

**LAPORAN PRAKTIKUM
ALGORITMA DAN PEMROGRAMAN 2**

**MODUL VI
REKURSIF**



Disusun Oleh :

AFRIZAL DWI NUGRAHA / 2311102136

S1 IF 11 05

Dosen Pengampu :

Arif Amrulloh, S.Kom., M.Kom.

PROGRAM STUDI S1 TEKNIK INFORMATIKA

FAKULTAS INFORMATIKA

TELKOM UNIVERSITY PURWOKERTO

2024

I. DASAR TEORI

Rekursif adalah sebuah teknik di pemrograman, di mana sebuah fungsi memanggil dirinya sendiri. Mirip dengan loop (pengulangan), yang kedua tujuannya memanggil aksi yang sama berkali-kali.

Namun rekursi sering digunakan untuk menyelesaikan masalah yang lebih kompleks :

- yang tidak bisa (sulit) diselesaikan dengan loop biasa
- atau kode implementasinya akan sangat sulit dibaca jika menggunakan loop (iterasi)

Contoh sederhana fungsi rekursif yang memanggil dirinya sendiri

```
function show(image) {  
  
    show(image); // <- memanggil dirinya sendiri "show"  
  
}
```

Komponen Rekursif

Algoritma rekursif terdiri dari dua komponen utama:

1. Base-case (Basis), yaitu bagian untuk menghentikan proses rekursif dan komponen menjadi penting di dalam sebuah rekursif.
2. Recursive-case, yaitu bagian pemanggilan subprogramnya

II. GUIDED

1. Program Sederhana untuk Membaca dan Menampilkan

Sourcecode

```
package main  
  
import (  
    "fmt"  
)  
  
func main() {
```

```

var n int
fmt.Print("Masukan Nilai :")
fmt.Scan(&n)
baris(n)
}
func baris(bilangan int) {
    if bilangan == 1 {
        fmt.Println(1)
    } else {
        fmt.Println(bilangan)
        baris(bilangan - 1)
    }
}

func main() {
    fmt.Println("Afri zal Dwi Nugraha")
}

```

Screenshoot Output

```

PROBLEMS 7 OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\Users\PC BRANDED\OneDrive\ドキュメント\Afri zal VS\new\Semester03> go run
temuan6\guided01.go
Masukan Nilai : 4
4
3
2
1
PS C:\Users\PC BRANDED\OneDrive\ドキュメント\Afri zal VS\new\Semester03>

```

Deskripsi Program

Dalam program ini, pengguna diminta untuk memasukkan bilangan bulat positif n dan kemudian mencetak bilangan tersebut secara berurutan dari n hingga 1. Untuk menampilkan setiap bilangan pada satu baris, program menggunakan fungsi rekursif yang disebut "baris". Pada setiap pemanggilan rekursif, bilangan dikurangi 1 hingga mencapai bilangan 1 sebagai kondisi dasar (base case), dan rekursi berhenti saat bilangan mencapai atau kurang dari 0.

2. Program Rekursif untuk Menghitung Penjumlahan Bilangan dari 1 hingga n

Sourcecode

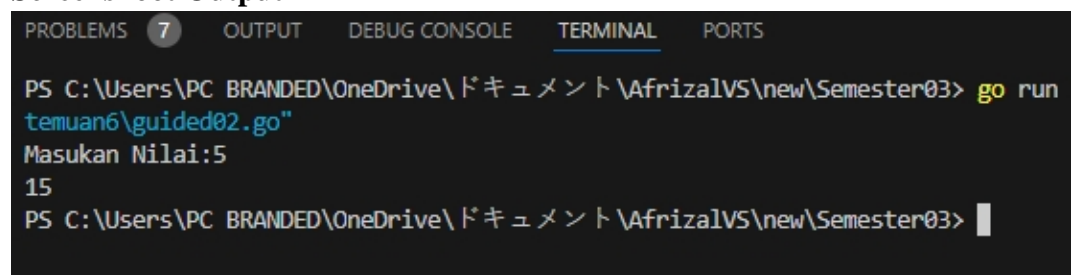
```
package main

import (
    "fmt"
)

func penjumlahan(n int) int {
    if n == 1 {
        return 1
    } else {
        return n + penjumlahan(n-1)
    }
}

func main() {
    var n int
    fmt.Print("Masukan Nilai : ")
    fmt.Scan(&n)
    fmt.Println(penjumlahan(n))
}
```

Screenshoot Output



```
PS C:\Users\PC BRANDED\OneDrive\ドキュメント\Afriza\VS\new\Semester03> go run
temuan6\guided02.go
Masukan Nilai:5
15
PS C:\Users\PC BRANDED\OneDrive\ドキュメント\Afriza\VS\new\Semester03>
```

Deskripsi Program

Program ini menggunakan rekursi untuk menghitung jumlah total dari 1 hingga n. Jika pengguna memasukkan bilangan n, program akan menggunakan fungsi penjumlahan untuk secara rekursif menambahkan semua bilangan bulat dari 1 hingga nn. Proses ini memanggil dirinya sendiri sampai kondisi dasar di mana nilai n sama dengan 1.

3. Program Rekursif untuk Menghitung 2 Pangkat n

Sourcecode

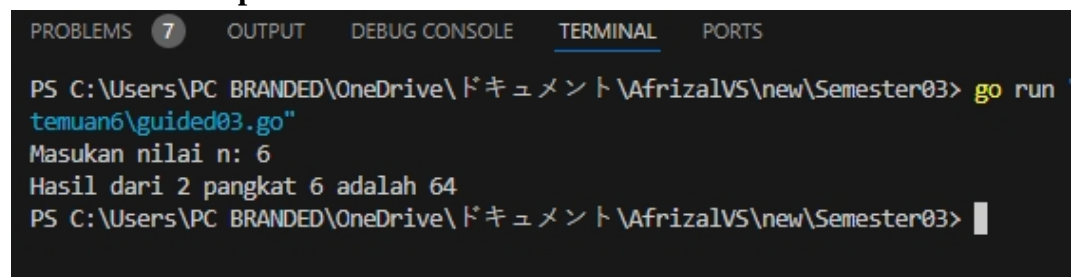
```
package main

import (
    "fmt"
)

// fungsi rekursif untuk menghitung 2^n
func pangkat(n int) int {
    if n == 0 {
        return 1
    } else {
        return 2 * pangkat(n-1)
    }
}

func main() {
    var n int
    fmt.Print("Masukan nilai n:")
    fmt.Scan(&n)
    fmt.Println("Hasil dari 2 pangkat", n, "adalah", pangkat(n))
}
```

Screenshoot Output



```
PROBLEMS 7 OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\Users\PC BRANDED\OneDrive\ドキュメント\AfriZalVS\new\Semester03> go run
temuan6\guided03.go
Masukan nilai n: 6
Hasil dari 2 pangkat 6 adalah 64
PS C:\Users\PC BRANDED\OneDrive\ドキュメント\AfriZalVS\new\Semester03> |
```

Deskripsi Program

Program ini menggunakan rekursi untuk menghitung hasil dari 2 n (dua pangkat n). Pengguna diminta untuk memasukkan bilangan bulat non-negatif nn , dan program akan menggunakan fungsi rekursif pangkat untuk menghitung hasil dari 2 yang dipangkatkan dengan n Fungsi rekursif ini bekerja dengan mengalikan 2 dengan hasil panggilan fungsi pangkat(n-1), hingga mencapai kondisi dasar ketika n = 0.

4. Program Rekursif untuk Menghitung Faktorial dari Bilangan n

Sourcecode

```
package main

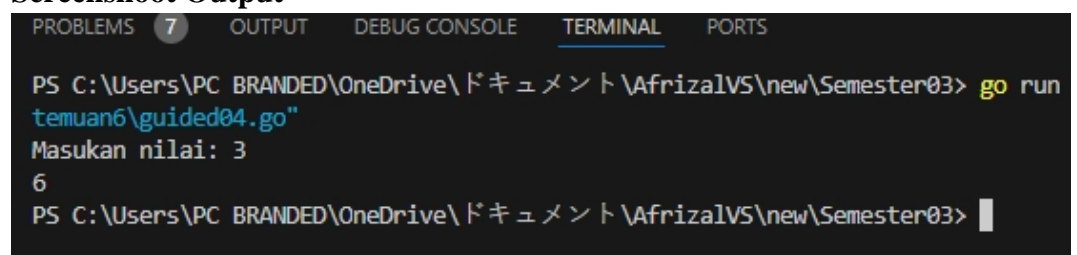
import (
    "fmt"
)

var n int

func faktorial(n int) int {
    if n == 0 || n == 1 {
        return 1
    } else {
        return n * faktorial(n-1)
    }
}

func main() {
    fmt.Print("Masukan nilai: ")
    fmt.Scan(&n)
    fmt.Println(faktorial(n))
}
```

Screenshoot Output



```
PS C:\Users\PC BRANDED\OneDrive\ドキュメント\AfrizalVS\new\Semester03> go run
temuan6\guided04.go
Masukan nilai: 3
6
PS C:\Users\PC BRANDED\OneDrive\ドキュメント\AfrizalVS\new\Semester03>
```

Deskripsi Program

Program ini menerima input pengguna dan menggunakan fungsi rekursif faktorial untuk menghitung nilai faktorial dari bilangan bulat positif n. Nilai faktorial dari bilangan n, yang dianggap sebagai n!, adalah hasil perkalian semua bilangan bulat positif dari 1 hingga n.

III. UNGUIDED

1. Deret Fibonacci adalah sebuah deret dengan nilai suku ke-0 dan ke-1 adalah 0 dan 1, dan nilai suku ke-n selanjutnya adalah hasil penjumlahan dua suku sebelumnya. Secara umum dapat diformulasikan $S_n = S_{n-1} + S_{n-2}$. Berikut ini adalah contoh nilai deret Fibonacci hingga suku ke-10. Buatlah program yang mengimplementasikan fungsi rekursif pada deret Fibonacci tersebut.

n	0	1	2	3	4	5	6	7	8	9	10
S_n	0	1	1	2	3	5	8	13	21	34	55

Sourcecode

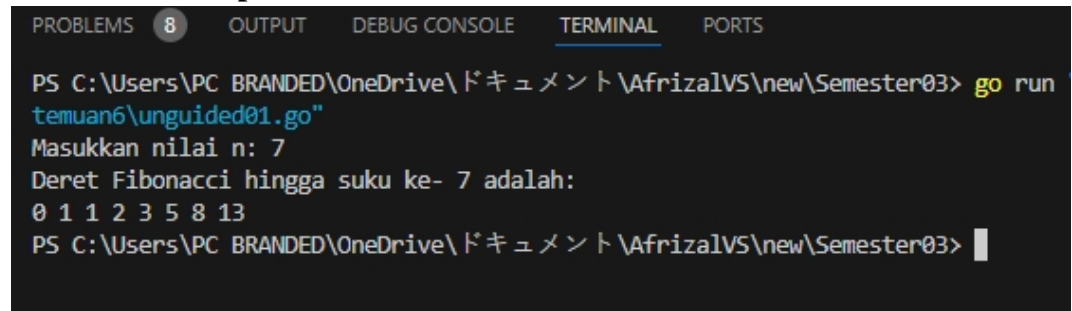
```
package main

import (
    "fmt"
)

// Fungsi rekursif untuk menghitung Fibonacci
func fibonacci(n int) int {
    if n == 0 {
        return 0
    } else if n == 1 {
        return 1
    } else {
        return fibonacci(n-1) + fibonacci(n-2)
    }
}

func main() {
    var n int
    fmt.Print("Masukkan nilai n: ")
    fmt.Scan(&n)
    fmt.Println("Deret Fibonacci hingga suku ke-", n,
"adalah:")
    for i := 0; i <= n; i++ {
        fmt.Print(fibonacci(i), " ")
    }
    fmt.Println()
}
```

Screenshoot Output



```
PROBLEMS 8 OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\Users\PC BRANDED\OneDrive\ドキュメント\AfrizalVS\new\Semester03> go run
temuan6\unguided01.go
Masukkan nilai n: 7
Deret Fibonacci hingga suku ke- 7 adalah:
0 1 1 2 3 5 8 13
PS C:\Users\PC BRANDED\OneDrive\ドキュメント\AfrizalVS\new\Semester03> |
```

Deskripsi Program

Dengan menggunakan fungsi rekursif, program ini menghitung deret Fibonacci hingga suku ke- n . Urutan bilangan yang dikenal sebagai deret Fibonacci adalah:

Suku pertama (suku ke-0) memiliki nilai 0 dan suku kedua (suku ke-1) memiliki nilai 1. Setiap suku berikutnya adalah hasil penjumlahan dari dua suku sebelumnya, yaitu $S_n = S_{n-1} + S_{n-2}$ dan $S_n = S_{n-1} + S_{n-2}$.

Deret Fibonacci hingga suku ke-10 adalah sebagai berikut:

0,1,1,2,3,5,8,13,21,34,55

Buatlah sebuah program yang digunakan untuk menampilkan pola bintang berikut ini dengan menggunakan fungsi rekursif. N adalah masukan dari pengguna.

Contoh masukan dan keluaran:

No	Masukan	Keluaran
1	5	* ** *** **** *****
2	1	*
3	3	* ** ***

Sourcecode

```
package main

import (
    "fmt"
)

// Fungsi rekursif untuk mencetak baris bintang
func cetakBintang(baris int) {
    if baris == 0 {
        return
    }
    cetakBintang(baris - 1)
    for i := 0; i < baris; i++ {
```

```

        fmt.Print("*")
    }
    fmt.Println()
}

func main() {
    var n int
    fmt.Print("Masukkan nilai N: ")
    fmt.Scan(&n)
    cetakBintang(n)
}

```

Screenshoot Output

```

PROBLEMS 10 OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS C:\Users\PC BRANDED\OneDrive\ドキュメント\AfriZalVS\new\Semester03> go run
temuan6\unguided02.go
Masukkan nilai N: 4
*
**
***
****
PS C:\Users\PC BRANDED\OneDrive\ドキュメント\AfriZalVS\new\Semester03>

```

Deskripsi Program

Menggunakan fungsi rekursif, program ini dapat mencetak pola segitiga bintang dengan tinggi N. Setelah pengguna memasukkan bilangan bulat positif N, program akan mencetak pola bintang dengan jumlah baris yang sesuai dengan nilai N, dengan bintang sebanyak nomor baris tersebut..

3. Buatlah program yang mengimplementasikan rekursif untuk menampilkan faktor bilangan dari suatu N, atau bilangan yang apa saja yang habis bagi N.

Masukan terdiri dari suatu bilangan bulat positif N.

Keluaran terdiri dari bilangan yang menjadi faktor dari N (terurut dari 1 hingga N ya).

Contoh masukan dan keluaran:

No	Masukan	Keluaran
1	5	1 5
2	12	1 2 3 4 6 12

Sourcecode

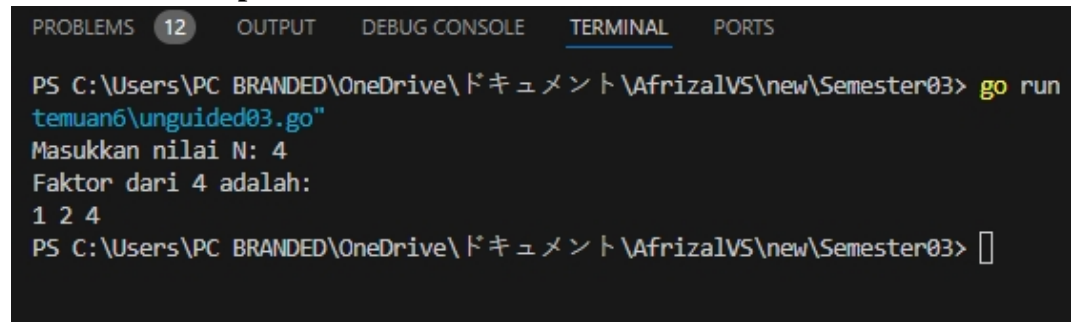
```
package main

import (
    "fmt"
)

// Fungsi rekursif untuk mencari faktor bilangan
func faktor(n int, i int) {
    if i > n {
        return
    }
    if n%i == 0 {
        fmt.Print(i, " ")
    }
    faktor(n, i+1)
}

func main() {
    var n int
    fmt.Print("Masukkan nilai N: ")
    fmt.Scan(&n)
    fmt.Println("Faktor dari ", n, " adalah:")
    faktor(n, 1)
    fmt.Println()
}
```

Screenshoot Output



```
PROBLEMS 12 OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\Users\PC BRANDED\OneDrive\ドキュメント\AfriZalVS\new\Semester03> go run
temuan6\unguided03.go
Masukkan nilai N: 4
Faktor dari 4 adalah:
1 2 4
PS C:\Users\PC BRANDED\OneDrive\ドキュメント\AfriZalVS\new\Semester03> 
```

Deskripsi Program

Program ini mencetak faktor-faktor dari suatu bilangan bulat positif yang dimasukkan oleh pengguna. Fungsi rekursif(n) menggunakan perulangan untuk memeriksa setiap bilangan dari 1 hingga n. Jika n habis dibagi oleh i (artinya $n \% i == 0$), maka i adalah faktor dari n dan dicetak. Di fungsi utama, program meminta pengguna memasukkan nilai a, lalu memanggil rekursif(a) untuk mencetak semua faktor dari nilai tersebut.

4. Buatlah program yang mengimplementasikan rekursif untuk menampilkan sejumlah bilangan tertentu.

Masukan terdiri dari sebuah bilangan bulat positif N.

Keluaran terdiri dari barisan bilangan dari N hingga 1 dan kembali ke N.

Contoh masukan dan keluaran:

No	Masukan	Keluaran
1	5	5 4 3 2 1 2 3 4 5
2	9	9 8 7 6 5 4 3 2 1 2 3 4 5 6 7 8 9

Sourcecode

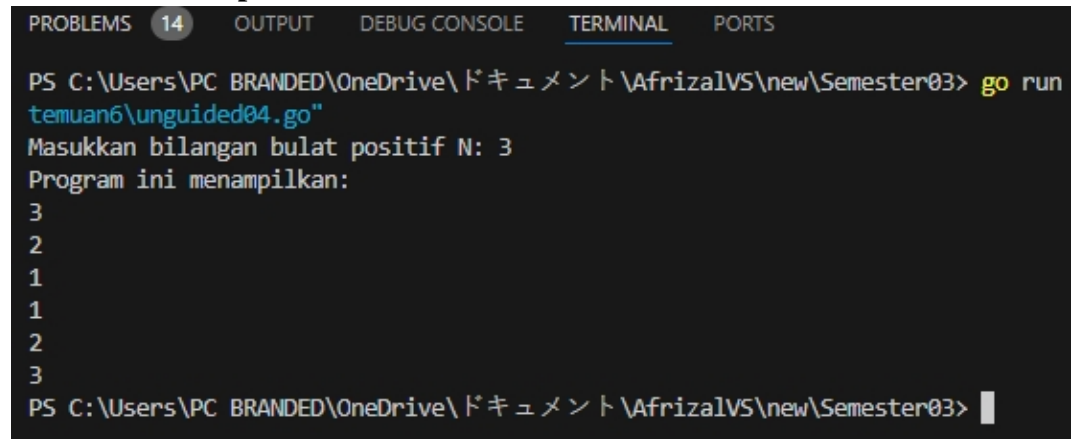
```
package main

import (
    "fmt"
)

// Fungsi rekursif untuk menampilkan angka dari N hingga 1
func displayNumbers(n int) {
    if n < 1 {
        return
    }
    fmt.Println(n)
    displayNumbers(n - 1)
    fmt.Println(n)
}

func main() {
    var N int
    fmt.Print("Masukkan bilangan bulat positif N: ")
    fmt.Scan(&N)
    fmt.Println("Program ini menampilkan:")
    displayNumbers(N)
}
```

Screenshoot Output



```
PROBLEMS 14 OUTPUT DEBUG CONSOLE TERMINAL PORTS

PS C:\Users\PC BRANDED\OneDrive\ドキュメント\AfriZalVS\new\Semester03> go run
temuan6\unguided04.go
Masukkan bilangan bulat positif N: 3
Program ini menampilkan:
3
2
1
1
2
3
PS C:\Users\PC BRANDED\OneDrive\ドキュメント\AfriZalVS\new\Semester03>
```

Deskripsi Program

Program ini bertujuan untuk menampilkan urutan bilangan bulat mulai dari N hingga 1, dan kemudian menggunakan teknik rekursif untuk kembali ke N.

5. Buatlah program yang mengimplementasikan rekursif untuk menampilkan barisan bilangan ganjil.
Masukan terdiri dari sebuah bilangan bulat positif N.
Keluaran terdiri dari barisan bilangan ganjil dari 1 hingga N.

Contoh masukan dan keluaran:

No	Masukan	Keluaran
1	5	1 3 5
2	20	1 3 5 7 9 11 13 15 17 19

Sourcecode

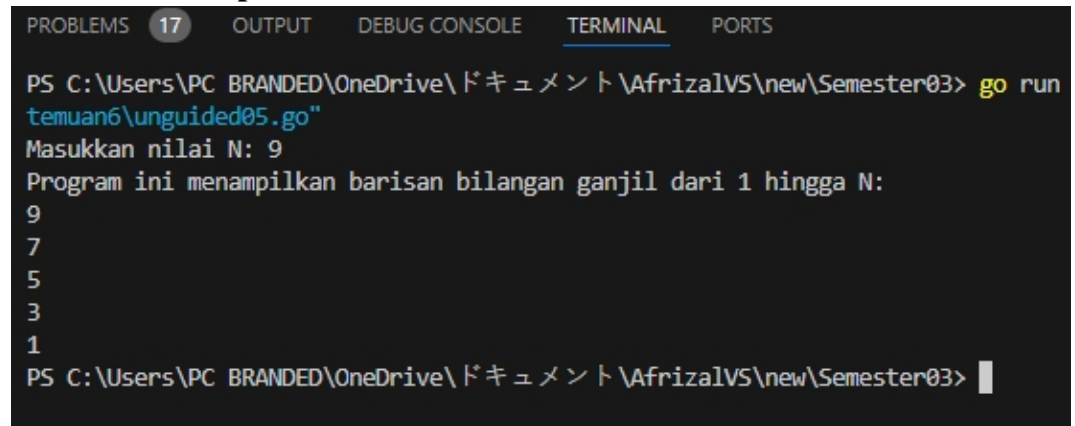
```
package main

import (
    "fmt"
)

// Fungsi rekursif untuk menampilkan bilangan ganjil dari 1 hingga N
func displayOddNumbers(n int) {
    if n < 1 {
        return
    }
    if n%2 != 0 {
        fmt.Println(n)
    }
    displayOddNumbers(n - 1)
}

func main() {
    var N int
    fmt.Print("Masukkan nilai N: ")
    fmt.Scan(&N)
    fmt.Println("Program ini menampilkan barisan bilangan ganjil dari 1 hingga N:")
    displayOddNumbers(N)
}
```

Screenshoot Output



```
PROBLEMS 17 OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\Users\PC BRANDED\OneDrive\ドキュメント\AfrizalVS\new\Semester03> go run
temuan6\unguided05.go
Masukkan nilai N: 9
Program ini menampilkan barisan bilangan ganjil dari 1 hingga N:
9
7
5
3
1
PS C:\Users\PC BRANDED\OneDrive\ドキュメント\AfrizalVS\new\Semester03>
```

Deskripsi Program

Tujuan program adalah untuk menampilkan semua bilangan ganjil dari 1 hingga N.

6. Buatlah program yang mengimplementasikan rekursif untuk mencari hasil pangkat dari dua buah bilangan.

Masukan terdiri dari bilangan bulat x dan y.

Keluaran terdiri dari hasil x yang dipangkatkan y.

Catatan: diperbolehkan menggunakan tanda bintang , tapi dilarang menggunakan import "math".

Contoh masukan dan keluaran:

No	Masukan	Keluaran
1	2 2	4
2	5 3	125

Sourcecode

```
package main

import (
    "fmt"
)

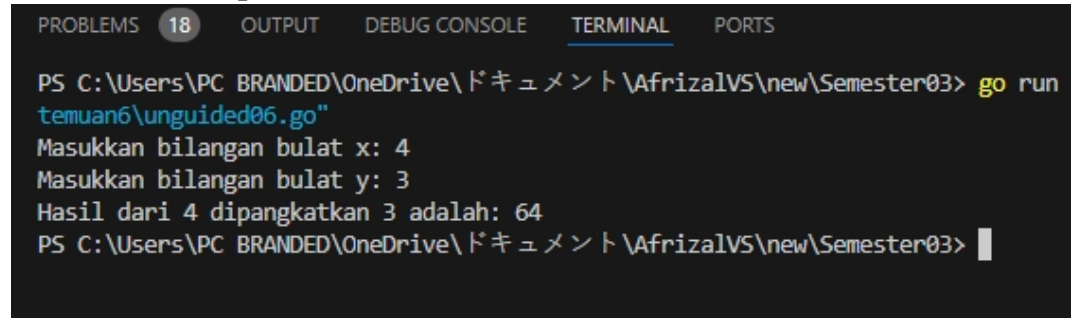
// Fungsi rekursif untuk menghitung x pangkat y
func power(x int, y int) int {
    if y == 0 {
        return 1
    }
    return x * power(x, y-1)
}

func main() {
    var x, y int
    fmt.Print("Masukkan bilangan bulat x: ")
    fmt.Scan(&x)
    fmt.Print("Masukkan bilangan bulat y: ")
    fmt.Scan(&y)

    // Memanggil fungsi power dan mencetak hasilnya
    result := power(x, y)
```

```
    fmt.Printf("Hasil dari %d dipangkatkan %d adalah: %d\n", x, y,
result)
}
```

Screenshoot Output

A screenshot of a terminal window from a code editor. The terminal shows the execution of a Go program. The prompt is 'PS C:\Users\PC BRANDED\OneDrive\ドキュメント\AfrizalVS\new\Semester03>'. The user enters 'go run temuan6\unguided06.go'. The program prompts for two integers: 'Masukkan bilangan bulat x: 4' and 'Masukkan bilangan bulat y: 3'. It then outputs 'Hasil dari 4 dipangkatkan 3 adalah: 64'. The prompt returns to 'PS C:\Users\PC BRANDED\OneDrive\ドキュメント\AfrizalVS\new\Semester03>'. The terminal window has tabs for 'PROBLEMS', 'OUTPUT', 'DEBUG CONSOLE', 'TERMINAL' (selected), and 'PORTS'. The 'PROBLEMS' tab shows 18 items.

```
PS C:\Users\PC BRANDED\OneDrive\ドキュメント\AfrizalVS\new\Semester03> go run
temuan6\unguided06.go
Masukkan bilangan bulat x: 4
Masukkan bilangan bulat y: 3
Hasil dari 4 dipangkatkan 3 adalah: 64
PS C:\Users\PC BRANDED\OneDrive\ドキュメント\AfrizalVS\new\Semester03>
```

Deskripsi Program

Tujuan program adalah untuk menghitung hasil dari x yang dipangkatkan dengan y dengan menggunakan fungsi rekursif.

Daftar Pustaka :

1. <https://sko.dev/wiki/rekursif>