\Downloads\Modul 5&6> go run .\tugas1\tugas1.go
Masukkan bilangan: 1
Hasil penjumlahan dari 1 sampai 1 adalah 1
PS C:\Users\LENOVO\Downloads\Modul 5&6> go run .\tugas1\tugas1.go
Masukkan bilangan: 5
Hasil penjumlahan dari 1 sampai 5 adalah 15
PS C:\Users\LENOVO\Downloads\Modul 5&6>
```

On the right side of the terminal, there is a small window titled "NIM 10908" containing the following text:

```
NIM 109082500178
KELAS S1IF-13-07
NAMA JEREMY CHRISTIAN SIHOMBING
```

Deskripsi program

Program ini digunakan untuk menghitung hasil penjumlahan dari angka 1 hingga angka n yang dimasukkan oleh pengguna. Pertama, program mendeklarasikan dua variabel yaitu n untuk menampung input bilangan dan $hasil$ untuk menyimpan total penjumlahan. Kemudian program meminta pengguna memasukkan sebuah bilangan bulat positif menggunakan `fmt.Scan(&n)`. Setelah itu, dilakukan proses perulangan `for` dari 1 sampai n . Di setiap iterasi, nilai i akan ditambahkan ke dalam variabel $hasil$, sehingga program secara bertahap menjumlahkan seluruh bilangan dari 1 hingga n . Terakhir, hasil penjumlahan tersebut ditampilkan menggunakan `fmt.Println`, sehingga pengguna dapat melihat total akhir dari penjumlahan angka-angka tersebut.

2. Tugas 2

Source code

```
package main

import (

    "fmt"

    "math"

)

func main() {

    var n int

    var r, t float64

    fmt.Print("Masukkan jumlah kerucut: ")

    fmt.Scan(&n)

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

        fmt.Print("Masukkan jari-jari dan tinggi: ")

        fmt.Scan(&r, &t)

        volume := (1.0 / 3.0) * math.Pi * math.Pow(r,

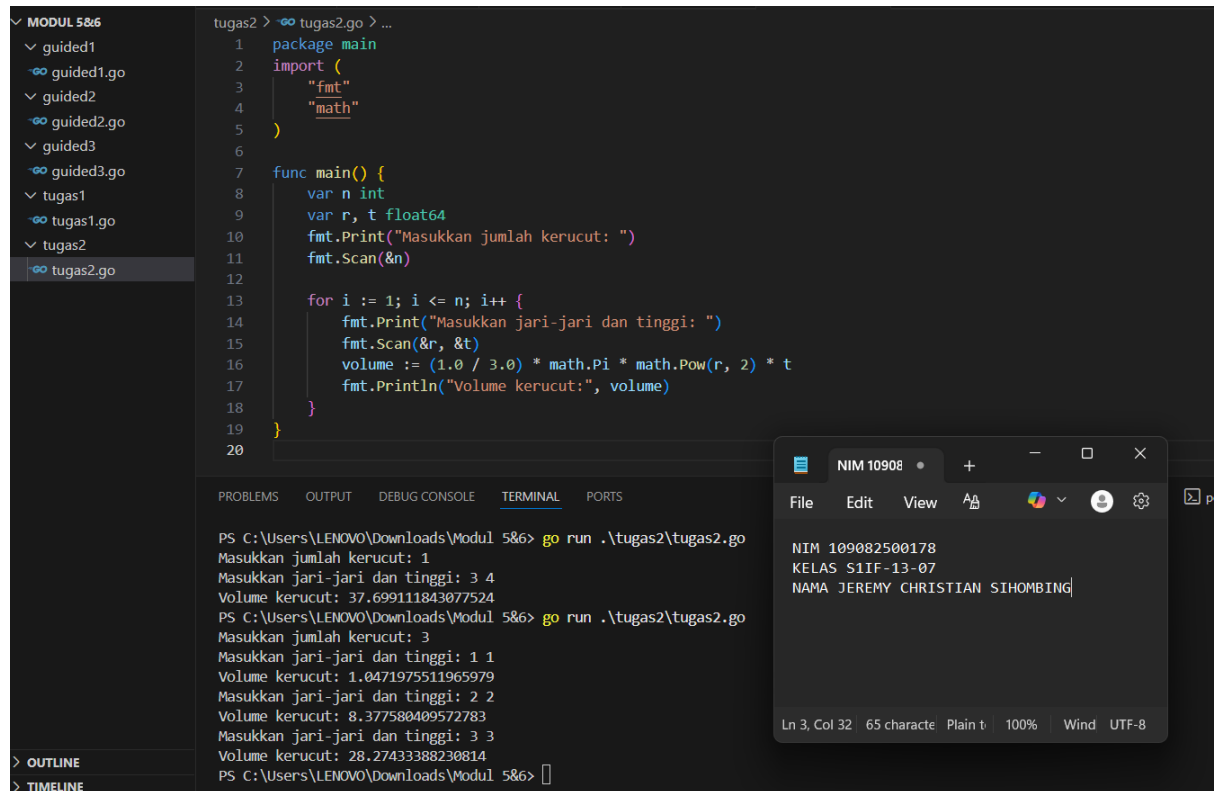
2) * t

        fmt.Println("Volume kerucut:", volume)

    }

}
```

Screenshoot program



```
1 package main
2 import (
3     "fmt"
4     "math"
5 )
6
7 func main() {
8     var n int
9     var r, t float64
10    fmt.Print("Masukkan jumlah kerucut: ")
11    fmt.Scan(&n)
12
13    for i := 1; i <= n; i++ {
14        fmt.Print("Masukkan jari-jari dan tinggi: ")
15        fmt.Scan(&r, &t)
16        volume := (1.0 / 3.0) * math.Pi * math.Pow(r, 2) * t
17        fmt.Println("Volume kerucut:", volume)
18    }
19 }
20
```

PS C:\Users\LENOVO\Downloads\Modul 5&6> go run .\tugas2\tugas2.go
Masukkan jumlah kerucut: 1
Masukkan jari-jari dan tinggi: 3 4
Volume kerucut: 37.699111843077524
PS C:\Users\LENOVO\Downloads\Modul 5&6> go run .\tugas2\tugas2.go
Masukkan jumlah kerucut: 3
Masukkan jari-jari dan tinggi: 1 1
Volume kerucut: 1.0471975511965979
Masukkan jari-jari dan tinggi: 2 2
Volume kerucut: 8.377580409572783
Masukkan jari-jari dan tinggi: 3 3
Volume kerucut: 28.27433388230814
PS C:\Users\LENOVO\Downloads\Modul 5&6>

Deskripsi program

Program ini digunakan untuk menghitung volume dari beberapa buah kerucut berdasarkan nilai jari-jari alas dan tinggi yang dimasukkan oleh pengguna. Langkah pertama, program meminta input berupa jumlah kerucut (n) yang akan dihitung. Setelah itu, program menjalankan perulangan sebanyak n kali untuk menerima data setiap kerucut, yaitu jari-jari alas (r) dan tinggi (t). Rumus yang digunakan untuk menghitung volume kerucut adalah $V = \frac{1}{3} \times \pi \times r^2 \times t$, di mana π diambil dari konstanta `math.Pi` dan pemangkatan r dilakukan dengan fungsi `math.Pow(r, 2)` agar hasilnya lebih presisi. Setelah volume dihitung, hasilnya langsung ditampilkan ke layar dalam satuan unit kubik. Dengan cara ini, program mampu menghitung beberapa volume kerucut sekaligus berdasarkan banyaknya data yang diberikan pengguna.

3. Tugas 3

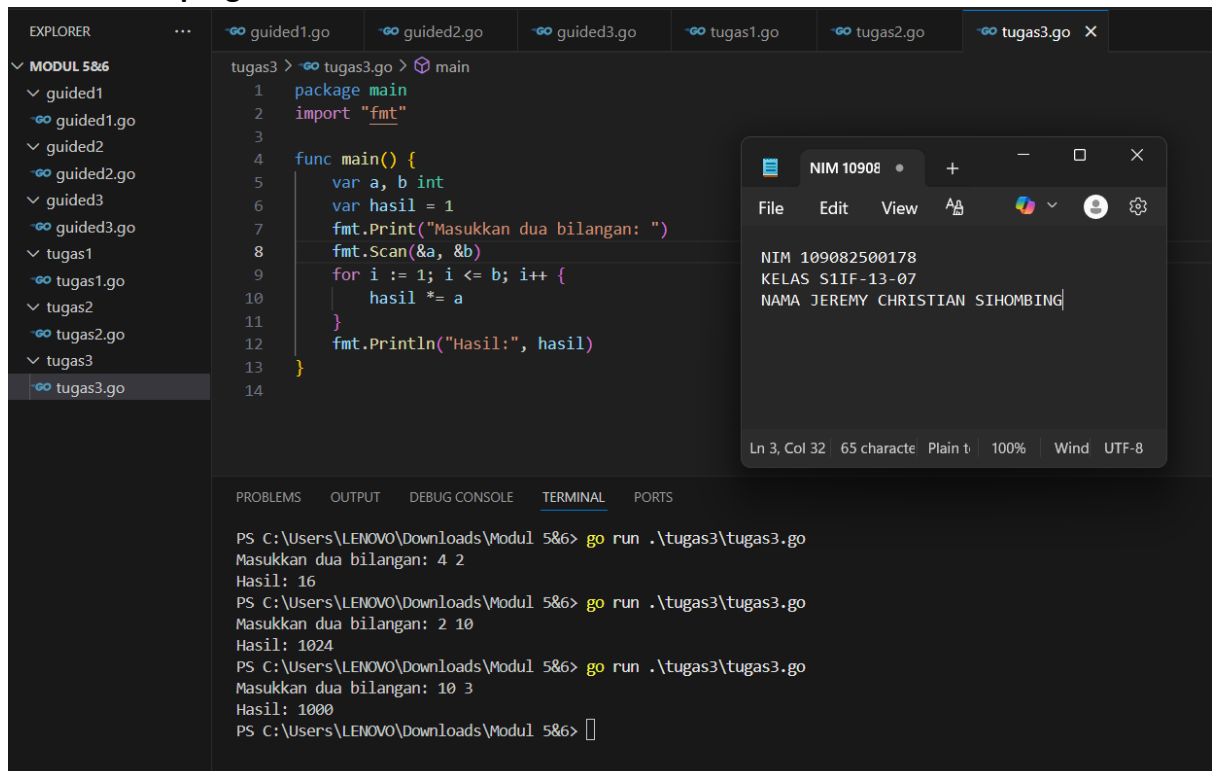
Source code

```
package main

import "fmt"

func main() {
    var a, b int
    var hasil = 1
    fmt.Print("Masukkan dua bilangan: ")
    fmt.Scan(&a, &b)
    for i := 1; i <= b; i++ {
        hasil *= a
    }
    fmt.Println("Hasil:", hasil)
}
```

Screenshoot program



Deskripsi program

Program ini berfungsi untuk menghitung hasil pemangkatan dari dua bilangan bulat positif menggunakan operator perkalian dan perulangan. Pada awal program, pengguna diminta memasukkan dua bilangan, yaitu bilangan dasar (a) dan bilangan pangkat (b). Variabel hasil diinisialisasi dengan nilai 1 sebagai nilai awal perkalian.

Selanjutnya, program menggunakan perulangan for sebanyak b kali. Setiap iterasi, nilai hasil dikalikan dengan a, sehingga pada akhir perulangan diperoleh nilai a pangkat b. Setelah proses perulangan selesai, hasil akhir ditampilkan di layar.

4. Tugas 4

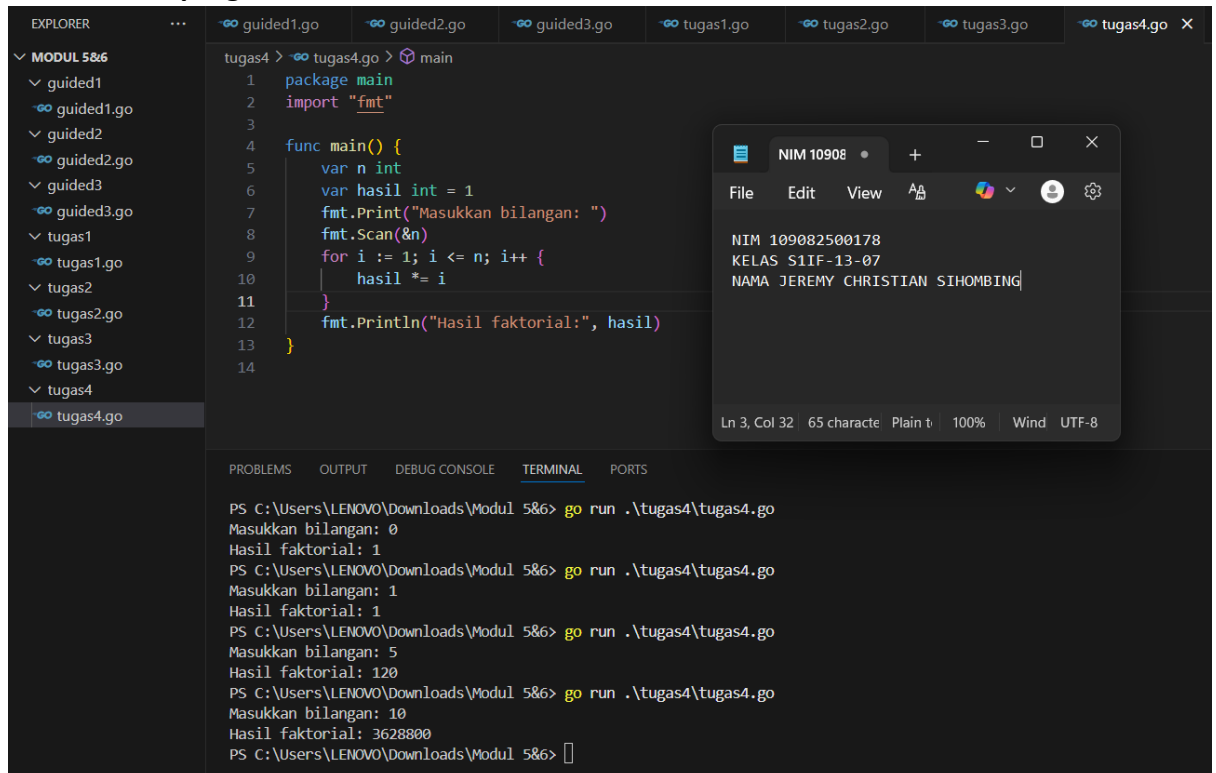
Source code

```
package main

import "fmt"

func main() {
    var n int
    var hasil int = 1
    fmt.Print("Masukkan bilangan: ")
    fmt.Scan(&n)
    for i := 1; i <= n; i++ {
        hasil *= i
    }
    fmt.Println("Hasil faktorial:", hasil)
}
```

Screenshoot program



The screenshot displays a Go IDE interface. On the left, the Explorer pane shows a project structure with files named 'guided1.go' through 'tugas4.go'. The main editor window shows the source code for 'tugas4.go', which is identical to the code provided in the 'Source code' section. A floating window titled 'NIM 10908' is open, containing the text: 'NIM 109082500178', 'KELAS S1IF-13-07', and 'NAMA JEREMY CHRISTIAN SIHOMBING'. At the bottom, the Terminal pane shows the execution of the program. It displays the prompt 'PS C:\Users\LENOVO\Downloads\Modul 5&6>' followed by the command 'go run .\tugas4\tugas4.go'. The output shows the program prompting for a number and calculating its factorial for inputs 0, 1, 5, and 10, resulting in 1, 1, 120, and 3628800 respectively.

```
package main
import "fmt"

func main() {
    var n int
    var hasil int = 1
    fmt.Print("Masukkan bilangan: ")
    fmt.Scan(&n)
    for i := 1; i <= n; i++ {
        hasil *= i
    }
    fmt.Println("Hasil faktorial:", hasil)
}
```

PS C:\Users\LENOVO\Downloads\Modul 5&6> go run .\tugas4\tugas4.go
Masukkan bilangan: 0
Hasil faktorial: 1
PS C:\Users\LENOVO\Downloads\Modul 5&6> go run .\tugas4\tugas4.go
Masukkan bilangan: 1
Hasil faktorial: 1
PS C:\Users\LENOVO\Downloads\Modul 5&6> go run .\tugas4\tugas4.go
Masukkan bilangan: 5
Hasil faktorial: 120
PS C:\Users\LENOVO\Downloads\Modul 5&6> go run .\tugas4\tugas4.go
Masukkan bilangan: 10
Hasil faktorial: 3628800
PS C:\Users\LENOVO\Downloads\Modul 5&6>

Deskripsi program

Program ini digunakan untuk menghitung hasil faktorial dari suatu bilangan bulat non-negatif. Faktorial dari bilangan n (ditulis $n!$) adalah hasil perkalian semua bilangan bulat positif dari 1 hingga n .

Pertama, program meminta pengguna untuk memasukkan sebuah bilangan bulat n . Variabel hasil diinisialisasi dengan nilai 1 karena nilai faktorial dimulai dari 1. Kemudian, program menggunakan perulangan for yang berjalan dari 1 hingga n . Pada setiap iterasi, nilai hasil dikalikan dengan nilai i , sehingga setelah perulangan selesai, hasil berisi nilai faktorial dari n . Sebagai contoh, jika pengguna memasukkan 5, maka program akan menghitung $1 \times 2 \times 3 \times 4 \times 5 = 120$, dan hasilnya ditampilkan di layar.