

LAPORAN PRAKTIKUM
Algoritma Pemrograman

EVALUASI



Disusun oleh:

Muhammad Fachri Auravyano Saka

103112430180

S1IF-12-07

PROGRAM STUDI S1 INFORMATIKA
FAKULTAS INFORMATIKA
TELKOM UNIVERSITY PURWOKERTO
2024

SOAL

1. SOAL 1

Source Code

```
package main

import "fmt"

func main() {
    var x, y int
    fmt.Println("Masukkan X: ")
    fmt.Scanln(&x, &y)

    if x > y {
        fmt.Println("Nilai x harus lebih kecil atau sama dengan y")
        return
    }

    jumlah := (y-x+1)*(x + y) / 2
    fmt.Printf("Jumlah bilangan dari %d sampai %d adalah %d\n", x, y, jumlah)
}
```

Screenshoot program

The screenshot shows a Go IDE with a dark theme. The editor displays the source code for a program that calculates the sum of integers from x to y. The code is as follows:

```
1 package main
2
3 import "fmt"
4
5 func main() {
6     var x, y int
7     fmt.Println("Masukkan X: ")
8     fmt.Scanln(&x, &y)
9
10    if x > y {
11        fmt.Println("Nilai x harus lebih kecil atau sama dengan y")
12        return
13    }
14
15    jumlah := (y-x+1)*(x + y) / 2
16    fmt.Printf("Jumlah bilangan dari %d sampai %d adalah %d\n", x, y, jumlah)
17 }
```

The terminal output shows the program being run twice. In the first run, x is 4 and y is 5, resulting in a sum of 9. In the second run, x is 2 and y is 100, resulting in a sum of 5049.

```
PS E:\Alpro2> go run "e:\Alpro2\UTS\Soal1.go"
Masukkan X:
4 5
Jumlah bilangan dari 4 sampai 5 adalah 9
PS E:\Alpro2> go run "e:\Alpro2\UTS\Soal1.go"
Masukkan X:
2 100
Jumlah bilangan dari 2 sampai 100 adalah 5049
PS E:\Alpro2>
```

Deskripsi program

Code di atas bertujuan untuk mengetahui dan menghitung jumlah semua bilangan bulat dalam kondisi tertentu. kondisi ini ditentukan oleh dua angka yang dimasukkan oleh pengguna, yaitu x dan y.

Deklarasi Variabel: Program meminta pengguna untuk memasukkan dua nilai, yaitu x dan y, yang bertipe integer.

Input Data: Program menerima input dari pengguna untuk dua variabel tersebut menggunakan `fmt.Scanln(&x, &y)`. Ini memungkinkan pengguna untuk memasukkan kedua angka dalam satu baris.

Validasi: Setelah mendapatkan input, program memeriksa apakah nilai x lebih besar dari y. Jika x lebih besar, program akan menampilkan pesan kesalahan, "Nilai x harus lebih kecil atau sama dengan y," dan berhenti (return).

Menghitung Jumlah Bilangan: Jika nilai x lebih kecil atau sama dengan y, program menghitung jumlah bilangan dari x hingga y

2. SOAL 2

Source Code

```
package main

import "fmt"

func main() {
    var n int
    fmt.Println("Masukkan nilai n ")
    fmt.Scanln(&n)
    if n <= 0 {
        fmt.Println("Nilai n bilangan bulat positif")
        return
    }
    for i := 1; i <= n; i++ {
        fmt.Printf("%d ", i*i)
    }
    fmt.Println()
}
```

Screenshoot program

```
Go Run ... Alpro2
Soal1.go 2 Soal2.go 1 X Soal3.go 1
UTS > Soal2.go > main
1 package main
2
3 import "fmt"
4
5 func main() {
6     var n int
7     fmt.Println("Masukkan nilai n ")
8     fmt.Scanln(&n)
9     if n <= 0 {
10        fmt.Println("Nilai n bilangan bulat positif")
11        return
12    }
13    for i := 1; i <= n; i++ {
14        fmt.Printf("%d ", i*i)
15    }
16    fmt.Println()
17 }

TERMINAL PORTS
PS E:\Alpro2> go run "e:\Alpro2\UTS\Soal2.go"
Masukkan nilai n
4
1 4 9 16
PS E:\Alpro2> go run "e:\Alpro2\UTS\Soal2.go"
Masukkan nilai n
10
1 4 9 16 25 36 49 64 81 100
PS E:\Alpro2>
```

Deskripsi program

Program ini bertujuan untuk memasukkan sebuah bilangan bulat positif n dan menampilkan kuadrat dari setiap bilangan bulat mulai dari 1 hingga n . Berikut penjelasan lengkapnya:

Deklarasi Variabel: Program mendeklarasikan variabel integer n , yang akan menampung input dari pengguna.

Input Data: Program meminta pengguna untuk memasukkan nilai n dan menyimpannya dalam variabel n menggunakan `fmt.Scanln(&n)`.

Validasi: Program memeriksa apakah nilai n kurang dari atau sama dengan 0:

Jika n kurang dari atau sama dengan 0, program menampilkan pesan "Nilai n bilangan bulat positif" dan berhenti (`return`). Ini memastikan bahwa input adalah bilangan bulat positif.

Perulangan dan Perhitungan Kuadrat:

Jika nilai n valid (bilangan bulat positif), program memasuki perulangan `for` yang berjalan dari 1 hingga n .

Pada setiap iterasi, program mencetak hasil kuadrat dari angka tersebut ($i*i$) dengan `fmt.Printf("%d ", i*i)`.

3. SOAL 3

Source Code

```
package main

import "fmt"
```

```

func main() {
    var qirat int
    var sisa int

    fmt.Println("Masukkan jumlah qirat ")

    fmt.Scanln(&qirat)

    dinar := qirat / 600

    sisa = qirat % 600

    dirham := sisa / 60

    sisa = sisa % 60

    fals := sisa / 6

    sisa = sisa % 6

    fmt.Printf("%d %d %d %d\n", dinar, dirham, fals,
sisa)
}

```

Screenshoot program

The screenshot shows a Go IDE with the following components:

- Source Code (Soal3.go):**

```

1 package main
2
3 import "fmt"
4
5 func main() {
6     var qirat int
7     var sisa int
8     fmt.Println("Masukkan jumlah qirat ")
9     fmt.Scanln(&qirat)
10    dinar := qirat / 600
11    sisa = qirat % 600
12    dirham := sisa / 60
13    sisa = sisa % 60
14    fals := sisa / 6
15    sisa = sisa % 6
16    fmt.Printf("%d %d %d %d\n", dinar, dirham, fals, sisa)
17 }

```
- Terminal Output:**

```

PS E:\Alpro2> go run "e:\Alpro2\UTS\Soal3.go"
Masukkan jumlah qirat
1000
1 6 6 4
PS E:\Alpro2> go run "e:\Alpro2\UTS\Soal3.go"
Masukkan jumlah qirat
679
1 1 3 1
PS E:\Alpro2> go run "e:\Alpro2\UTS\Soal3.go"
Masukkan jumlah qirat
100
0 1 6 4
PS E:\Alpro2>

```
- IDE Interface:** The IDE shows the file explorer with 'Soal1.go 2', 'Soal2.go 1', and 'Soal3.go 1'. The terminal is at the bottom, and the code editor is in the center.

Deskripsi program

Program ini bertujuan untuk mengonversi jumlah satuan mata uang qirat menjadi beberapa pecahan mata uang yang lebih besar pada masa kekhalifahan Harun Al Rasyid, yaitu dinar, dirham, fals, dan sisa qirat. Konversi ini didasarkan pada aturan:

1 dinar = 600 qirat

1 dirham = 60 qirat

1 fals = 6 qirat

Program ini membantu menghitung berapa banyak dinar, dirham, fals, dan sisa qirat dari total jumlah qirat yang dimasukkan. Konversi ini memudahkan untuk memahami jumlah uang dalam satuan pecahan yang lebih besar, yang mungkin lebih mudah dipahami atau digunakan pada saat itu.