

**LAPORAN PRAKTIKUM ALGORITMA
DAN PEMROGRAMAN 1**

**MODUL 12
WHILE-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

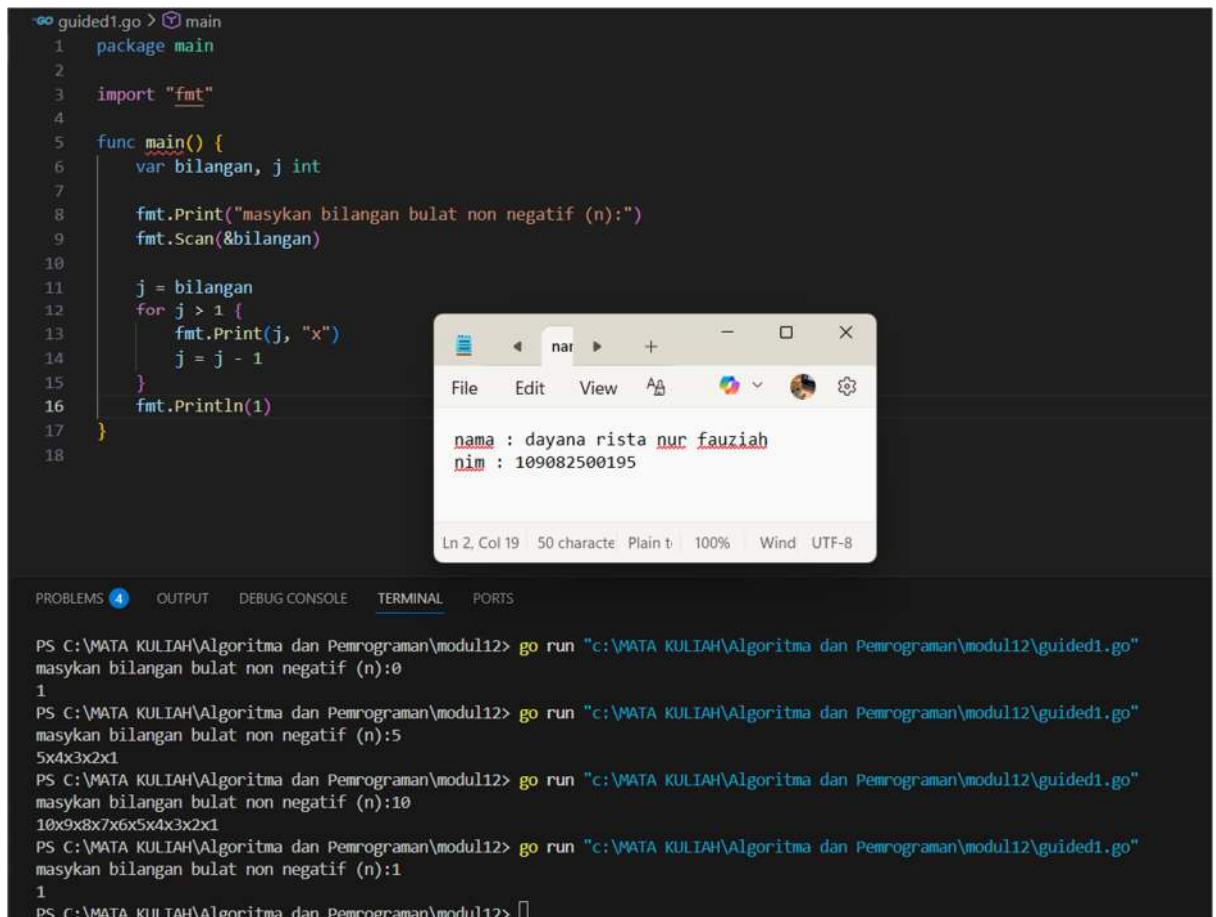
import "fmt"

func main() {
    var bilangan, j int

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

    j = bilangan
    for j > 1 {
        fmt.Print(j, "x")
        j = j - 1
    }
    fmt.Println(1)
}
```

Screenshoot program



The screenshot displays a Go program in a code editor and its execution results in a terminal. The program, named `guided1.go`, is located in the `main` package and imports the `fmt` package. It defines a `main` function that prompts the user to enter a non-negative integer `n`. It then uses a `for` loop to print the factorial of `n` in descending order, with each number followed by a space. The terminal shows the program being run three times with inputs 0, 5, and 10, resulting in outputs 1, 5x4x3x2x1, and 10x9x8x7x6x5x4x3x2x1 respectively.

```
guided1.go > main
1 package main
2
3 import "fmt"
4
5 func main() {
6     var bilangan, j int
7
8     fmt.Print("masykan bilangan bulat non negatif (n):")
9     fmt.Scan(&bilangan)
10
11     j = bilangan
12     for j > 1 {
13         fmt.Print(j, "x")
14         j = j - 1
15     }
16     fmt.Println(1)
17 }
18
```

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

```
PS C:\MATA KULIAH\Algoritma dan Pemrograman\modul12> go run "c:\MATA KULIAH\Algoritma dan Pemrograman\modul12\guided1.go"
masykan bilangan bulat non negatif (n):0
1
PS C:\MATA KULIAH\Algoritma dan Pemrograman\modul12> go run "c:\MATA KULIAH\Algoritma dan Pemrograman\modul12\guided1.go"
masykan bilangan bulat non negatif (n):5
5x4x3x2x1
PS C:\MATA KULIAH\Algoritma dan Pemrograman\modul12> go run "c:\MATA KULIAH\Algoritma dan Pemrograman\modul12\guided1.go"
masykan bilangan bulat non negatif (n):10
10x9x8x7x6x5x4x3x2x1
PS C:\MATA KULIAH\Algoritma dan Pemrograman\modul12> go run "c:\MATA KULIAH\Algoritma dan Pemrograman\modul12\guided1.go"
masykan bilangan bulat non negatif (n):1
1
PS C:\MATA KULIAH\Algoritma dan Pemrograman\modul12>
```

Deskripsi program

Program meminta untuk memasukkan sebuah bilangan bulat non-negatif, lalu mencetak representasi perkalian faktorialnya secara mundur. Misalnya, masukkan 4, outputnya adalah 4x3x2x1. Program menggunakan perulangan `for` untuk menghitung mundur dari `n` sampai 2, mencetak setiap angka diikuti `x`, dan terakhir mencetak angka 1.

2. Guided 2

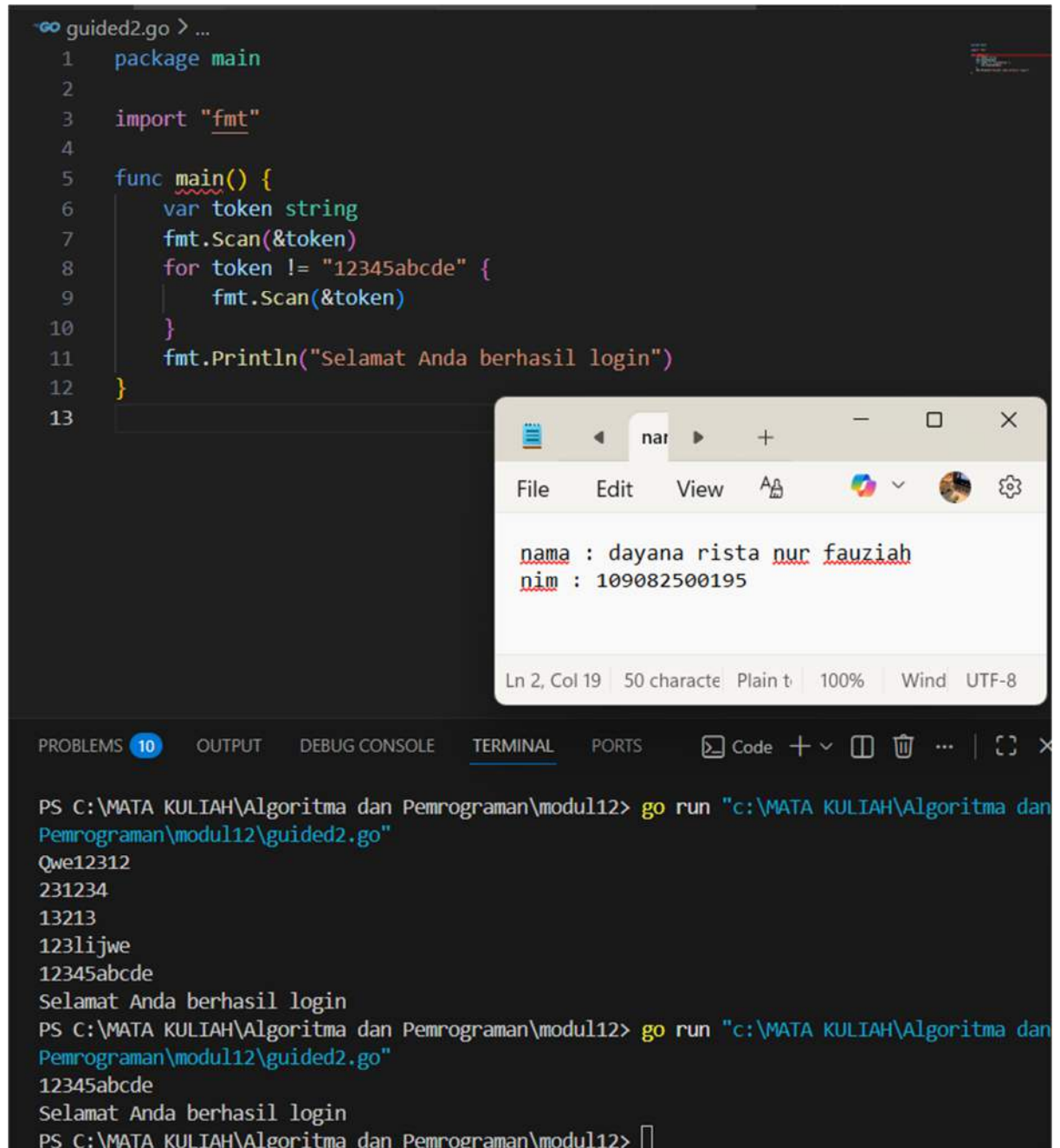
Source Code

```
package main

import "fmt"

func main() {
    var token string
    fmt.Scan(&token)
    for token != "12345abcde" {
        fmt.Scan(&token)
    }
    fmt.Println("Selamat Anda berhasil login")
}
```

Screenshoot program



The screenshot shows a Go program in a code editor and its execution in a terminal. The code is as follows:

```
1 package main
2
3 import "fmt"
4
5 func main() {
6     var token string
7     fmt.Scan(&token)
8     for token != "12345abcde" {
9         fmt.Scan(&token)
10    }
11    fmt.Println("Selamat Anda berhasil login")
12 }
13
```

The terminal shows the program being run with the command `go run "c:\MATA KULIAH\Algoritma dan Pemrograman\modul12\guided2.go"`. The input sequence is: `Qwe12312`, `231234`, `13213`, `123lijwe`, and `12345abcde`. The output is `Selamat Anda berhasil login`. The program is then run again with the same command and input sequence, producing the same output.

Deskripsi program

program akan terus meminta *input* berupa *string* (disimpan di variabel *token*) dari pengguna berulang kali selama *input* yang dimasukkan bukan "12345abcde". Perulangan *for* akan terus berjalan selama kondisi *token* `!= "12345abcde"` bernilai benar. Begitu pengguna berhasil memasukkan *string* "12345abcde", kondisi perulangan menjadi salah, program keluar dari perulangan, dan mencetak pesan "Selamat Anda berhasil login".

3. Guided 3

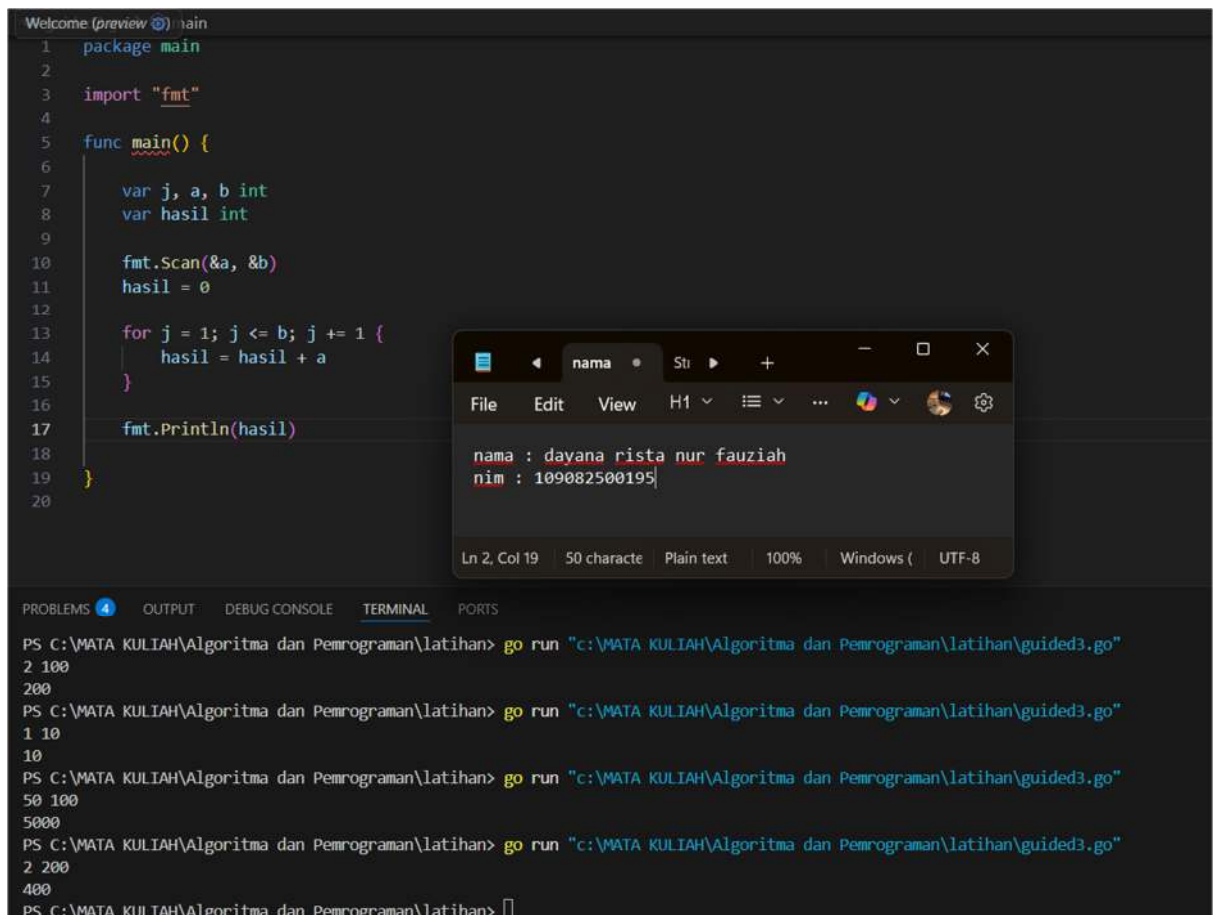
Source Code

```
package main

import "fmt"

func main() {
    var N, x, y, j, temp int
    fmt.Scan(&N)
    x = 0
    y = 1
    j = 0
    for j < N {
        fmt.Print(x, " ")
        temp = x + y
        x = y
        y = temp
        j = j + 1
    }
}
```

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
```

```
File Edit View H1 ... 100% Windows UTF-8
nama : dayana rista nur fauziah
nim : 109082500195
```

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

Deskripsi program

dimulai dengan menetapkan dua angka awal deret, yaitu x disetel ke 0 dan y disetel ke 1. Selanjutnya, program menjalankan pengulangan (for) sebanyak N kali. Dalam setiap putaran, angka Fibonacci saat ini (x) dicetak, dan angka berikutnya (temp) dihitung dengan menjumlahkan x dan y. untuk melanjutkan deret, program kemudian memperbarui nilai, di mana x mengambil nilai y sebelumnya, dan y mengambil nilai temp yang baru dihitung, proses ini terus berlanjut hingga seluruh N suku ditampilkan.

TUGAS

1. Tugas 1

Source code

```
package main

import "fmt"

func main() {
    var user, pass string
    gagal := 0

    for user != "Admin" || pass != "Admin" {
        fmt.Scan(&user, &pass)

        if user != "Admin" || pass != "Admin" {
            gagal++ // sama aja kaya (gagal=gagal+1)
        }
    }

    fmt.Println(gagal, "percobaan gagal login")
}
```


Screenshot program

```

1 package main
2
3 import "fmt"
4
5 func main() {
6     var user, pass string
7     gagal := 0
8
9     for user != "Admin" || pass != "Admin" {
10         fmt.Scan(&user, &pass)
11
12         if user != "Admin" || pass != "Admin" {
13             gagal++ // sama aja kaya (gagal=gagal+1)
14         }
15     }
16
17     fmt.Println(gagal, "percobaan gagal login")
18 }
19

```

PROBLEMS 10 OUTPUT DEBUG CONSOLE TERMINAL PORTS Code + - [] [X] ... [] [X] [X]

```

PS C:\MATA KULIAH\Algoritma dan Pemrograman\modul12> go run "c:\MATA KULIAH\Algoritma dan
Pemrograman\modul12\soal1.go"
User123 user123
User admin
Admin admin
Admin Admin123
Admin Admin
4 percobaan gagal login
PS C:\MATA KULIAH\Algoritma dan Pemrograman\modul12> go run "c:\MATA KULIAH\Algoritma dan
Pemrograman\modul12\soal1.go"
Admin Admin
0 percobaan gagal login
PS C:\MATA KULIAH\Algoritma dan Pemrograman\modul12>

```

Deskripsi program

Code di atas membuat sebuah sistem login sederhana yang terus meminta input username dan password hingga pengguna memasukkan pasangan yang benar, yaitu "Admin" untuk keduanya. Variabel gagal digunakan untuk menghitung berapa kali percobaan login yang salah dilakukan. Perulangan `for user != "Admin" || pass != "Admin"` akan terus berjalan selama username atau password tidak sesuai. Di dalam perulangan, program membaca dua input (user dan pass) menggunakan `fmt.Scan`, lalu jika salah satu tidak cocok dengan nilai yang ditetapkan, variabel gagal ditambah satu. Setelah pengguna akhirnya memasukkan username dan password yang benar, loop berhenti dan program menampilkan jumlah total percobaan login yang gagal sebelum login berhasil.

2. Tugas 2

Source code

```
package main

import "fmt"

func main() {
    var bil int

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

    for bil > 0 {
        digit := bil % 10
        fmt.Println(digit)
        bil = bil / 10
    }
}
```

Screenshoot program

The screenshot shows a Go program in a code editor and its execution output in a terminal. The program is designed to reverse a positive integer by printing its digits from right to left. It uses a loop that repeatedly takes the remainder of the number divided by 10 (to get the last digit) and then divides the number by 10 (to remove the last digit) until the number becomes zero.

```
soal2.go > ...
1  package main
2
3  import "fmt"
4
5  func main() {
6      var bil int
7
8      fmt.Print("Masukkan bilangan bulat positif: ")
9      fmt.Scan(&bil)
10
11     for bil > 0 {
12         digit := bil % 10
13         fmt.Println(digit)
14         bil = bil / 10
15     }
16 }
17
```

The terminal output shows the program being run from a command prompt. It prompts the user to enter a positive integer. Two examples are shown: 2544 and 3423554654. The output for 2544 is 4, 4, 5, 2. The output for 3423554654 is 4, 5, 6, 5, 4, 3, 2, 5, 4, 3.

```
PS C:\MATA KULIAH\Algoritma dan Pemrograman\modul12> go run "c:\MATA KULIAH\Algoritma dan
Pemrograman\modul12\soal2.go"
Masukkan bilangan bulat positif: 2544
4
4
5
2
PS C:\MATA KULIAH\Algoritma dan Pemrograman\modul12> go run "c:\MATA KULIAH\Algoritma dan
Pemrograman\modul12\soal2.go"
Masukkan bilangan bulat positif: 3423554654
4
5
6
5
4
3
2
5
4
3
PS C:\MATA KULIAH\Algoritma dan Pemrograman\modul12>
```

Deskripsi program

menampilkan setiap digit dari sebuah bilangan bulat positif secara terbalik (dari digit paling belakang ke paling depan). Program diawali dengan meminta pengguna memasukkan sebuah bilangan, lalu nilai tersebut disimpan dalam variabel bil. Selama nilai bil masih lebih besar dari nol, program akan mengambil digit terakhir dengan operasi $\text{bil} \% 10$, kemudian mencetak digit tersebut. Setelah itu nilai bil diperkecil dengan membaginya menggunakan $\text{bil} / 10$, sehingga digit terakhir terhapus. Proses ini berulang hingga seluruh digit habis dan program selesai.

3. Tugas 3

Source code

```
package main

import "fmt"

func main() {
    var y, sisa int

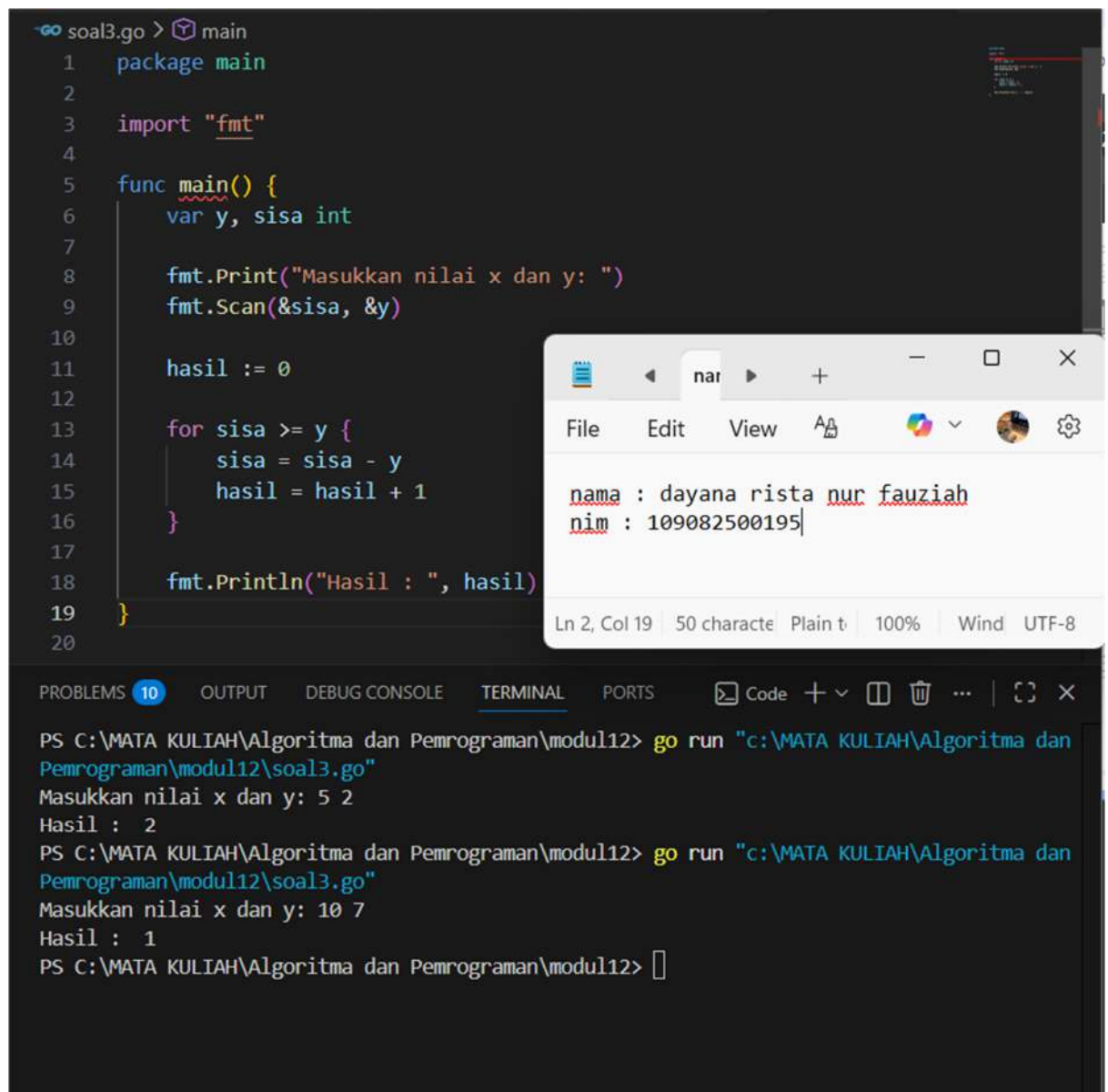
    fmt.Print("Masukkan nilai x dan y: ")
    fmt.Scan(&sisa, &y)

    hasil := 0

    for sisa >= y {
        sisa = sisa - y
        hasil = hasil + 1
    }

    fmt.Println("Hasil : ", hasil)
}
```

Screenshoot program



The screenshot shows a Go program in a code editor and its execution in a terminal. The code is a simple division program that takes two integers, x and y, and calculates the integer division result. It uses a loop to subtract y from x until x is less than y, counting the number of subtractions.

```
soal3.go > main
1 package main
2
3 import "fmt"
4
5 func main() {
6     var y, sisa int
7
8     fmt.Print("Masukkan nilai x dan y: ")
9     fmt.Scan(&sisa, &y)
10
11     hasil := 0
12
13     for sisa >= y {
14         sisa = sisa - y
15         hasil = hasil + 1
16     }
17
18     fmt.Println("Hasil : ", hasil)
19 }
20
```

The terminal shows the program being run twice. In the first run, the user enters 5 and 2, and the output is 2. In the second run, the user enters 10 and 7, and the output is 1.

```
PS C:\MATA KULIAH\Algoritma dan Pemrograman\modul12> go run "c:\MATA KULIAH\Algoritma dan Pemrograman\modul12\soal3.go"
Masukkan nilai x dan y: 5 2
Hasil : 2
PS C:\MATA KULIAH\Algoritma dan Pemrograman\modul12> go run "c:\MATA KULIAH\Algoritma dan Pemrograman\modul12\soal3.go"
Masukkan nilai x dan y: 10 7
Hasil : 1
PS C:\MATA KULIAH\Algoritma dan Pemrograman\modul12>
```

Deskripsi program

Setelah meminta dua input (bilangan yang dibagi disimpan di sisa, dan pembagi disimpan di y), program akan menjalankan perulangan for yang terus-menerus mengurangi y dari sisa, dan setiap pengurangan yang berhasil akan menambah nilai penghitung (hasil) sebanyak satu. Perulangan akan berhenti tepat ketika sisa menjadi lebih kecil dari y, dan program kemudian mencetak nilai akhir dari hasil, yang merupakan hasil pembagian bilangan bulat dari dua angka yang dimasukkan pengguna.

