

LAPORAN PRAKTIKUM
Algoritma Pemrograman

MODUL 5 & 6
FOR-LOOP



Disusun oleh:

LEONARDO FARRIZ GARCYA

109082530036

S1IF-13-04

PROGRAM STUDI S1 INFORMATIKA
FAKULTAS INFORMATIKA
TELKOM UNIVERSITY PURWOKERTO
2025

LATIHAN KELAS – GUIDED

1. Guided 1 Source Code

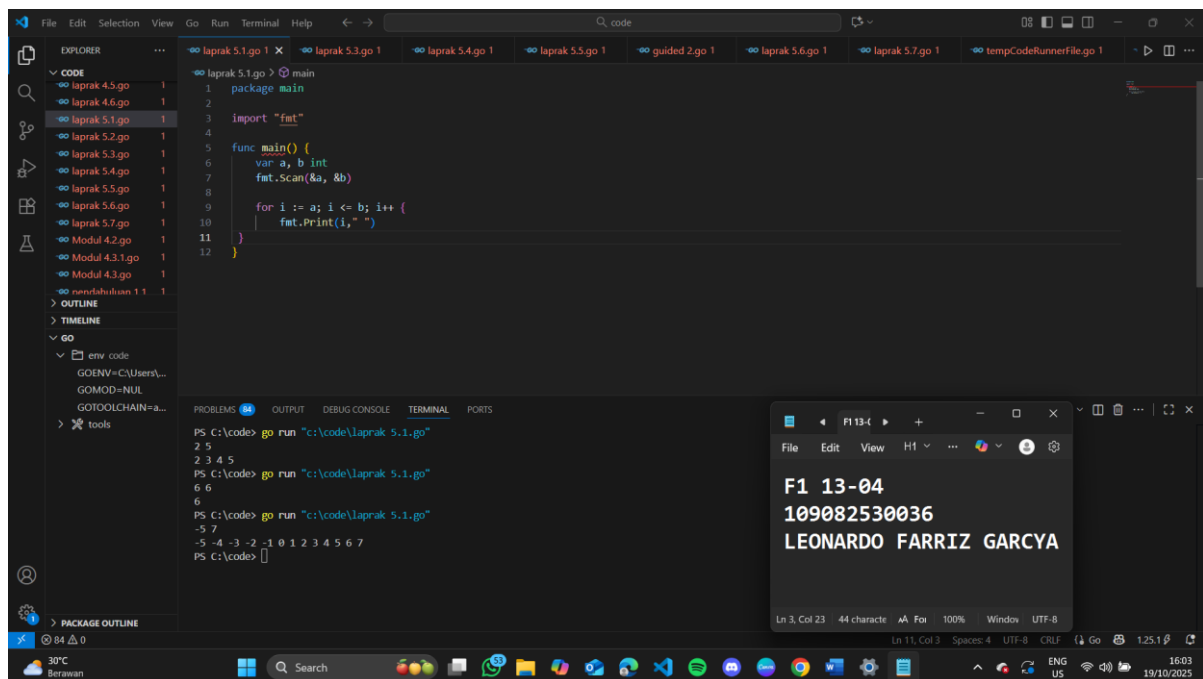
```
package main

import "fmt"

func main() {
    var a, b int
    fmt.Scan(&a, &b)

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

Screenshoot program



Deskripsi program

1.package main

- program ini adalah program utama yang bisa langsung dijalankan.
- Tanpa main, kode tidak bisa dieksekusi sebagai aplikasi.

2.import "fmt"

- Format (fmt), digunakan untuk input/output standar.
- Seperti "fmt.Scanln" : untuk membaca input dari user.
- "fmt.Println" : menampilkan output ke layar.

3.func main () { ... }

- Fungsi main adalah untuk titik awal eksekusi program Go.
- Semua intruksi akan berjalan dari fungsi ini.

4.var a, b int

- Menyatakan dua variabel a dan b dengan tipe data bilangan bulat.
- Variabel ini digunakan untuk menyimpan dua angka yang dimasukkan oleh pengguna

5.fmt.Scan(&a, &b)

- Membaca dua input dari pengguna, lalu menyimpannya ke dalam variabel a dan b.
- & artinya mengambil alamat memori dari variabel, agar fmt.Scan bisa mengisi Nilainya

6. for i := a; i <= b; i++ { ... }

- Ini adalah perulangan for.
- Mulai dari **i = a**, selama **i** kurang dari atau sama dengan **b**, lakukan perintah di dalam kurung **{}** dan setiap kali tambahkan **i** satu (**i++**).

7. `fmt.Print(i, " ")`

- Setiap kali perulangan berjalan, program akan mencetak nilai **i** diikuti dengan spasi " ".
- Tidak ada baris baru, jadi semua angka akan ditampilkan di satu baris.

2. Guided 2

Source Code

```
package main

import "fmt"

func main() {
    var n int
    fmt.Scan(&n)

    for i := 0; i < n; i++ {
        var alas, tinggi float64
        fmt.Scan(&alas, &tinggi)
        luas := (alas * tinggi) / 2
        fmt.Println(luas)
    }
}
```

Screenshoot program

```
1 package main
2
3 import "fmt"
4
5 func main() {
6     var n int
7     fmt.Scan(&n)
8
9     for i := 0; i < n; i++ {
10        var alas, tinggi float64
11        fmt.Scan(&alas, &tinggi)
12        luas := (alas * tinggi) / 2
13        fmt.Println(luas)
14    }
15 }
```

Terminal Output:

```
PS C:\code> go run "c:\code\laprak 5.2.go"
5
11 2
11
32 14
224
6 2
6
15 15
112.5
20 35
350
```

Deskripsi program

1.package main

- program ini adalah program utama yang bisa langsung dijalankan.
- Tanpa main, kode tidak bisa dieksekusi sebagai aplikasi.

2.import "fmt"

- Format (fmt), digunakan untuk input/output standar.
- Seperti "fmt.Scanln" : untuk membaca input dari user.
- "fmt.Println" : menampilkan output ke layar.

3.func main () { ... }

- Fungsi main adalah untuk titik awal eksekusi program Go.
- Semua intruksi akan berjalan dari fungsi ini.

4. var n int

- Deklarasi variabel n bertipe bilangan bulat, yang akan digunakan untuk menyimpan jumlah data.

5.fmt.Scan(&n)

- Untuk membaca input pengguna dan menyimpannya ke dalam variabel n.
- & berarti alamat memori dari variabel n, supaya nilai dari input bisa disimpan di sana.

6. for i := 0; i < n; i++ { ... }

- Perulangan (loop) yang akan berjalan sebanyak n kali.
- Jika **n = 3**, maka perulangan terjadi untuk **i = 0, 1, 2**.

7. `var alas, tinggi float64`
 - menyatakan dua variabel bertipe `float64`, yaitu `alas` dan `tinggi`, untuk menampung nilai `alas` dan `tinggi` segitiga (bilangan desimal).
8. `fmt.Scan(&alas, &tinggi)`
 - Perintah untuk membaca dua nilai input dari pengguna:
nilai pertama untuk `alas`
nilai kedua untuk `tinggi`.
9. `luas := (alas * tinggi) / 2`
 - Rumus untuk menghitung luas segitiga.
10. `fmt.Println(luas)`
 - Menampilkan hasil perhitungan luas segitiga ke layar.

3. Guided 3

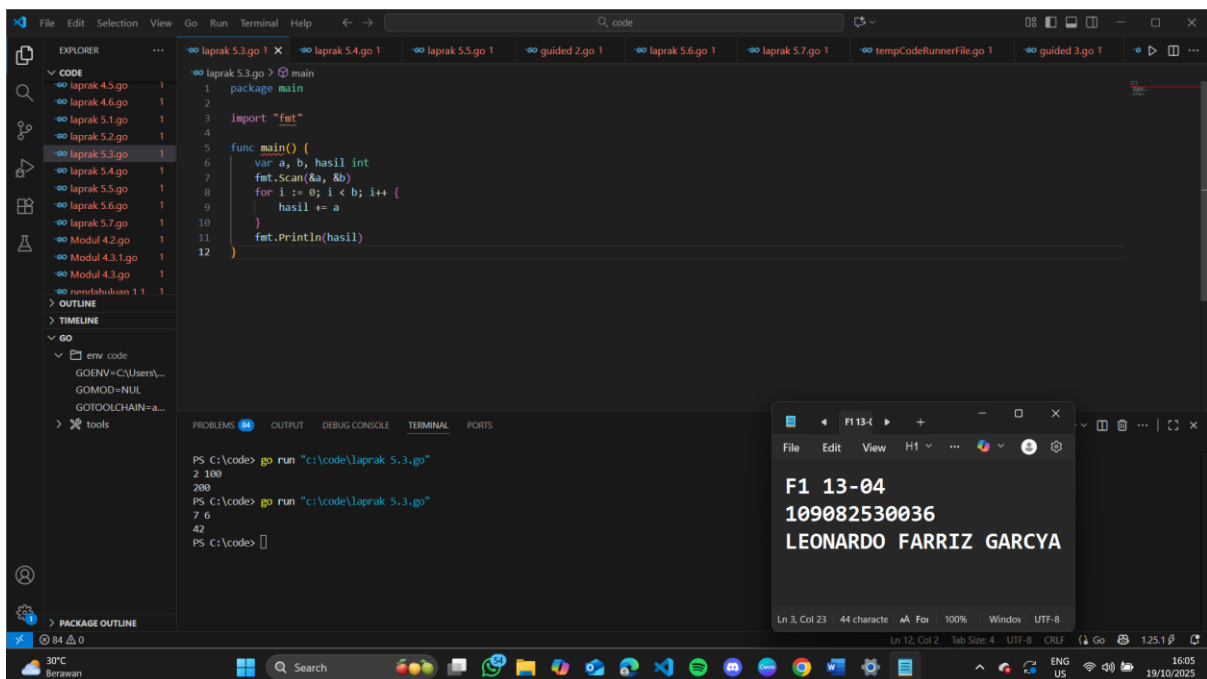
Source Code

```
package main

import "fmt"

func main() {
    var a, b, hasil int
    fmt.Scan(&a, &b)
    for i := 0; i < b; i++ {
        hasil += a
    }
    fmt.Println(hasil)
}
```

Screenshoot program



Deskripsi program

1.package main

- program ini adalah program utama yang bisa langsung dijalankan.
- Tanpa main, kode tidak bisa dieksekusi sebagai aplikasi.

2.import "fmt"

- Format (fmt), digunakan untuk input/output standar.
- Seperti "fmt.Scanln" : untuk membaca input dari user.
- "fmt.Println" : menampilkan output ke layar.

3.func main () { ... }

- Fungsi main adalah untuk titik awal eksekusi program Go.
- Semua intruksi akan berjalan dari fungsi ini.

4. var a, b, hasil int

Menyatakan tiga variabel bertipe integer:

a : bilangan pertama (faktor pertama)

b : bilangan kedua (faktor kedua)

hasil : variabel untuk menyimpan hasil perhitungan

5.fmt.Scan(&a, &b)

- Membaca dua input dari pengguna dan menyimpannya ke variabel **a** dan **b**.
- Tanda **&** digunakan untuk memberikan alamat memori variabel agar nilainya bisa diubah oleh Scan.

6. `for i := 0; i < b; i++ { hasil += a }`

- `for i := 0`; untuk membuat variabel perhitungan `i` dengan nilai awal 0.

- `i < b`; , selama `i` masih kecil dari `b`, maka blok di `{}` akan terus berjalan dan `i++`, nilai `i` akan bertambah satu

- `hasil += a`, sebagai perulangan, jika `i > b` maka perulangan akan berhenti.

7. `fmt.Println(hasil)`

Perintah ini menampilkan hasil akhir dari perhitungan ke layar.

TUGAS

1. Tugas 1

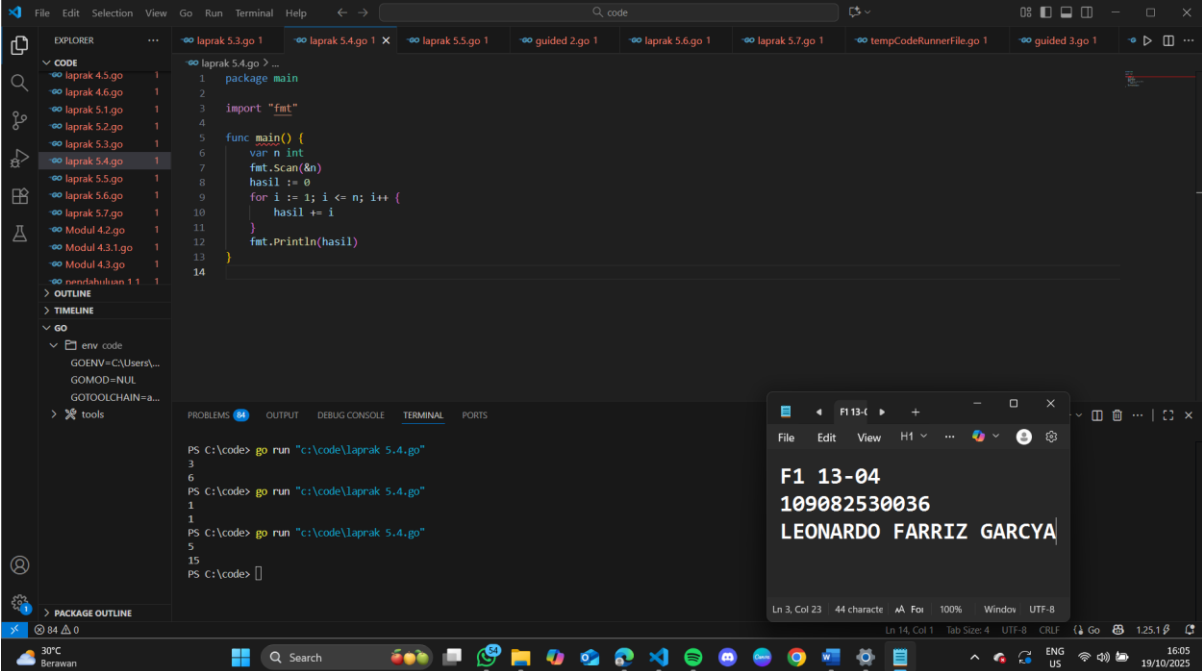
Source code

```
package main

import "fmt"

func main() {
    var n int
    fmt.Scan(&n)
    hasil := 0
    for i := 1; i <= n; i++ {
        hasil += i
    }
    fmt.Println(hasil)
}
```


Screenshoot program



```
1 package main
2
3 import "fmt"
4
5 func main() {
6     var n int
7     fmt.Scan(&n)
8     hasil := 0
9     for i := 1; i <= n; i++ {
10         hasil += i
11     }
12     fmt.Println(hasil)
13 }
14
```

Terminal output:

```
PS C:\code> go run "c:\code\laprak 5.4.go"
3
6
PS C:\code> go run "c:\code\laprak 5.4.go"
1
1
PS C:\code> go run "c:\code\laprak 5.4.go"
5
15
PS C:\code>
```

Output window:

```
F1 13-04
109082530036
LEONARDO FARRIZ GARCIA
```

Deskripsi program

1.package main

- program ini adalah program utama yang bisa langsung dijalankan.
- Tanpa main, kode tidak bisa dieksekusi sebagai aplikasi.

2.import "fmt"

- Format (fmt), digunakan untuk input/output standar.
- Seperti "fmt.Scanln" : untuk membaca input dari user.
- "fmt.Println" : menampilkan output ke layar.

3.func main () { ... }

- Fungsi main adalah untuk titik awal eksekusi program Go.
- Semua intruksi akan berjalan dari fungsi ini.

4.var n int

- Menyatakan variabel n bertipe integer (bilangan bulat).

5. fmt.Scan(&n)

- Membaca input dari pengguna dan menyimpannya ke variabel **n**
- Tanda **&** digunakan untuk memberikan alamat memori variabel agar nilainya bisa diubah oleh Scan.

6.hasil := 0

- Inisialisasi variabel hasil dengan nilai awal 0.

7. for i := 1; i <= n; i++ { hasil += i }

- for i := 1 , untuk membuat variabel perhitungan i dengan nilai awal 1.
- i <= n , perulangan akan terus berjalan selama i masih kurang dari atau sama dengan n dan i++ untuk setiap kali satu putaran selesai, nilai i akan bertambah 1.
- hasil += i , setiap kali perulangan berjalan, nilai i ditambahkan ke variabel hasil.

8. fmt.Println(hasil)

- Perintah ini menampilkan hasil akhir dari perhitungan, ke layar.

2. Tugas 2

Source code

```
package main

import "fmt"

func main() {

    const pi = 3.141592653589793

    var n int

    fmt.Scan(&n)

    for i := 0; i < n; i++ {

        var r, t float64

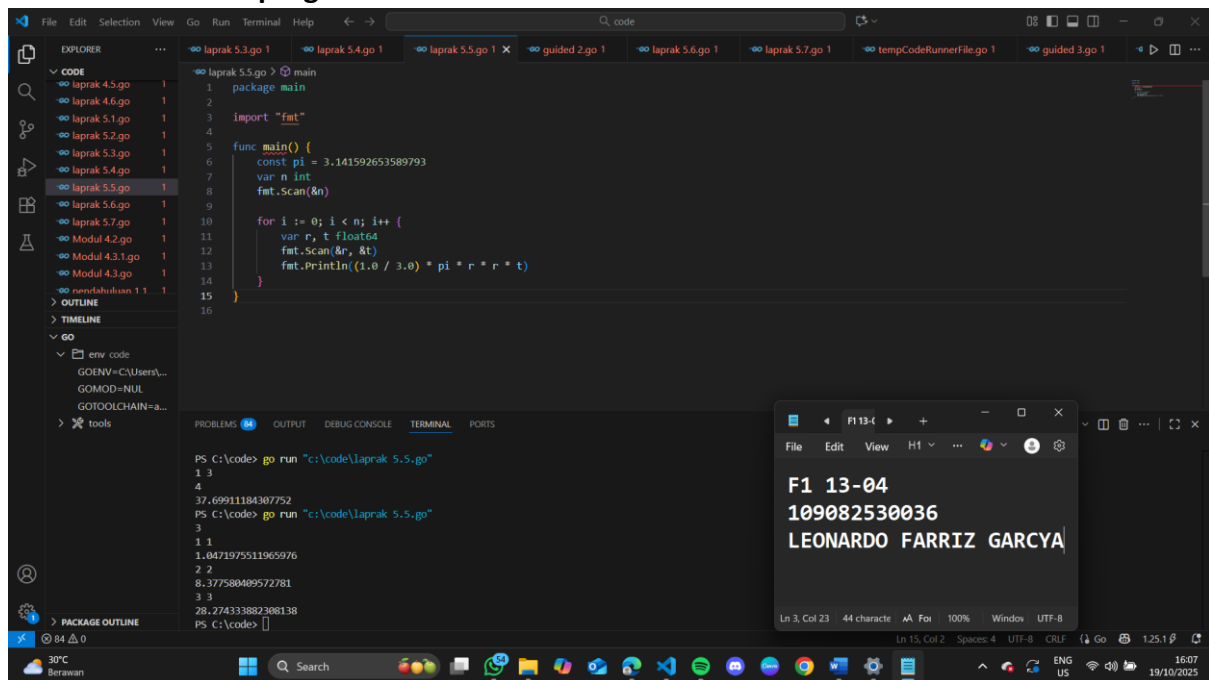
        fmt.Scan(&r, &t)

        fmt.Println((1.0 / 3.0) * pi * r * r * t)

    }

}
```

Screenshoot program



```
1 package main
2
3 import "fmt"
4
5 func main() {
6     const pi = 3.141592653589793
7     var n int
8     fmt.Scan(&n)
9
10    for i := 0; i < n; i++ {
11        var r, t float64
12        fmt.Scan(&r, &t)
13        fmt.Println((1.0 / 3.0) * pi * r * r * t)
14    }
15 }
16
```

PS C:\code> go run "c:\code\laprak 5.5.go"

1 3
4
37.69911184307752
PS C:\code> go run "c:\code\laprak 5.5.go"

3
1 1
1.0471975511965976
2 2
0.377580409572781
3 3
28.274333882308138
PS C:\code>

F1 13-04
109082530036
LEONARDO FARRIZ GARCYA

Deskripsi program

1.package main

- program ini adalah program utama yang bisa langsung dijalankan.
- Tanpa main, kode tidak bisa dieksekusi sebagai aplikasi.

2.import "fmt"

- Format (fmt), digunakan untuk input/output standar.
- Seperti "fmt.Scanln" : untuk membaca input dari user.
- "fmt.Println" : menampilkan output ke layar.

3.func main () { ... }

- Fungsi main adalah untuk titik awal eksekusi program Go.
- Semua intruksi akan berjalan dari fungsi ini.

4.const pi = 3.141592653589793

- Mendefinisikan **konstanta** bernama pi ,pi adalah nilai tetap

5.var n int

- Membuat variabel n bertipe bilangan bulat.
- Variabel ini akan menyimpan jumlah data yang akan diproses.

6.fmt.Scan(&n)

- Membaca input dari pengguna dan menyimpannya ke variabel **n**
- Tanda **&** digunakan untuk memberikan alamat memori variabel agar nilainya bisa diubah oleh Scan.

7. `for i := 0; i < n; i++ { ... }`

- `for i := 0` , perulangan di mulai dari 0 dan `i < n` , perulangan akan berhenti jika `i` sudah sama dengan `n`.

- `i++` , setiap perulangan, nilai `i` bertambah 1

8. `fmt.Scan(&r, &t)`

- Untuk membaca dua input dari pengguna (nilai `r` dan `t`) setiap kali perulangan dijalankan.

9. `fmt.Println((1.0 / 3.0) * pi * r * r * t)`

- untuk menghitung rumus volume kerucut dan langsung menampilkan hasilnya di layar.

3. Tugas 3

Source code

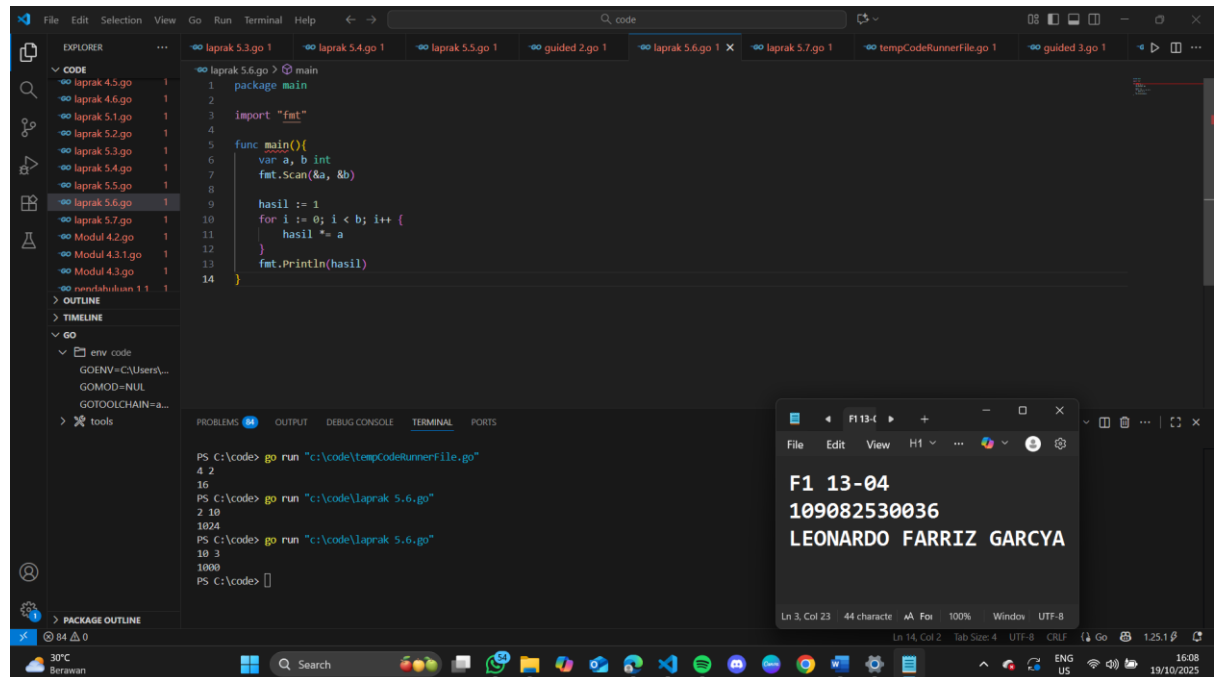
```
package main

import "fmt"

func main(){
    var a, b int
    fmt.Scan(&a, &b)

    hasil := 1
    for i := 0; i < b; i++ {
        hasil *= a
    }
    fmt.Println(hasil)
}
```

Screenshoot program



Deskripsi program

1. package main

- program ini adalah program utama yang bisa langsung dijalankan.
- Tanpa main, kode tidak bisa dieksekusi sebagai aplikasi.

2. import "fmt"

- Format (fmt), digunakan untuk input/output standar.
- Seperti "fmt.Scanln" : untuk membaca input dari user.
- "fmt.Println" : menampilkan output ke layar.

3. func main () { ... }

- Fungsi main adalah untuk titik awal eksekusi program Go.
- Semua intruksi akan berjalan dari fungsi ini.

4. var a, b int

Menyatakan tiga variabel bertipe integer:

a : sebagai bilangan dasar (basis)

b : sebagai bilangan pangkat (eksponen)

5. fmt.Scan(&a, &b)

- Membaca dua input dari pengguna dan menyimpannya ke variabel **a** dan **b**.
- Tanda **&** digunakan untuk memberikan alamat memori variabel agar nilainya bisa diubah oleh Scan.

6. for i := 0; i < b; i++ { hasil *= a }

- berfungsi untuk mengalikan **a** sebanyak **b** kali, yaitu cara manual untuk menghitung pangkat.

- $i := 0 \rightarrow$ perulangan dimulai dari **0** dan $i < b$, akan terus berulang sampai i mencapai **$b - 1$** .
- $i++$, setiap perulangan, i bertambah 1.
- $hasil *= a$, artinya sama dengan $hasil = hasil * a$.

7.fmt.Println(hasil)

- Perintah ini menampilkan hasil akhir dari perhitungan, ke layar.

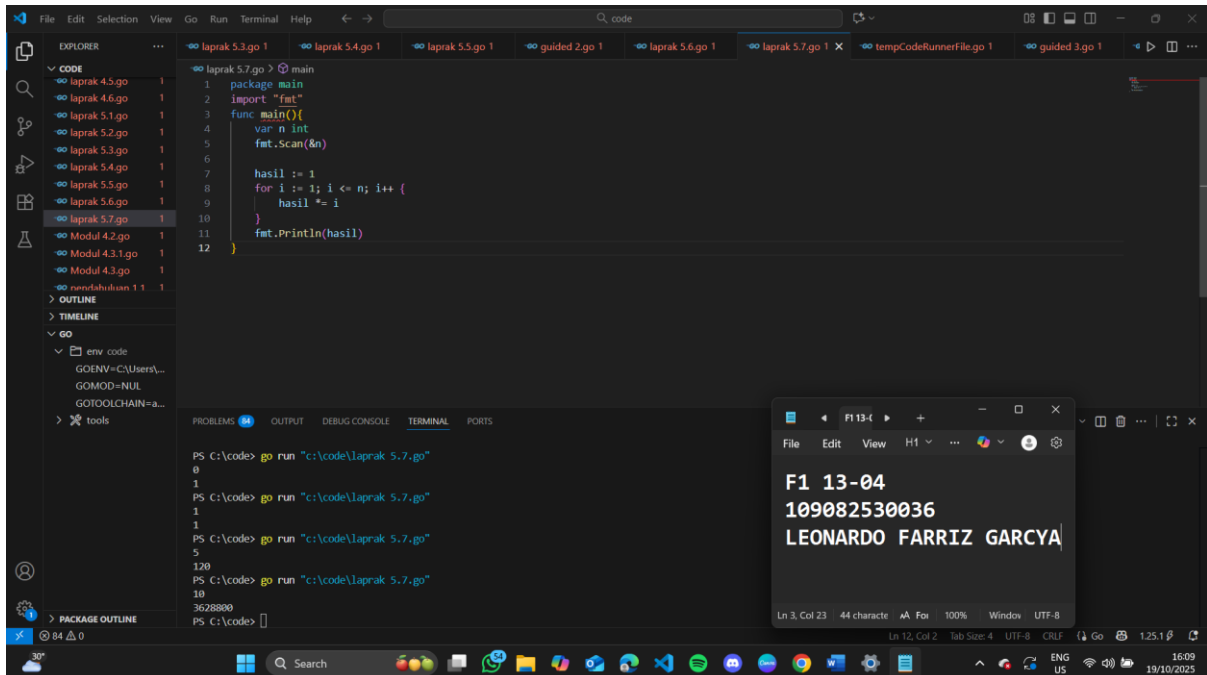
4. Tugas 4

Source code

```
package main
import "fmt"
func main(){
    var n int
    fmt.Scan(&n)

    hasil := 1
    for i := 1; i <= n; i++ {
        hasil *= i
    }
    fmt.Println(hasil)
}
```

Screenoot program



Deskripsi program

1.package main

- program ini adalah program utama yang bisa langsung dijalankan.
- Tanpa main, kode tidak bisa dieksekusi sebagai aplikasi.

2.import "fmt"

- Format (fmt), digunakan untuk input/output standar.
- Seperti "fmt.Scanln" : untuk membaca input dari user.
- "fmt.Println" : menampilkan output ke layar.

3.func main () { ... }

- Fungsi main adalah untuk titik awal eksekusi program Go.
- Semua intruksi akan berjalan dari fungsi ini.

4. var n int

- Membuat variabel n bertipe bilangan bulat.
- Variabel ini akan menyimpan jumlah data yang akan diproses.

5.fmt.Scan(&n)

- Membaca input dari pengguna dan menyimpannya ke variabel **n**
- Tanda **&** digunakan untuk memberikan alamat memori variabel agar nilainya bisa diubah oleh Scan.

6.hasil := 1

- Inisialisasi variabel hasil dengan nilai awal 1.

7. for i := 1; i <= n; i++ { hasil *= i }

- Perulangan (loop) yang digunakan untuk menghitung nilai faktorial.

- for i := 1; perulangan dimulai dari angka 1 dan i <= n , perulangan berjalan sampai i mencapai n.

- i++ , setiap kali perulangan selesai, i bertambah 1.

- hasil *= i → artinya hasil = hasil * i,

8. fmt.Println(hasil)

- Perintah ini menampilkan hasil akhir dari perhitungan, ke layar.