

**LAPORAN PRAKTIKUM ALGORITMA
DAN PEMROGRAMAN 1**

**MODUL 05-06
FOR-LOOP**



Disusun oleh:

FARID HERDIYANTO VITASANDI

109082500123

S1IF-13-02

Asisten Praktikum

Adithana dharma putra

Alfin Ilham Berlianto

PROGRAM STUDI S1 INFORMATIKA

FAKULTAS INFORMATIKA

TELKOM UNIVERSITY PURWOKERTO

2025

LATIHAN KELAS – GUIDED

1. Guided 1 Source Code

```
package main

import "fmt"

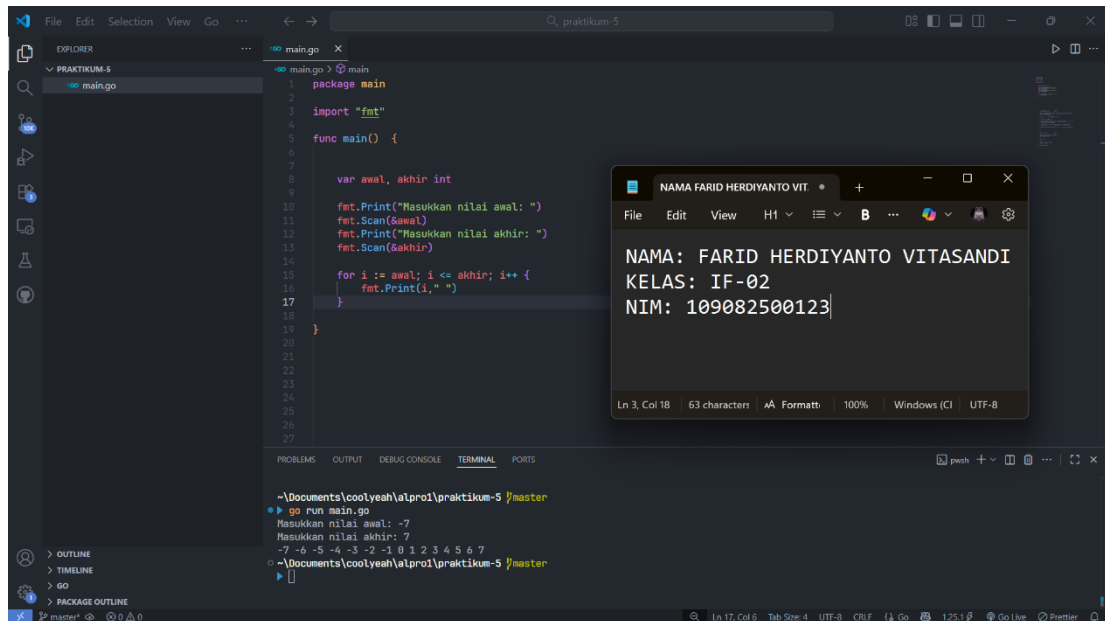
func main() {
    var awal, akhir int

    fmt.Print("Masukkan nilai awal: ")
    fmt.Scan(&awal)

    fmt.Print("Masukkan nilai akhir: ")
    fmt.Scan(&akhir)

    for i := awal; i <= akhir; i++ {
        fmt.Print(i, " ")
    }
}
```

Screenshoot program



The screenshot shows a Go program in a text editor and its execution output in a terminal window. The program is located in `main.go` and uses the `fmt` package to prompt the user for two integers, `awal` and `akhir`. It then uses a `for` loop to print a sequence of numbers from `awal` to `akhir`.

```
1 package main
2
3 import "fmt"
4
5 func main() {
6
7
8     var awal, akhir int
9
10    fmt.Print("Masukkan nilai awal: ")
11    fmt.Scan(&awal)
12    fmt.Print("Masukkan nilai akhir: ")
13    fmt.Scan(&akhir)
14
15    for i := awal; i <= akhir; i++ {
16        fmt.Print(i, " ")
17    }
18
19 }
20
21
22
23
24
25
26
27
```

The terminal window shows the output of the program, displaying the sequence of numbers from -7 to 7, separated by spaces.

```
~\Documents\coolyeah\alpro1\praktikum-5 /master
go run main.go
Masukkan nilai awal: -7
Masukkan nilai akhir: 7
-7 -6 -5 -4 -3 -2 -1 0 1 2 3 4 5 6 7
```

Deskripsi program

Program di atas berfungsi untuk menampilkan deretan angka dari nilai awal hingga akhir yang diinput oleh User. Program meminta User untuk memasukkan dua angka, yaitu nilai awal dan akhir. Selanjutnya, perulangan `for` dengan variable `i` yang dimulai dari nilai awal akan terus bertambah hingga mencapai nilai akhir seperti yang terlihat pada hasil tangkapan layar di atas.

2. Guided 2

Source Code

```
package main

import "fmt"

func main() {

    var segitiga int

    fmt.Print("Masukkan jumlah segitiga yang akan dihitung: ")

    fmt.Scan(&segitiga)

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

        var alas, tinggi int

        fmt.Print("Masukkan Alas dan Tinggi Segitiga(",i,"): ")

        fmt.Scan(&alas, &tinggi)

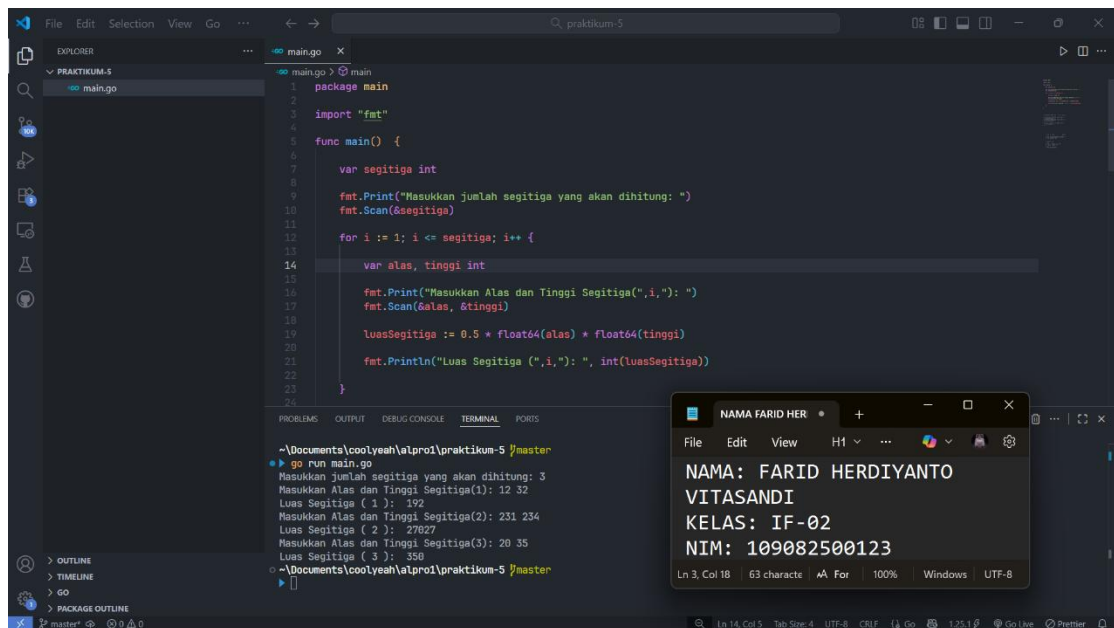
        luasSegitiga := 0.5 * float64(alas) *
float64(tinggi)

        fmt.Println("Luas Segitiga (",i,"): ",
int(luasSegitiga))

    }

}
```

Screenshoot program



The screenshot shows a Go program in a VS Code editor and its execution output in a terminal window. The program is located in `main.go` under the `praktikum-5` directory. The code defines a `main` function that prompts the user for the number of triangles, then iterates to collect base and height values, calculate the area for each triangle, and print the results.

```
1 package main
2
3 import "fmt"
4
5 func main() {
6
7     var segitiga int
8
9     fmt.Print("Masukkan jumlah segitiga yang akan dihitung: ")
10    fmt.Scan(&segitiga)
11
12    for i := 1; i <= segitiga; i++ {
13
14        var alas, tinggi int
15
16        fmt.Print("Masukkan Alas dan Tinggi Segitiga(", i, "): ")
17        fmt.Scan(&alas, &tinggi)
18
19        luasSegitiga := 0.5 * float64(alas) * float64(tinggi)
20        fmt.Println("Luas Segitiga (", i, "): ", int(luasSegitiga))
21    }
22 }
23
24
```

The terminal window shows the following output:

```
~\Documents\cool\yeah\alpro1\praktikum-5 $go run main.go
Masukkan jumlah segitiga yang akan dihitung: 3
Masukkan Alas dan Tinggi Segitiga(1): 12 32
Luas Segitiga ( 1 ): 192
Masukkan Alas dan Tinggi Segitiga(2): 231 234
Luas Segitiga ( 2 ): 27027
Masukkan Alas dan Tinggi Segitiga(3): 20 35
Luas Segitiga ( 3 ): 350
```

Deskripsi program

Program di atas dibuat untuk menghitung luas beberapa segitiga berdasarkan input User. Program akan meminta banyak segitiga yang akan dihitung, lalu disimpan di variable *segitiga*. Setelah itu program akan menjalankan perulangan for sebanyak jumlah segitiga yang di-input oleh User. Di setiap perulangan, User akan diminta memasukkan nilai alas dan tinggi segitiga, nilai tersebut akan kemudian akan dikonversi ke tipe float64 agar bisa dihitung dalam rumus $0.5 * \text{alas} * \text{tinggi}$. Hasil perhitungan selanjutnya akan disimpan dalam variable *luasSegitiga* dan ditampilkan di layar sebagai hasil dari luas segitiga ke-I seperti yang terlihat pada hasil screenshoot di atas.

3. Guided 3

Source Code

```
package main

import "fmt"

func main() {
    var a, b int

    fmt.Print("Masukkan bilangan: ")

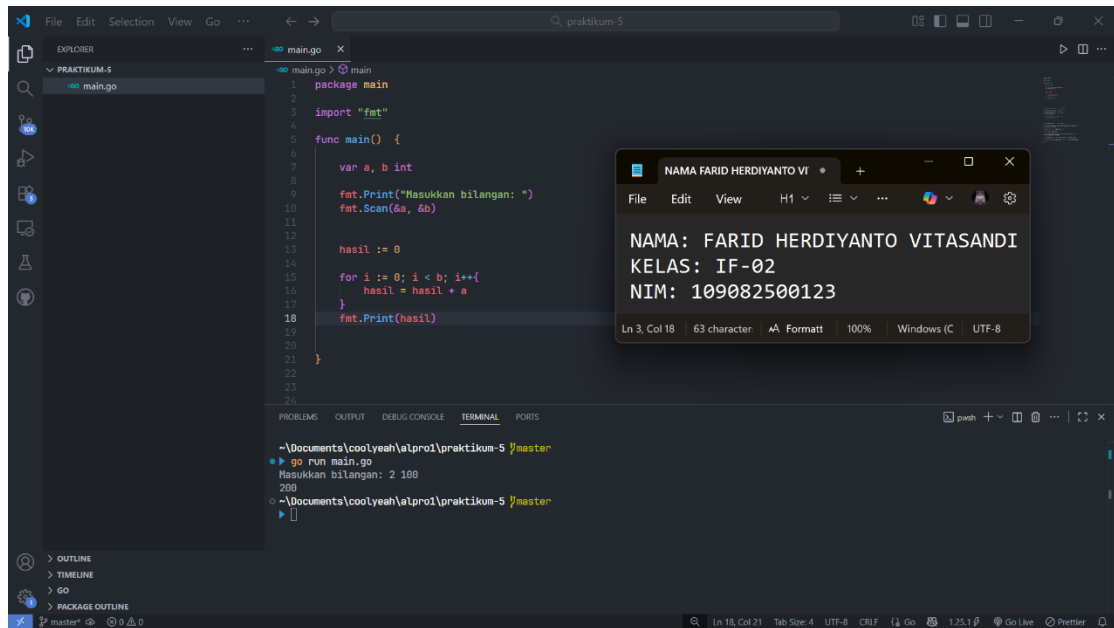
    fmt.Scan(&a, &b)

    hasil := 0

    for i := 0; i < b; i++){
        hasil = hasil + a
    }

    fmt.Print(hasil)
}
```

Screenshoot program



```
1 package main
2
3 import "fmt"
4
5 func main() {
6
7     var a, b int
8
9     fmt.Print("Masukkan bilangan: ")
10    fmt.Scan(&a, &b)
11
12    hasil := 0
13
14    for i := 0; i < b; i++{
15        hasil = hasil + a
16    }
17    fmt.Print(hasil)
18 }
```

Terminal Output:

```
~\Documents\coolyeah\alpro1\praktikum-5 $master
go run main.go
Masukkan bilangan: 2 100
200
~\Documents\coolyeah\alpro1\praktikum-5 $master
```

Deskripsi program

Kode di atas adalah program yang dibuat untuk menghitung hasil perkalian tanpa menggunakan operator perkalian (*). Program akan meminta User untuk memasukkan dua bilangan bulat, yaitu a dan b. Variabel hasil diinisialisasi dengan nilai 0 untuk menyimpan nilai akhir. Lalu, Program akan menggunakan perulangan for sebanyak b kali, dan di setiap perulangan, nilai a ditambahkan ke variable hasil. Dengan cara ini, a dijumlahkan sebanyak b kali yang secara logika hasilnya akan sama dengan perkalian $a * b$.

TUGAS

1. Soal 1

Source Code

```
package main

import "fmt"

func main() {

    var n, hasil int

    fmt.Print("Masukkan n: ")

    fmt.Scan(&n)

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

        var bilangan int

        fmt.Printf("Masukkan bilangan ke-%d: ", i)

        fmt.Scan(&bilangan)

        for j :=1; j <= bilangan; j++){

            hasil = hasil + 1

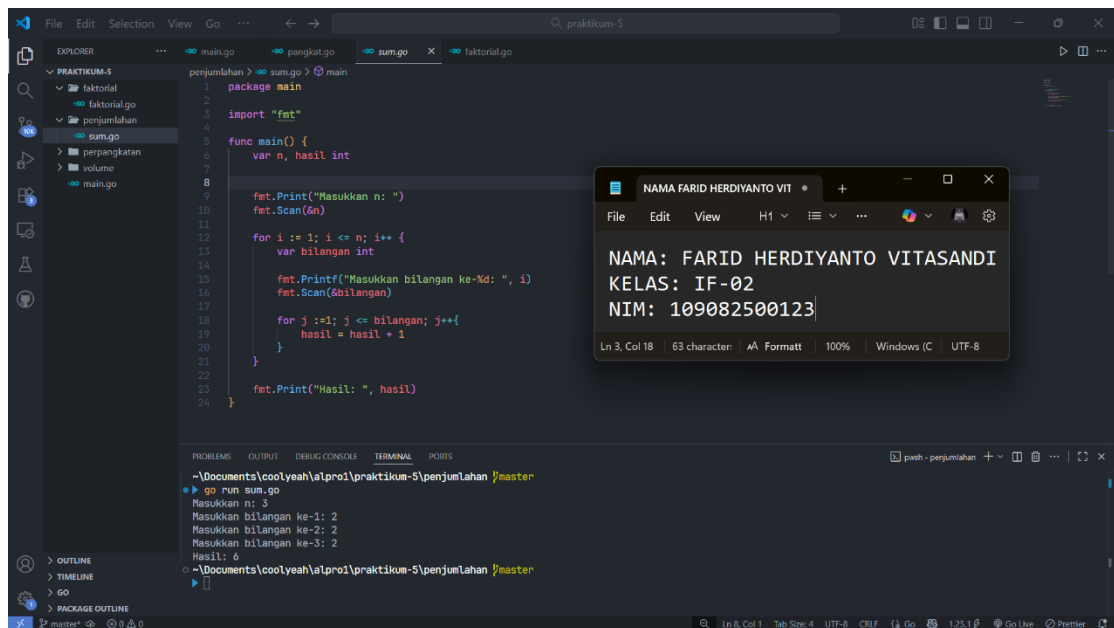
        }

    }

    fmt.Print("Hasil: ", hasil)

}
```


Screenshoot program



The screenshot shows a Go program in a VS Code editor. The program is located in `sum.go` within the `penjumlahan` package. The code uses `fmt` for input/output and a nested loop to calculate the sum of numbers from 1 to `n`. The terminal shows the program being run, with input values 3, 2, 2, 2 and the output 6.

```
1 package main
2
3 import "fmt"
4
5 func main() {
6     var n, hasil int
7
8     fmt.Print("Masukkan n: ")
9     fmt.Scan(&n)
10
11     for i := 1; i <= n; i++ {
12         var bilangan int
13
14         fmt.Printf("Masukkan bilangan ke-%d: ", i)
15         fmt.Scan(&bilangan)
16
17         for j := 1; j <= bilangan; j++ {
18             hasil = hasil + 1
19         }
20     }
21
22     fmt.Print("Hasil: ", hasil)
23 }
24
```

Terminal Output:

```
-Documents\cool\yeah\alpro\praktikum-5\penjumlahan /master
go run sum.go
Masukkan n: 3
Masukkan bilangan ke-1: 2
Masukkan bilangan ke-2: 2
Masukkan bilangan ke-3: 2
Hasil: 6
```

Deskripsi program

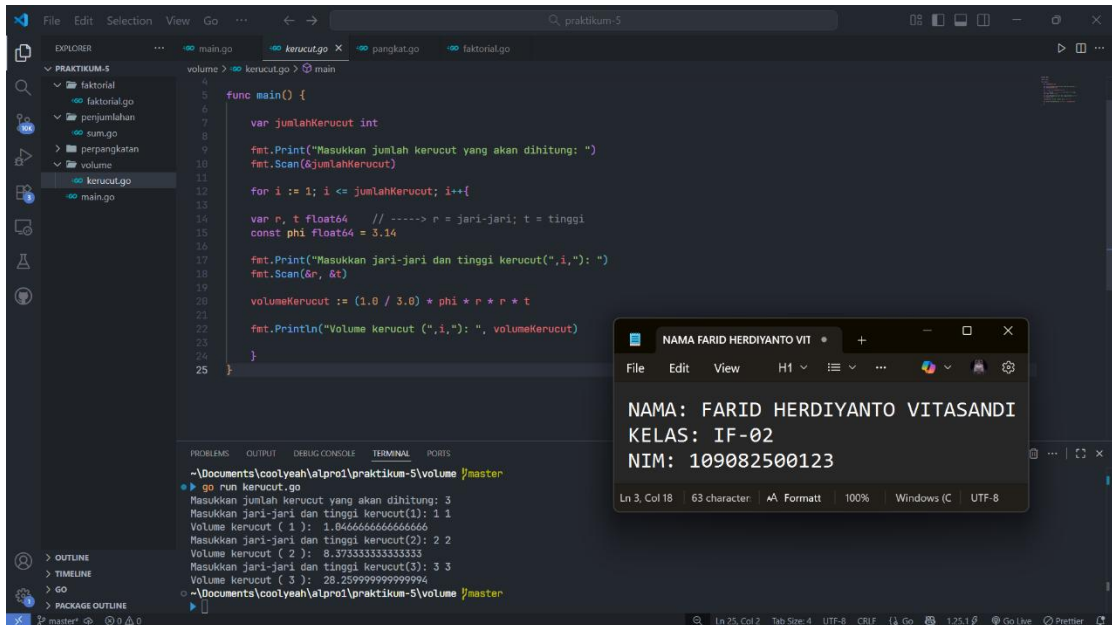
Program di atas adalah program yang dibuat untuk menghitung hasil dari penjumlahan beberapa bilangan tanpa menggunakan operator tambah (+) secara langsung. Program akan meminta User untuk memasukkan nilai `n` (jumlah bilangan yang akan diproses). Setelah itu program akan menjalankan perulangan `for` dari `i = 1` hingga `n`, dimana pada setiap perulangan pengguna akan diminta untuk memasukkan satu bilangan. Untuk setiap bilangan yang dimasukkan, program akan menjalankan perulangan dari `j = 1` hingga `bilangan`, dan di setiap perulangan nilai `hasil` akan ditambah 1. Setelah semua bilangan selesai diproses, program akan menampilkan total nilai yang disimpan dalam variable `hasil`.

2. Soal 2

Source Code

```
func main() {  
  
    var jumlahKerucut int  
  
    fmt.Print("Masukkan jumlah kerucut yang akan dihitung:  
")  
  
    fmt.Scan(&jumlahKerucut)  
  
    for i := 1; i <= jumlahKerucut; i++){  
  
        var r, t float64  
  
        const phi float64 = 3.14  
  
        fmt.Print("Masukkan jari-jari dan tinggi  
kerucut(", i, "): ")  
  
        fmt.Scan(&r, &t)  
  
        volumeKerucut := (1.0 / 3.0) * phi * r * r * t  
  
        fmt.Println("Volume kerucut (", i, "): ", volumeKerucut)  
  
    }  
}
```

Screenshoot program



```
func main() {  
    var jumlahKerucut int  
    fmt.Print("Masukkan jumlah kerucut yang akan dihitung: ")  
    fmt.Scan(&jumlahKerucut)  
    for i := 1; i <= jumlahKerucut; i++{  
        var r, t float64 // ----> r = jari-jari; t = tinggi  
        const phi float64 = 3.14  
        fmt.Print("Masukkan jari-jari dan tinggi kerucut(", i, "): ")  
        fmt.Scan(&r, &t)  
        volumeKerucut := (1.0 / 3.0) * phi * r * r * t  
        fmt.Println("Volume kerucut (", i, "): ", volumeKerucut)  
    }  
}
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
-Documents\cool\yeah\alpro1\praktikum-5\volume /master  
➤ go run kerucut.go  
Masukkan jumlah kerucut yang akan dihitung: 3  
Masukkan jari-jari dan tinggi kerucut(1): 1 1  
Volume kerucut ( 1 ): 1.0466666666666666  
Masukkan jari-jari dan tinggi kerucut(2): 2 2  
Volume kerucut ( 2 ): 8.373333333333333  
Masukkan jari-jari dan tinggi kerucut(3): 3 3  
Volume kerucut ( 3 ): 28.259999999999994  
-Documents\cool\yeah\alpro1\praktikum-5\volume /master
```

Deskripsi program

Program di atas berfungsi untuk menghitung volume beberapa kerucut berdasarkan input User. Program akan meminta jumlah banyaknya kerucut yang akan dihitung, lalu akan menjalankan perulangan for sebanyak jumlah yang diinputkan oleh User. Di setiap perulangan, User akan diminta untuk memasukkan jari-jari (r) dan tinggi (t) kerucut. Nilai-nilai tersebut nantinya akan digunakan dalam rumus volume kerucut, yaitu $V = \frac{1}{3} \times \pi \times r^2 \times t$ dengan nilai π (phi) ditetapkan sebagai 3.14. Hasil perhitungan kemudian akan disimpan dalam *volumeKerucut* dan langsung ditampilkan seperti yang terlihat pada hasil screenshoot di atas.

3. Soal 3

Source Code

```
package main

import "fmt"

func main() {

    var x, y, hasil int

    fmt.Print("Masukkan bilangan yang akan dipangkatkan: ")

    fmt.Scan(&x)

    fmt.Print("Masukkan pangkat: ")

    fmt.Scan(&y)

    hasil = 1

    for i := 1; i <= y; i++){

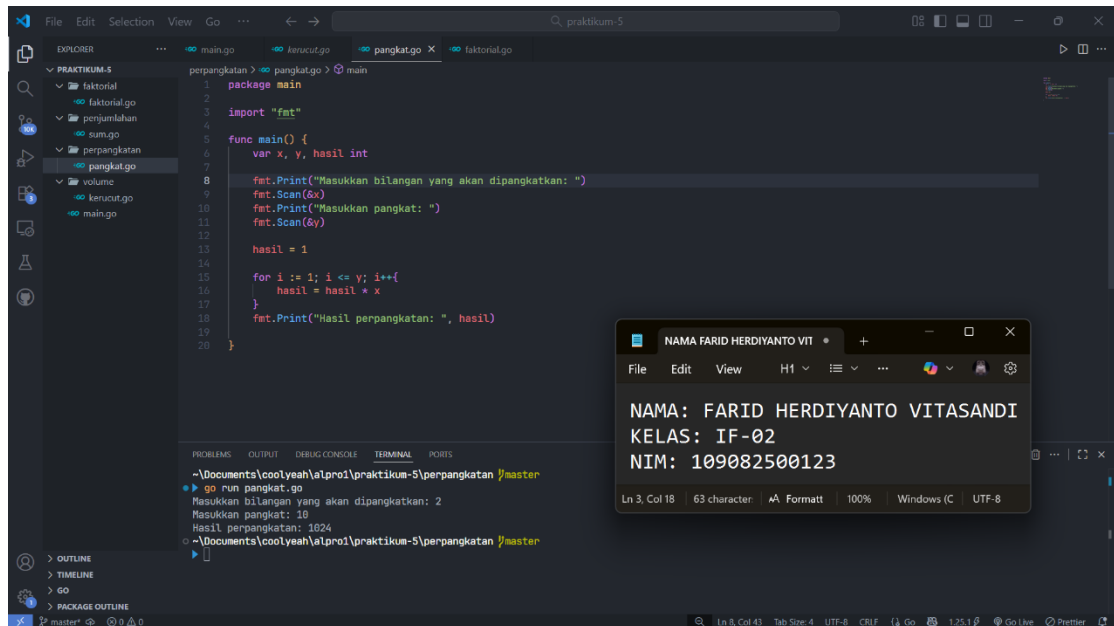
        hasil = hasil * x

    }

    fmt.Print("Hasil perpangkatan: ", hasil)

}
```

Screenshoot program



The screenshot shows a Go program in a VS Code editor. The program is located in a file named `perpangkatan.go` within a project structure. The code defines a `main` function that prompts the user for a base number (`x`) and an exponent (`y`), then calculates the power of `x` to the `y`th power. The result is printed to the console.

```
1 package main
2
3 import "fmt"
4
5 func main() {
6     var x, y, hasil int
7
8     fmt.Println("Masukkan bilangan yang akan dipangkatkan: ")
9     fmt.Scan(&x)
10    fmt.Println("Masukkan pangkat: ")
11    fmt.Scan(&y)
12
13    hasil = 1
14
15    for i := 1; i <= y; i++ {
16        hasil = hasil * x
17    }
18    fmt.Println("Hasil perpangkatan: ", hasil)
19 }
20
```

The terminal output shows the program being executed with the following input and output:

```
-Documents\cool\yeah\alpro\praktikum-5\perpangkatan /master
> go run perpangkatan.go
Masukkan bilangan yang akan dipangkatkan: 2
Masukkan pangkat: 10
Hasil perpangkatan: 1024
```

Deskripsi program

Program di atas dibuat untuk menghitung perpangkatan bilangan. User akan diminta untuk memasukkan bilangan yang akan dipangkatkan (x) dan pangkatnya (y). Variabel diinisialisasi dengan nilai 1, karena dalam perpangkatan nilai awal harus 1. Lalu, program akan menjalankan perulangan for sebanyak y kali, dan di setiap perulangan, variable `hasil` akan dikalikan dengan x . Setelah proses selesai, program akan menampilkan hasil dari x pangkat y .

4. Soal 4

Source Code

```
package main

import "fmt"

func main() {

    var n int

    fmt.Print("Masukkkan bilangan: ")

    fmt.Scan(&n)

    faktorial := 1

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

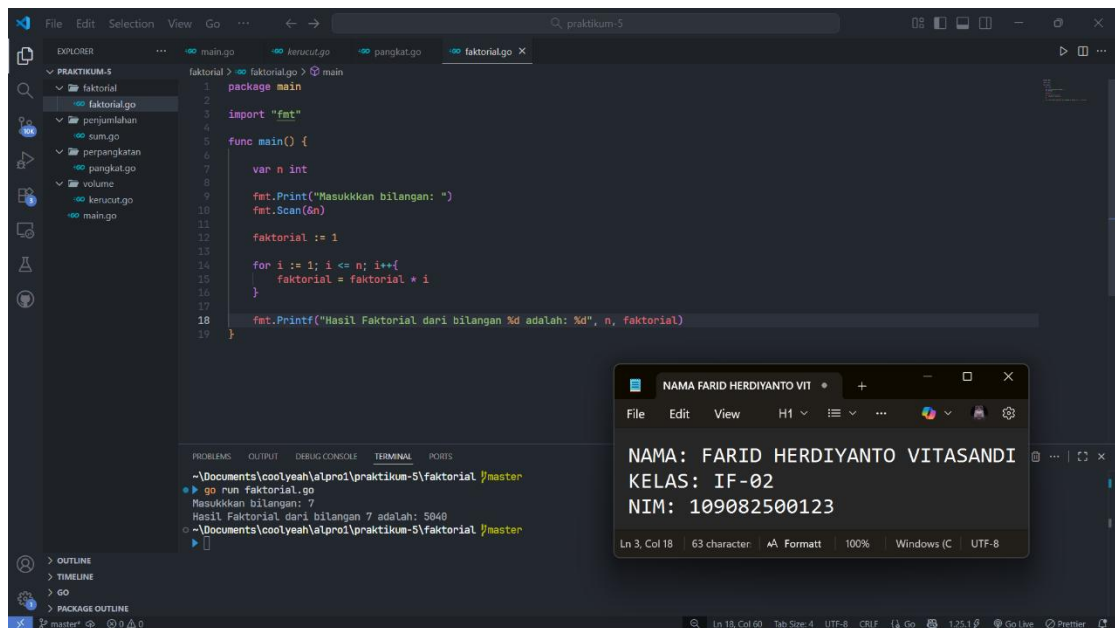
        faktorial = faktorial * i

    }

    fmt.Printf("Hasil Faktorial dari bilangan %d adalah: %d", n, faktorial)

}
```

Screenshoot program



Deskripsi program

Program di atas dibuat untuk menghitung hasil factorial dari sebuah bilangan. Program akan meminta User untuk memasukkan sebuah bilangan bulat n . Nilai awal variable *factorial* diatur menjadi 1. Kemudian, program akan menjalankan perulangan for dari 1 hingga n , dan di setiap perulangan, nilai *factorial* akan dikalikan dengan i . Setelah perulangan selesai, hasl faktorial akan ditampilkan menggunakan *fmt.Printf* seperti yang terlihat pada screenshoot di atas.