

**LAPORAN PRAKTIKUM ALGORITMA
DAN PEMROGRAMAN 1**

**MODUL V
FOR-LOOP**



Disusun oleh:

NAMA : DAYANA RISTA NUR FAUZIAH

NIM : 109082500195

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

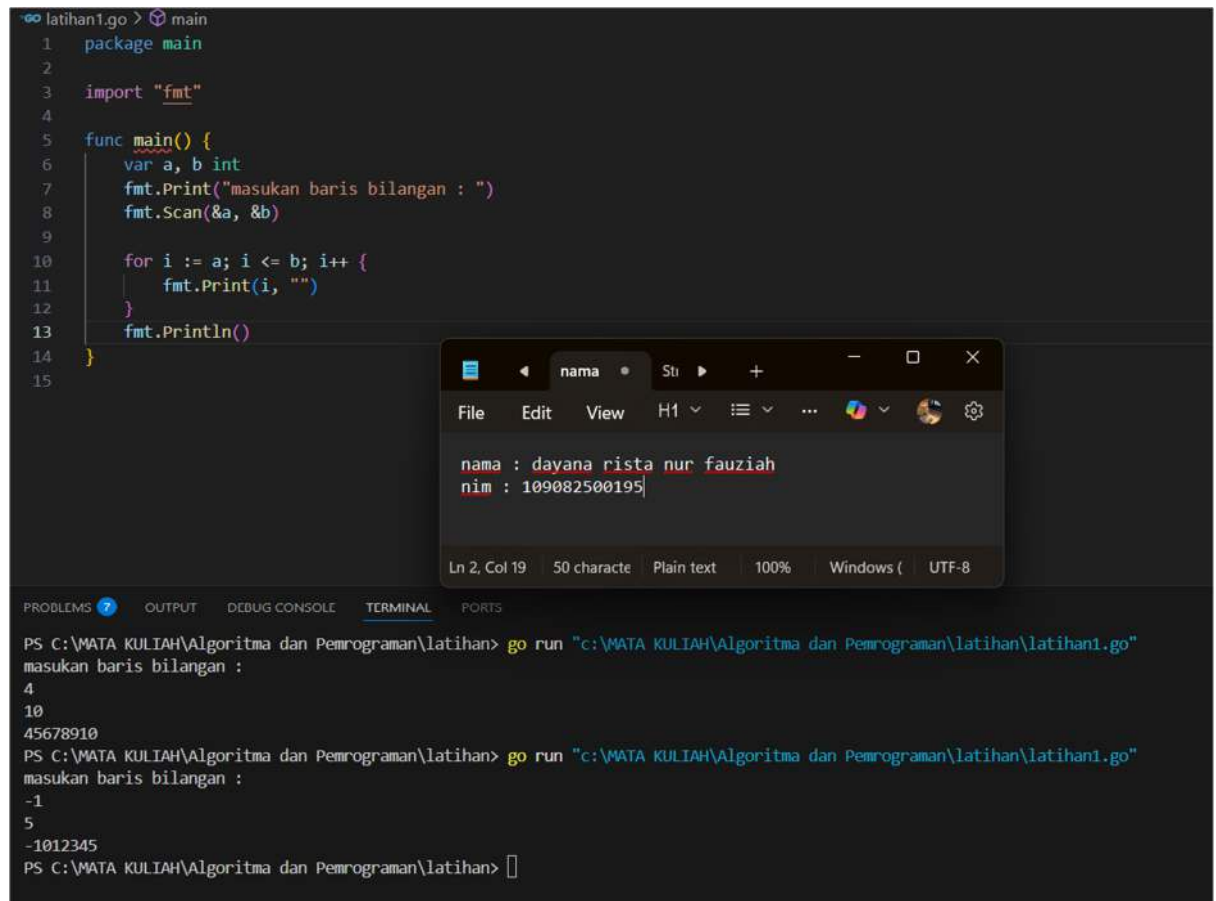
```
package main

import "fmt"

func main() {
    var a, b int
    fmt.Print("masukan baris bilangan : ")
    fmt.Scan(&a, &b)

    for i := a; i <= b; i++ {
        fmt.Print(i, " ")
    }
    fmt.Println()
}
```

Screenshoot program



```
latihan1.go > main
1 package main
2
3 import "fmt"
4
5 func main() {
6     var a, b int
7     fmt.Print("masukan baris bilangan : ")
8     fmt.Scan(&a, &b)
9
10    for i := a; i <= b; i++ {
11        fmt.Print(i, " ")
12    }
13    fmt.Println()
14 }
15
```

```
nama : dayana rista nur fauziah
nim : 109082500195
```

```
PS C:\MATA KULIAH\Algoritma dan Pemrograman\latihan> go run "c:\MATA KULIAH\Algoritma dan Pemrograman\latihan\latihan1.go"
masukan baris bilangan :
4
10
45678910
PS C:\MATA KULIAH\Algoritma dan Pemrograman\latihan> go run "c:\MATA KULIAH\Algoritma dan Pemrograman\latihan\latihan1.go"
masukan baris bilangan :
-1
5
-1012345
PS C:\MATA KULIAH\Algoritma dan Pemrograman\latihan>
```

Deskripsi program

Program meminta untuk memasukan dua angka berbentunk integer (a sebagai awal dan b sebagai akhir keluaran). Lalu menggunakan perulangan (for) untuk mencetak semua bilangan bulat secara berurutan, dimulai dari a hingga b. Loop akan terus berjalan selama nilai i kurang dari atau sama dengan nilai variabel b atau angka akhir. Proses ini berulang, mencetak a, lalu a+1, lalu a+2 dan seterusnya, sampai i melebihi b. Setelah loop selesai mencetak seluruh deret angka, perintah *fmt.Println()* mencetak satu baris baru, memastikan kursor pindah ke baris berikutnya.

2. Guided 2

Source Code

```
package main

import "fmt"

func main() {

    var j, alas, tinggi, n int
    var luas float64

    fmt.Scan(&n)
    for j = 1; j <= n; j += 1 {
        fmt.Scan(&alas, &tinggi)
        luas = 0.5 * float64(alas*tinggi)
        fmt.Println(luas)
    }
}
```

Screenshoot program

The screenshot shows a Go program in a code editor and its execution in a terminal. The program is a simple loop that calculates the area of a triangle for a given number of iterations. A small window is also open showing user input.

```
guided2.go > ...
1 package main
2
3 import "fmt"
4
5 func main() {
6
7     var j, alas, tinggi, n int
8     var luas float64
9
10    fmt.Scan(&n)
11    for j = 1; j <= n; j += 1 {
12        fmt.Scan(&alas, &tinggi)
13        luas = 0.5 * float64(alas*tinggi)
14        fmt.Println(luas)
15    }
16 }
17
```

nama : dayana rista nur fauziah
nim : 109082500195

```
PROBLEMS 9 OUTPUT DEBUG CONSOLE TERMINAL
.\guided2.go:12:15: undefined: luassegitiga
PS C:\WATA KULIAH\Algoritma dan Pemrograman\latihan> go run "c:\WATA KULIAH\Algoritma dan Pemrograman\latihan\guided2.go"
11 2
11
32 14
224
6 2
6
15 15
112.5
20 35
350
PS C:\WATA KULIAH\Algoritma dan Pemrograman\latihan> go run "c:\WATA KULIAH\Algoritma dan Pemrograman\latihan\guided2.go"
```

Deskripsi program

n: Untuk menyimpan jumlah total segitiga yang luasnya akan dihitung.

j: Sebagai variabel penghitung (counter) untuk perulangan.

alas: Untuk menyimpan nilai alas segitiga.

tinggi: Untuk menyimpan nilai tinggi segitiga.

Perintah `fmt.Scan(&n)` membaca nilai integer dan menyimpan ke variabel `n`, untuk menentukan berapa kali proses perhitungan luas segitiga akan diulang. Loop akan terus berjalan selama `J` kurang dari atau sama dengan nilai jumlah total segitiga(`n`). Lalu nilai `J` akan ditingkatkan sebesar 1(`j+1=1` atau `j++`). Hasilnya diubah menjadi Float 64 agar perkalian dengan 0,5 menghasilkan nilai luas yang akurat lalu proses output dengan perintah `fmt.Println(luas)` mencetak hasil perhitungan luas.

3. Guided 3

Source Code

```
package main

import "fmt"

func main() {

    var j, a, b int
    var hasil int

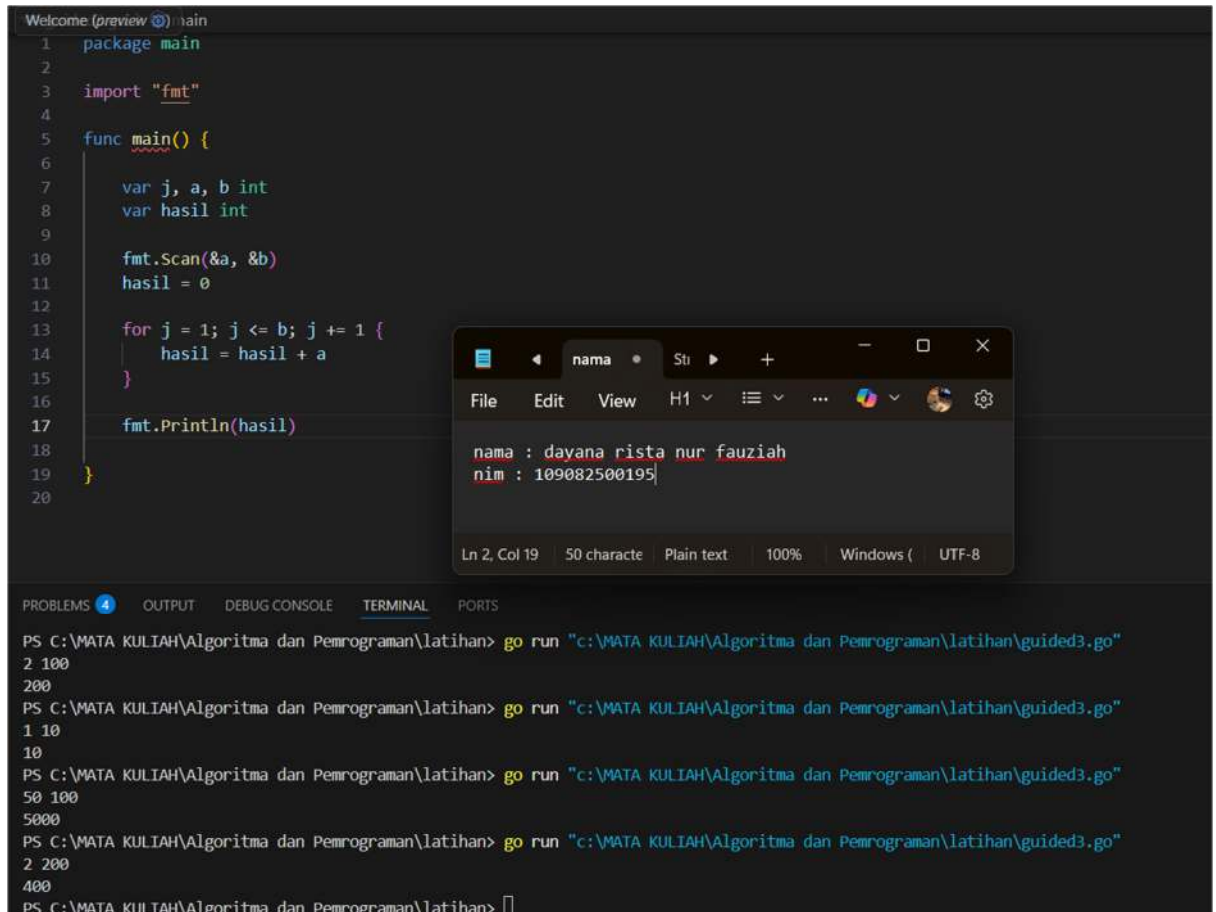
    fmt.Scan(&a, &b)
    hasil = 0

    for j = 1; j <= b; j += 1 {
        hasil = hasil + a
    }

    fmt.Println(hasil)

}
```

Screenshoot program



```
1 package main
2
3 import "fmt"
4
5 func main() {
6
7     var j, a, b int
8     var hasil int
9
10    fmt.Scan(&a, &b)
11    hasil = 0
12
13    for j = 1; j <= b; j += 1 {
14        hasil = hasil + a
15    }
16
17    fmt.Println(hasil)
18
19 }
20
```

```
PS C:\MATA KULIAH\Algoritma dan Pemrograman\latihan> go run "c:\MATA KULIAH\Algoritma dan Pemrograman\latihan\guided3.go"
2 100
200
PS C:\MATA KULIAH\Algoritma dan Pemrograman\latihan> go run "c:\MATA KULIAH\Algoritma dan Pemrograman\latihan\guided3.go"
2 200
400
PS C:\MATA KULIAH\Algoritma dan Pemrograman\latihan> go run "c:\MATA KULIAH\Algoritma dan Pemrograman\latihan\guided3.go"
1 10
10
PS C:\MATA KULIAH\Algoritma dan Pemrograman\latihan> go run "c:\MATA KULIAH\Algoritma dan Pemrograman\latihan\guided3.go"
50 100
5000
PS C:\MATA KULIAH\Algoritma dan Pemrograman\latihan>
```

Deskripsi program

Perintah `fmt.Scan(&a, &b)` membaca dua nilai integer dari pengguna. Angka pertama disimpan di `a` dan angka kedua di `b`. Variabel `hasil` diatur ke nilai 0, Variabel penghitung `j` diatur ke 1. Loop akan terus berjalan selama nilai `j` kurang dari atau sama dengan nilai variabel `b`. setelah `a` dijumlahkan sebanyak `b` kali, perintah `fmt.Println(hasil)` mencetak **nilai akhir** dari variabel `hasil` (yaitu hasil perkalian $a*b$)

TUGAS

1. Tugas 1

Source code

```
package main

import "fmt"

func main() {

    var n int
    var jumlah int

    fmt.Print("Masukkan bilangan bulat positif (n): ")
    fmt.Scan(&n)

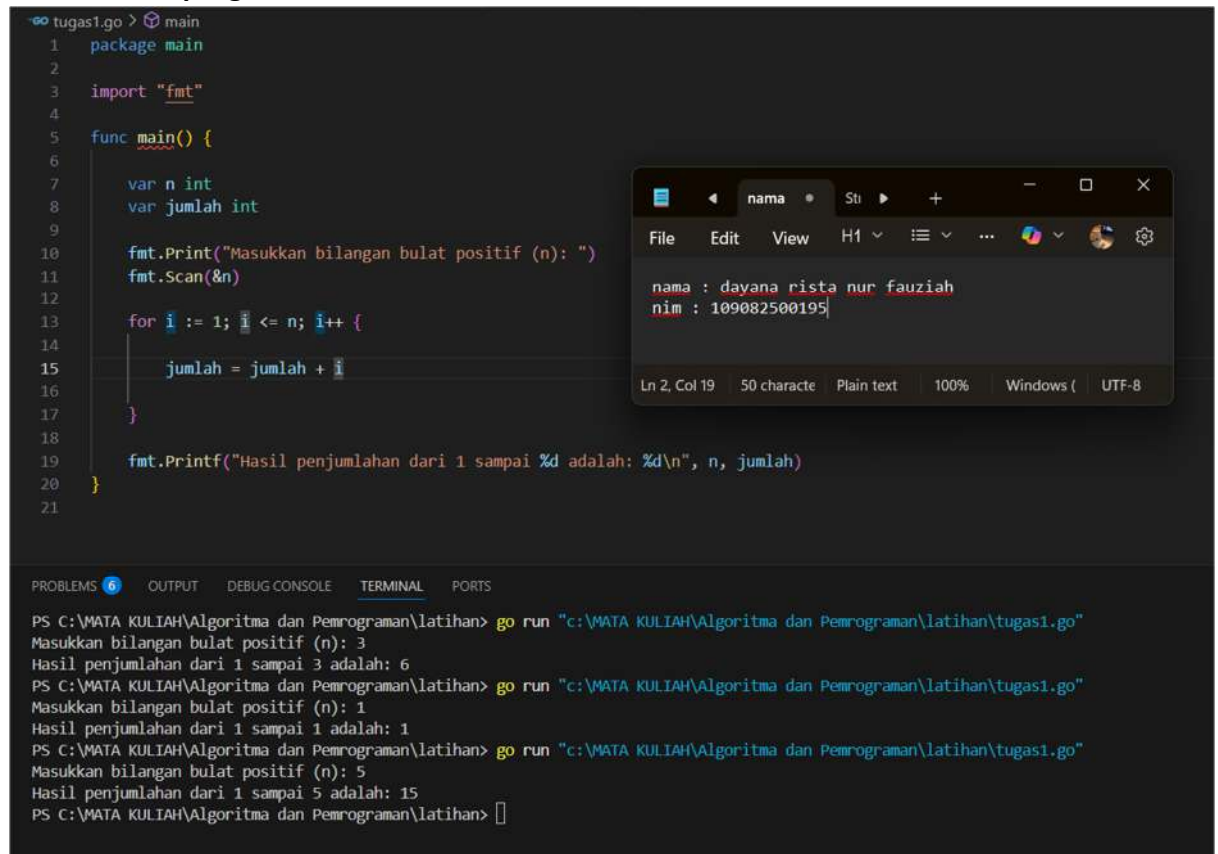
    for i := 1; i <= n; i++ {

        jumlah = jumlah + i

    }

    fmt.Printf("Hasil penjumlahan dari 1 sampai %d adalah: %d\n", n,
jumlah)
}
```


Screenshoot program



The screenshot shows a Go program in a text editor and its execution in a terminal. The program is a simple loop that calculates the sum of integers from 1 to n.

```
tugas1.go > main
1 package main
2
3 import "fmt"
4
5 func main() {
6
7     var n int
8     var jumlah int
9
10    fmt.Print("Masukkan bilangan bulat positif (n): ")
11    fmt.Scan(&n)
12
13    for i := 1; i <= n; i++ {
14
15        jumlah = jumlah + i
16
17    }
18
19    fmt.Printf("Hasil penjumlahan dari 1 sampai %d adalah: %d\n", n, jumlah)
20 }
21
```

The terminal output shows the program being run and the results of three test cases:

```
PS C:\MATA KULIAH\Algoritma dan Pemrograman\latihan> go run "c:\MATA KULIAH\Algoritma dan Pemrograman\latihan\tugas1.go"
Masukkan bilangan bulat positif (n): 3
Hasil penjumlahan dari 1 sampai 3 adalah: 6
PS C:\MATA KULIAH\Algoritma dan Pemrograman\latihan> go run "c:\MATA KULIAH\Algoritma dan Pemrograman\latihan\tugas1.go"
Masukkan bilangan bulat positif (n): 1
Hasil penjumlahan dari 1 sampai 1 adalah: 1
PS C:\MATA KULIAH\Algoritma dan Pemrograman\latihan> go run "c:\MATA KULIAH\Algoritma dan Pemrograman\latihan\tugas1.go"
Masukkan bilangan bulat positif (n): 5
Hasil penjumlahan dari 1 sampai 5 adalah: 15
PS C:\MATA KULIAH\Algoritma dan Pemrograman\latihan>
```

Deskripsi program

(i := 1) adalah Variabel penghitung i dimulai dari 1.

(i <= n) adalah Perulangan akan terus berjalan selama i masih kurang dari atau sama dengan nilai n yang dimasukkan.

(i++): Setelah setiap iterasi, nilai i bertambah 1.

(jumlah = jumlah + i) Di setiap putaran, nilai i saat ini (yaitu 1, 2, 3,..., n) ditambahkan ke dalam variabel jumlah.

Lalu Setelah perulangan selesai, program mencetak kalimat yang berisi nilai n dan nilai total jumlah sebagai hasil akhir.

2. Tugas 2

Source code

```
package main

import (
    "fmt"
    "math"
)

func main() {

    var n int
    var jariJari, tinggi float64
    var volume float64

    fmt.Print("Masukkan jumlah kerucut (n): ")
    fmt.Scan(&n)

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

        fmt.Scan(&jariJari, &tinggi)

        volume = (1.0 / 3.0) * math.Pi *
math.Pow(jariJari, 2) * tinggi

        fmt.Println( volume)
    }
}
```



Screenshoot program

```
3 import (
4     "fmt"
5     "math"
6 )
7
8 func main() {
9
10    var n int
11    var jariJari, tinggi float64
12    var volume float64
13
14    fmt.Print("Masukkan jumlah kerucut (n): ")
15    fmt.Scan(&n)
16
17    for i := 1; i <= n; i++ {
18
19        fmt.Scan(&jariJari, &tinggi)
20
21        volume = (1.0 / 3.0) * math.Pi * math.Pow(jariJari, 2) * tinggi
22
23        fmt.Println(volume)
24    }
25 }
```

nama : dayana rista nur fauziah
nim : 109082500195

Ln 2, Col 19 | 50 character | Plain text | 100% | Windows (| UTF-8

PROBLEMS 8 OUTPUT DEBUG CONSOLE TERMINAL PORTS

Masukkan jumlah kerucut (n): 1
3 4
37.699111843077524
PS C:\MATA KULIAH\Algoritma dan Pemrograman\latihan> go run "c:\MATA KULIAH\Algoritma dan Pemrograman\latihan\tugas2.go"
Masukkan jumlah kerucut (n): 3
1 1
1.0471975511965979
2 2
8.377580409572783
3 3
28.27433388230814
PS C:\MATA KULIAH\Algoritma dan Pemrograman\latihan>

Ln 23,

Deskripsi program

- Tipe float64 digunakan karena perhitungan volume (melibatkan phi dan pembagian $1/3$) akan menghasilkan angka desimal yang panjang dan harus disimpan secara akurat.
- `fmt.Scan(&n)` membaca baris pertama masukan, yaitu jumlah total kerucut yang akan diproses, dan menyimpannya di variable n.
- Perulangan `for i := 1; i <= n; i++` memastikan bahwa semua langkah di dalamnya akan diulang tepat sebanyak n kali.
- Perintah `fmt.Println(volume)` mencetak hasil perhitungan volume kerucut i ke layar, mengikuti format keluaran yang diminta.

3. Tugas 3

Source code

```
package main

import "fmt"

func main() {

    var a, b int
    var hasil int

    fmt.Print("Masukkan a dan b: ")
    fmt.Scan(&a, &b)

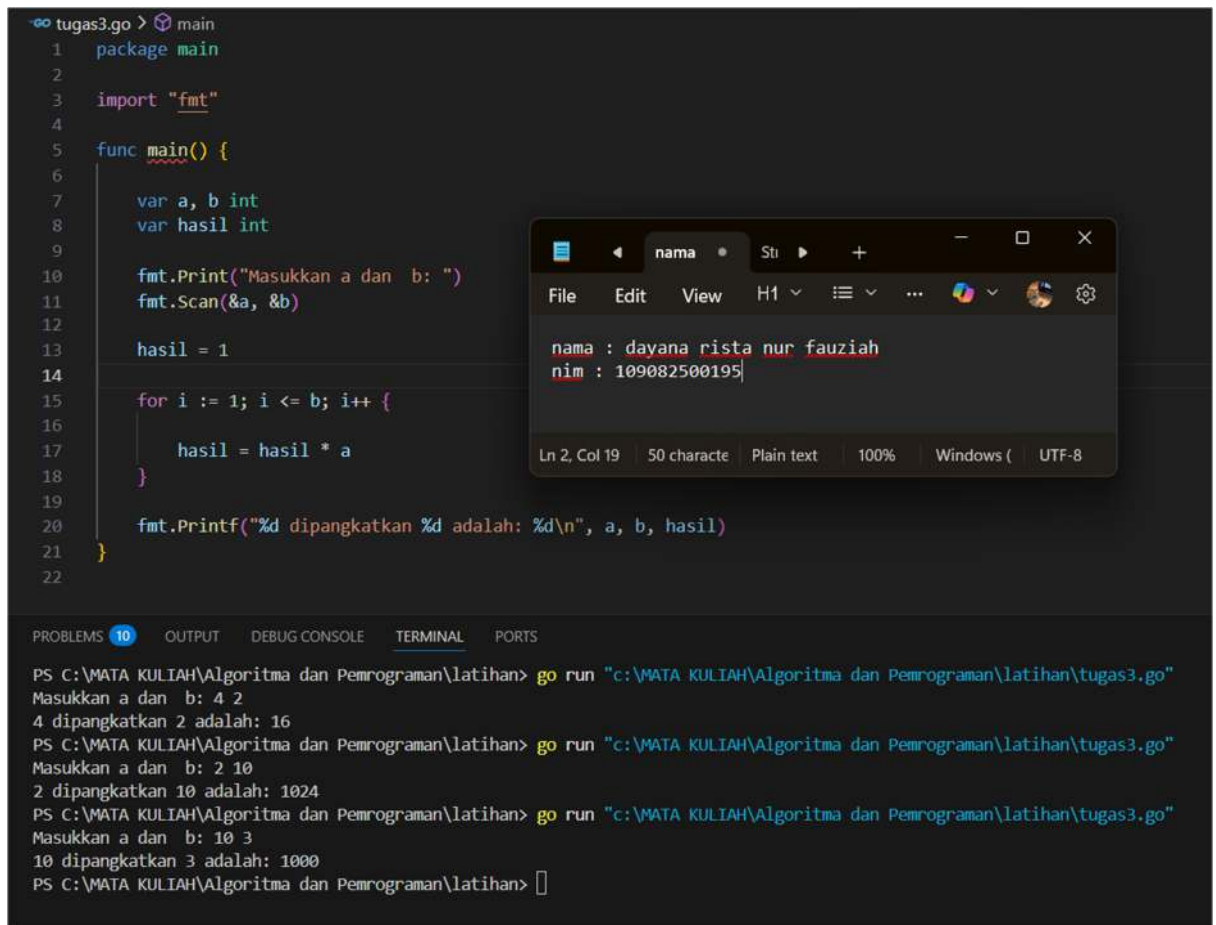
    hasil = 1

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

        hasil = hasil * a
    }

    fmt.Printf("%d dipangkatkan %d adalah: %d\n", a, b,
    hasil)
}
```

Screenshoot program



The screenshot shows a Go program in a text editor and its execution in a terminal. The program, `tugas3.go`, is a simple calculator that takes two integers `a` and `b` as input and calculates `a` raised to the power of `b` using a loop. The terminal shows three test cases: `4 2` resulting in `16`, `2 10` resulting in `1024`, and `10 3` resulting in `1000`.

```
tugas3.go > main
1 package main
2
3 import "fmt"
4
5 func main() {
6
7     var a, b int
8     var hasil int
9
10    fmt.Print("Masukkan a dan b: ")
11    fmt.Scan(&a, &b)
12
13    hasil = 1
14
15    for i := 1; i <= b; i++ {
16
17        hasil = hasil * a
18    }
19
20    fmt.Printf("%d dipangkatkan %d adalah: %d\n", a, b, hasil)
21 }
22
```

```
PS C:\MATA KULIAH\Algoritma dan Pemrograman\latihan> go run "c:\MATA KULIAH\Algoritma dan Pemrograman\latihan\tugas3.go"
Masukkan a dan b: 4 2
4 dipangkatkan 2 adalah: 16
PS C:\MATA KULIAH\Algoritma dan Pemrograman\latihan> go run "c:\MATA KULIAH\Algoritma dan Pemrograman\latihan\tugas3.go"
Masukkan a dan b: 2 10
2 dipangkatkan 10 adalah: 1024
PS C:\MATA KULIAH\Algoritma dan Pemrograman\latihan> go run "c:\MATA KULIAH\Algoritma dan Pemrograman\latihan\tugas3.go"
Masukkan a dan b: 10 3
10 dipangkatkan 3 adalah: 1000
PS C:\MATA KULIAH\Algoritma dan Pemrograman\latihan>
```

Deskripsi program

- Perintah `fmt.Scan(&a, &b)` akan membaca dua bilangan bulat dari pengguna.
Contoh: Jika pengguna memasukkan `4 2`, maka `a=4` dan `b=2`.
- `hasil = 1`. Karena kita akan menggunakan perkalian berulang, nilai awal harus `1`.
(Jika diinisialisasi `0`, hasil perkalian apapun akan selalu `0`.)
- Perulangan `for` ini dirancang untuk berjalan tepat sebanyak nilai dari `b` kali.
- Setelah loop selesai, variabel `hasil` sudah menyimpan nilai akhir dari perpangkatan (`16`),

4. Tugas 4

Source code

```
package main

import "fmt"

func main() {

    var n int

    fmt.Print("Masukkan bilangan bulat non-negatif (n): ")
    fmt.Scan(&n)

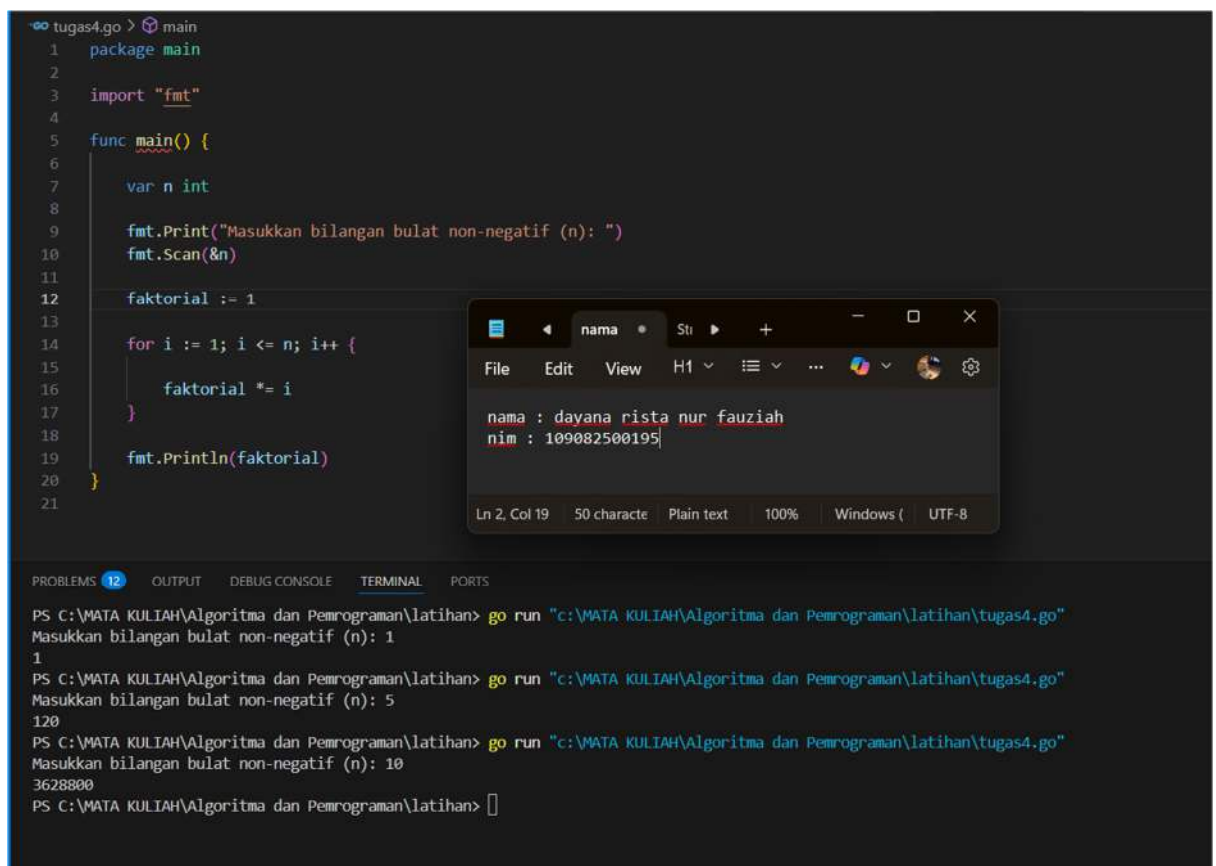
    faktorial := 1

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

        faktorial *= i
    }

    fmt.Println(faktorial)
}
```

Screenshoot program



The screenshot shows a Go program in a code editor and its execution in a terminal. The program calculates the factorial of a non-negative integer. The terminal shows three runs: for input 1, 5, and 10, with the corresponding factorial results.

```
tugas4.go > main
1  package main
2
3  import "fmt"
4
5  func main() {
6
7      var n int
8
9      fmt.Print("Masukkan bilangan bulat non-negatif (n): ")
10     fmt.Scan(&n)
11
12     faktorial := 1
13
14     for i := 1; i <= n; i++ {
15
16         faktorial *= i
17     }
18
19     fmt.Println(faktorial)
20
21 }
```

nama : dayana rista nur fauziah
nim : 109082500195

PS C:\MATA KULIAH\Algoritma dan Pemrograman\latihan> go run "c:\MATA KULIAH\Algoritma dan Pemrograman\latihan\tugas4.go"
Masukkan bilangan bulat non-negatif (n): 1
1
PS C:\MATA KULIAH\Algoritma dan Pemrograman\latihan> go run "c:\MATA KULIAH\Algoritma dan Pemrograman\latihan\tugas4.go"
Masukkan bilangan bulat non-negatif (n): 5
120
PS C:\MATA KULIAH\Algoritma dan Pemrograman\latihan> go run "c:\MATA KULIAH\Algoritma dan Pemrograman\latihan\tugas4.go"
Masukkan bilangan bulat non-negatif (n): 10
3628800
PS C:\MATA KULIAH\Algoritma dan Pemrograman\latihan>

Deskripsi program

- Variabel faktorial dideklarasikan dan langsung diatur ke nilai 1.
- Perulangan (for loop) dimulai dengan variabel i dari 1 dan terus bertambah (i++) selama i masih kurang dari atau sama dengan n. Perulangan ini akan mengulang operasi perkalian sebanyak n kali.
- Perintah faktorial *= i adalah bentuk singkat dari faktorial = faktorial * i.
- Setelah perulangan selesai, variabel **faktorial** sudah berisi nilai akhir dari n (dalam contoh, 24), dan nilai ini dicetak ke layar.