

LAPORAN PRAKTIKUM
Algoritma Pemrograman

MODUL 13
REPEAT-UNTIL



Disusun oleh:
NUR FITRI RACHMADILLA DEWANTI
109082500057
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 kata string

    var n int

    fmt.Scan(&kata, &n)


    i := 0

    for kondisi := false; !kondisi; {

        fmt.Println(kata)

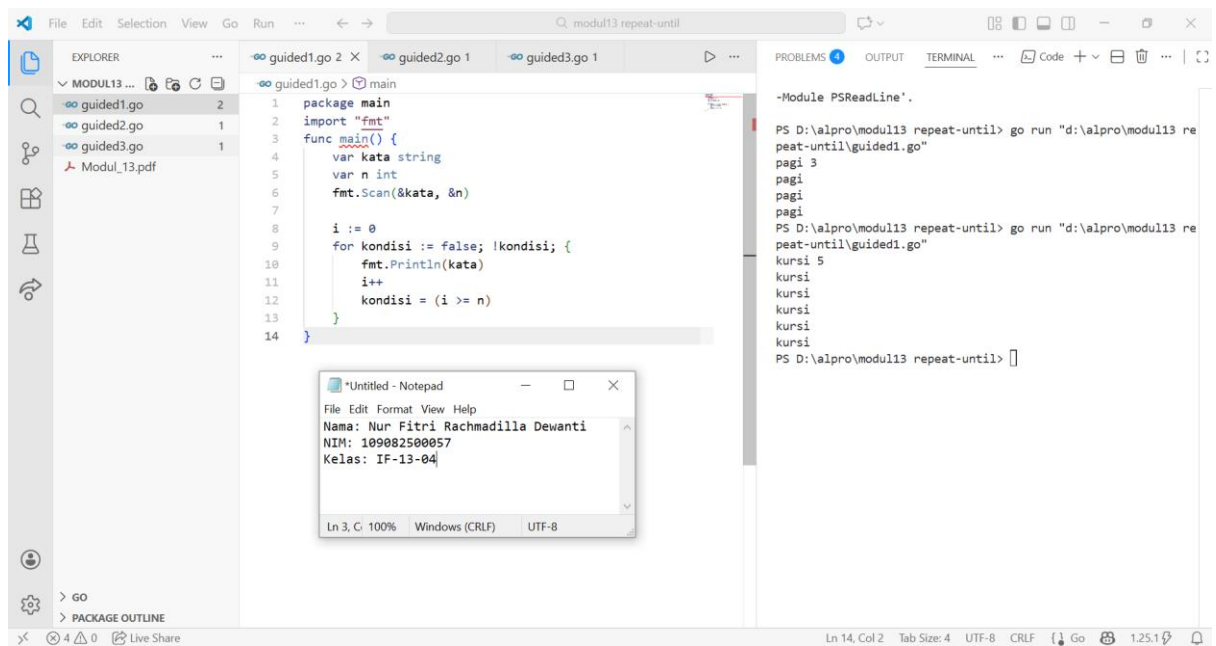
        i++

        kondisi = (i >= n)

    }

}
```

Screenshoot program



Deskripsi program

Program ini dibuat untuk menampilkan sebuah kata sebanyak n kali dengan jumlah n yang di inputkan. Variabel i digunakan sebagai penghitung jumlah perulangannya. Setiap perulangan, program akan menampilkan kata. Nilai i bertambah setiap perulangannya dan perulangan akan berhenti ketika jumlah output sudah mencapai n.

2. Guided 2

Source Code

```
package main

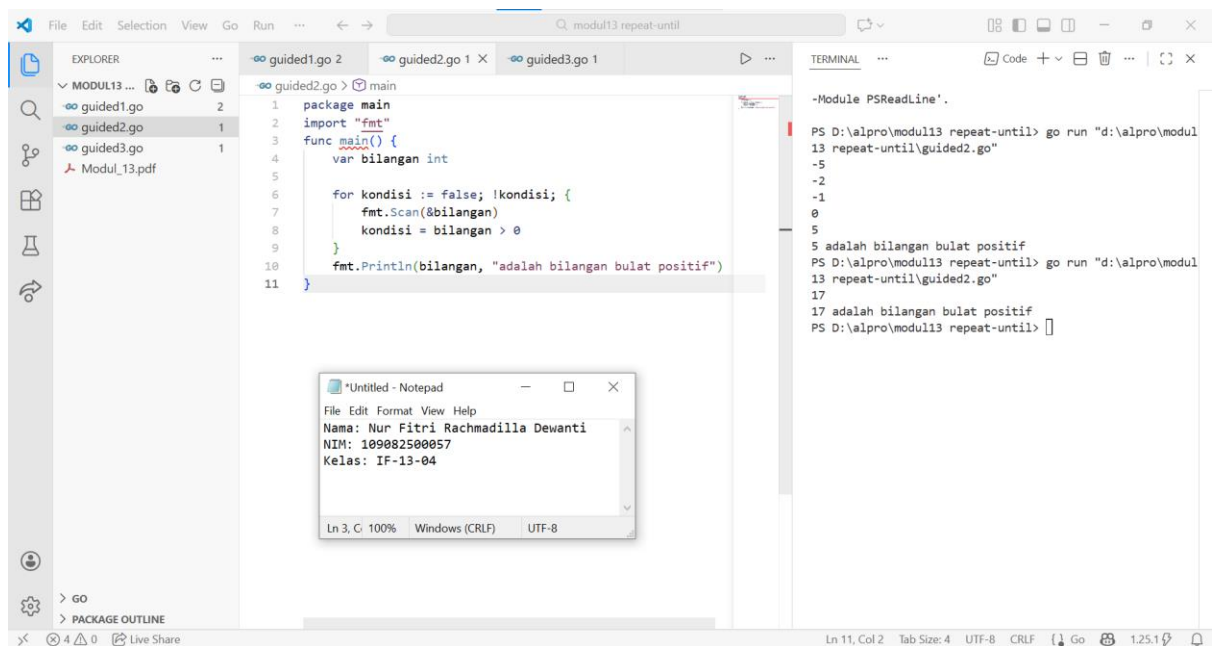
import "fmt"

func main() {
    var bilangan int

    for kondisi := false; !kondisi; {
        fmt.Scan(&bilangan)
        kondisi = bilangan > 0
    }

    fmt.Println(bilangan, "adalah bilangan bulat positif")
}
```

Screenshoot program



Deskripsi program

Program ini digunakan untuk menerima input bilangan bulat hingga input berupa bilangan bulat. Program akan terus meminta input selama bilangan yang di input belum bernilai positif. Setiap input bilangan akan di cek apakah bilangan tersebut lebih besar dari 0. Setelah perulangan selesai, program akan menampilkan output jika bilangan tersebut adalah bilangan bulat positif.

3. Guided 3

Source Code

```
package main

import "fmt"

func main() {

    var x, y int

    fmt.Scan(&x, &y)

    hasil := x

    for kondisi := false; !kondisi; {

        hasil = hasil - y

        fmt.Println(hasil)
```

```

        kondisi = hasil <= 0
    }

    fmt.Println(hasil == 0)
}

```

Screenshoot program

The screenshot shows a Go IDE with the following code in `guided3.go`:

```

1 package main
2 import "fmt"
3 func main() {
4     var x, y int
5     fmt.Scan(&x, &y)
6     hasil := x
7
8     for kondisi := false; !kondisi; {
9         hasil = hasil - y
10        fmt.Println(hasil)
11        kondisi = hasil <= 0
12    }
13    fmt.Println(hasil == 0)
14 }

```

The terminal output shows the program running and prompting for input:

```

-Module PSReadLine'.
PS D:\alpro\modul13 repeat-until> go run "d:\alpro\modul13 repeat-until\guided3.go"
5 2
3
1
-1
false
PS D:\alpro\modul13 repeat-until> go run "d:\alpro\modul13 repeat-until\guided3.go"
15 3
12
9
6
3
0
true
PS D:\alpro\modul13 repeat-until> go run "d:\alpro\modul13 repeat-until\guided3.go"
25 5
20
15
10
5
0
true
PS D:\alpro\modul13 repeat-until>

```

A small Notepad window is also visible, containing the following text:

```

*Untitled - Notepad
File Edit Format View Help
Nama: Nur Fitri Rachmadilla Dewanti
NIM: 109082500057
Kelas: IF-13-04

```

Deskripsi program

Program ini dibuat untuk mengurangi nilai x dan y secara berulang sampai hasil pengurangan bernilai 0 atau kurang dari sama dengan 0. Program juga menampilkan hasil akhir apakah hasil tersebut sama dengan 0 atau tidak. Perulangan akan berhenti jika hasil kurang dari atau sama dengan 0. Program akan menampilkan hasil "true" jika hasil akhir sama dengan 0, dan false jika tidak sama dengan 0.

TUGAS

1. Tugas 1

Source code

```
package main

import "fmt"

func main() {

    var bilangan, digit int

    fmt.Scan(&bilangan)

    for kondisi := false; !kondisi; {

        bilangan = bilangan/10

        digit++

        if bilangan == 0{

            kondisi = true

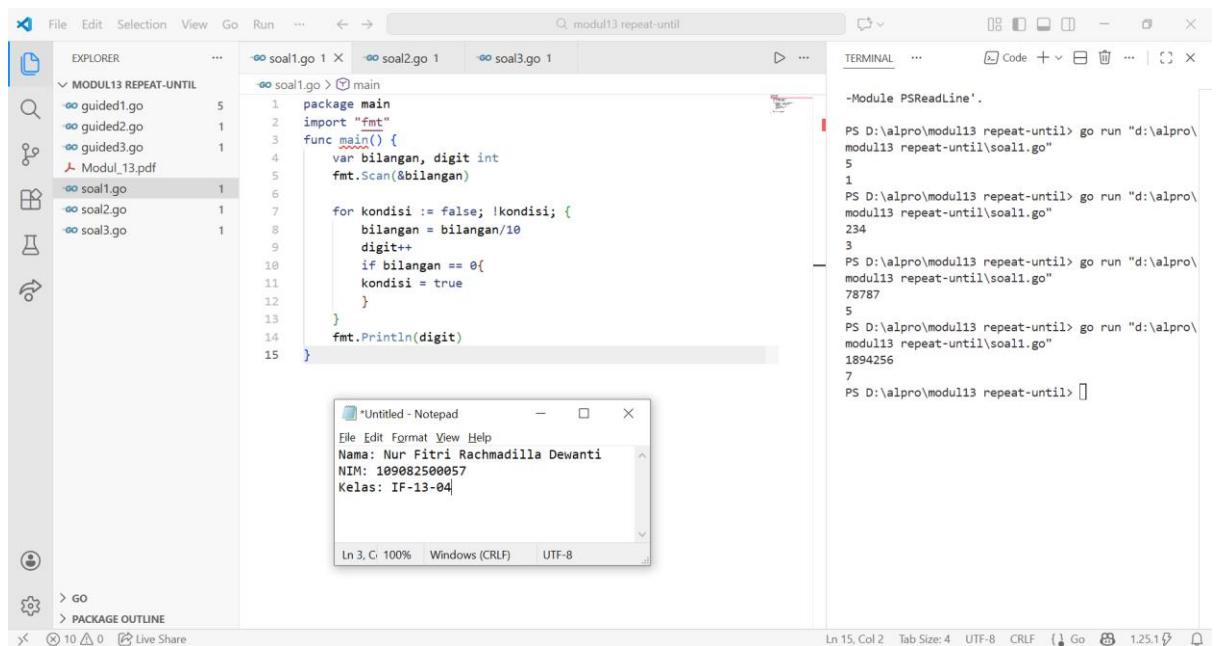
        }

    }

    fmt.Println(digit)

}
```

Screenshoot program



Deskripsi program

Program ini dibuat untuk menghitung jumlah digit dari sebuah bilangan bulat yang diinputkan. Program akan membagi bilangan dengan 10 secara berulang untuk menghilangkan satu digit di setiap perulangan. Proses perulangan akan berhenti ketika bilangan sudah bernilai 0. Hasil akhir yang akan di tampilkan adalah jumlah digit dari bilangan yang di inputkan.

2. Tugas 2

Source code

```
package main

import "fmt"

func main() {

    var bilangan float64

    var hasil int

    fmt.Scan(&bilangan)


    hasil = int(bilangan)

    if bilangan > float64(hasil) {

        hasil++

    }


    for kondisi := false; !kondisi; {

        bilangan = bilangan + 0.1

        fmt.Printf("%.1f\n", bilangan)


        if bilangan+0.000001 >= float64(hasil) {

            kondisi = true

        }

    }
```

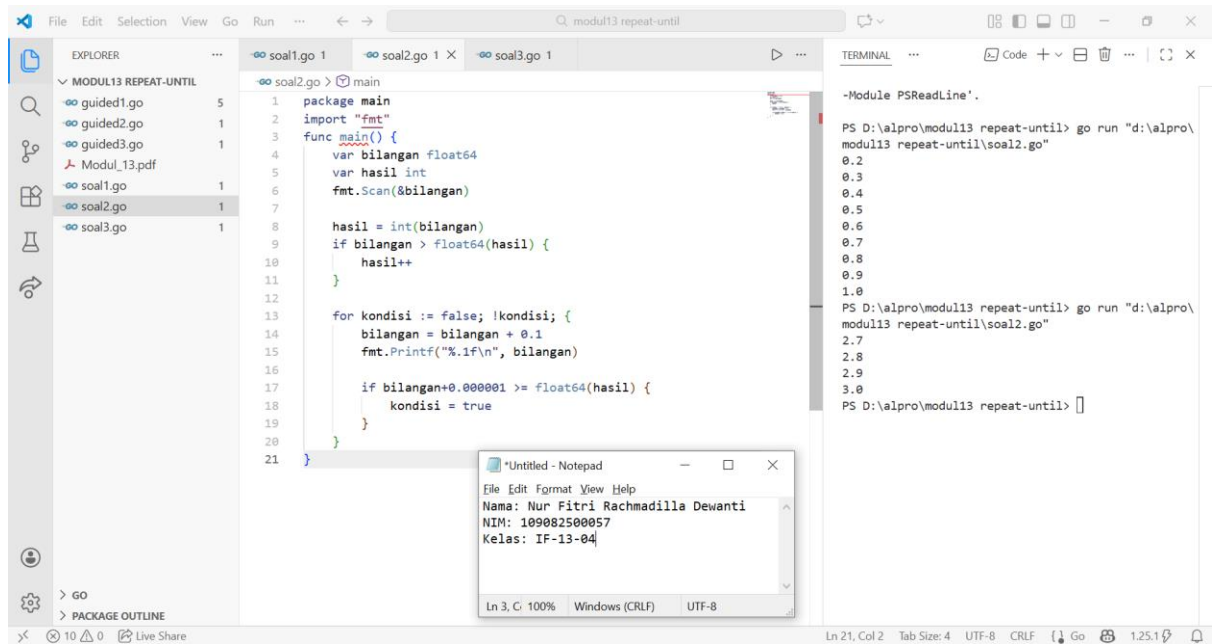
```

    }

}

```

Screenshoot program



Deskripsi program

Program ini dibuat untuk menampilkan deretan bilangan desimal yang bertambah 0.1 setiap perulangan, dimulai dari bilangan desimal yang di inputkan hingga mencapai pembulatan ke atas hingga bilangan tersebut. Variabel "hasil" digunakan untuk batas perulangan, yaitu hasil pembulatan ke atas dari bilangan yang di input.

```
hasil = int(bilangan)
```

```

    if bilangan > float64(hasil) {

        hasil++

    }

```

Kode ini digunakan untuk melakukan pembulatan keatas, Jika bilangan masih memiliki nilai desimal, maka nilai akhir ditambah 1.

```
bilangan = bilangan + 0.1
```

```
fmt.Printf("%.1f\n", bilangan)
```

bilangan di tambah 0.1 setiap perulangannya, fmt.Printf digunakan untuk menampilkan bilangan dengan satu di belakang koma.

```
if bilangan+0.000001 >= float64(hasil) {
```



```
        kondisi = true
    }
}
```

Kode ini digunakan untuk menghentikan perulangan jika nilai bilangan sudah mencapai atau mendekati nilai pembulatan ke atas. Penambahan "0.000001" berfungsi untuk menghindari kesalahan perhitungan bilangan pecahan.

3. Tugas 3

Source code

```
package main
import "fmt"
func main() {
    var target, donasi, total, donatur int
    fmt.Scan(&target)

    for kondisi := false; !kondisi; {
        fmt.Scan(&donasi)
        donatur++
        total = total + donasi

        fmt.Println("Donatur", donatur, ": Menyumbang",
donasi,
        ". Total terkumpul:", total,
        )

        if total >= target {
            kondisi = true
        }
    }
    fmt.Println("Target tercapai! Total donasi:",
total, "dari",
        donatur, "donatur.",
        )
}
```

Screenshoot program

The screenshot shows a Go program in VS Code. The Explorer pane on the left shows a project named 'MODUL13 REPEAT-UNTIL' with files 'guided1.go', 'guided2.go', 'guided3.go', 'Modul_13.pdf', 'soal1.go', 'soal2.go', and 'soal3.go'. The main editor shows 'soal3.go' with the following code:

```
1 package main
2 import "fmt"
3 func main() {
4     var target, donasi, total, donatur int
5     fmt.Scan(&target)
6
7     for kondisi := false; !kondisi; {
8         fmt.Scan(&donasi)
9         donatur++
10        total = total + donasi
11
12        fmt.Println("Donatur", donatur, ": Menyumbang", donasi,
13            ". Total terkumpul:", total,
14        )
15
16        if total >= target {
17            kondisi = true
18        }
19    }
20    fmt.Println("Target tercapai! Total donasi:", total, "dari",
21        donatur, "donatur.",
22    )
23 }
```

The terminal on the right shows the execution of the program. It prompts for a target value (300) and then for donator values (100, 50, 200, 300). The output shows the running total and the number of donors. After the 4th donor, the target is reached, and the program prints the final total and number of donors.

```
-Module PSReadLine'.
PS D:\alpro\modul13 repeat-until> go run "d:\alpro\modul13 repeat-until\soal3.go"
300
100
Donatur 1 : Menyumbang 100 . Total terkumpul: 100
50
Donatur 2 : Menyumbang 50 . Total terkumpul: 150
200
Donatur 3 : Menyumbang 200 . Total terkumpul: 350
Target tercapai! Total donasi: 350 dari 3 donatur.
PS D:\alpro\modul13 repeat-until> go run "d:\alpro\modul13 repeat-until\soal3.go"
500
150
Donatur 1 : Menyumbang 150 . Total terkumpul: 150
100
Donatur 2 : Menyumbang 100 . Total terkumpul: 250
50
Donatur 3 : Menyumbang 50 . Total terkumpul: 300
300
Donatur 4 : Menyumbang 300 . Total terkumpul: 600
Target tercapai! Total donasi: 600 dari 4 donatur.
PS D:\alpro\modul13 repeat-until> go run "d:\alpro\modul13 repeat-until\soal3.go"
200
300
Donatur 1 : Menyumbang 300 . Total terkumpul: 300
Target tercapai! Total donasi: 300 dari 1 donatur.
PS D:\alpro\modul13 repeat-until> 
```

Deskripsi program

Program ini digunakan untuk mencatat dan menjumlahkan donasi dari beberapa donatur. Program terus menerima input dari setiap donatur hingga total donasi mencapai atau melebihi target yang di inputkan pada inputan awal. Setiap donasi di inputkan, program akan menampilkan informasi donatur seberapa, jumlah donasi, serta total donasi yang sudah terkumpul. Setelah mencapai target, program akan menampilkan total donasi akhir dan jumlah donatur.

```
if total >= target {
    kondisi = true
}
```

Digunakan untuk mengecek total donasi telah mencapai atau melebihi target. Jika kondisi terpenuhi maka perulangan akan berhenti.