

**LAPORAN PRAKTIKUM
ALGORITMA DAN PEMROGRAMAN 2
MODUL 5
MATERI REKURSIF**



Disusun Oleh:

MUHAMMAD GAMEL AL GHIFARI

NIM: 103112400028

KELAS: 12 IF 01

**PROGRAM STUDI S1 TEKNIK INFORMATIKA
FAKULTAS INFORMATIKA
TELKOM UNIVERSITY PURWOKERTO
2025/2026**

I. DASAR TEORI

Rekursif adalah suatu teknik dalam pemrograman di mana sebuah fungsi atau prosedur memanggil dirinya sendiri untuk menyelesaikan sub-masalah yang lebih kecil dari masalah utama. Teknik ini sangat berguna untuk memecahkan masalah yang memiliki struktur berulang atau dapat dibagi menjadi bagian-bagian yang lebih sederhana. Kondisi recursive case adalah komplemen atau negasi dari base case. Algoritma rekursif terdiri dari dua komponen yaitu base case (basis), dan yang kedua ada recursive case.

II. GUIDED' SOURCE CODE

```
package main

import "fmt"

func pangkatIteratif(base, exp int) int {
    hasil := 1

    for i := 0; i < exp; i++ {
        hasil *= base
    }

    return hasil
}

func faktorialIteratif(n int) int {
    hasil := 1
    for i := 2; i <= n; i++ {
        hasil *= i
    }
    return hasil
}

func main() {
    var base, exp, n int
    fmt.Print("Masukkan bilangan: ")
    fmt.Scanln(&base)
    fmt.Print("Masukkan pangkat: ")
    fmt.Scanln(&exp)

    fmt.Printf("%d^%d = %d\n", base, exp, pangkatIteratif(base, exp))
    fmt.Print("Masukkan angka untuk faktorial: ")
    fmt.Scanln(&n)
    fmt.Printf("%d! = %d\n", n, faktorialIteratif(n))
}
```

```
PS C:\Users\User\Documents\103112400028_MODUL6> go run "c:\Users\User\Documents\103112400028_MODUL6\main.go"
Masukkan bilangan: 5
Masukkan pangkat: 2
5^2 = 25
Masukkan angka untuk faktorial: 5
5! = 120
PS C:\Users\User\Documents\103112400028_MODUL6> |
```

Deskripsi: program di atas merupakan Bahasa Go yang berfungsi untuk menghitung pangkat dgn mengalihkan exp line. Program tersebut akan berjalan Ketika user memasukkan input berupa bilangan bulat.


```

package main

import "fmt"

func pangkatRekursif(base, exp int) int {
    if exp == 0 {
        return 1
    }
    return base * pangkatRekursif(base, exp-1)
}

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

func main() {
    var base, exp, n int
    // Input pangkat
    fmt.Print("Masukkan bilangan: ")
    fmt.Scanln(&base)
    fmt.Print("Masukkan pangkat: ")
    fmt.Scanln(&exp)

    fmt.Printf("%d^%d = %d\n", base, exp, pangkatRekursif(base, exp))

    // Input faktorial
    fmt.Print("Masukkan angka untuk faktorial: ")
    fmt.Scanln(&n)

    fmt.Printf("%d! = %d\n", n, faktorialRekursif(n))
}

```

```

PS C:\Users\User\Documents\103112400028_MODUL6> go run "c:\Users\User\Documents
Masukkan bilangan: 5
Masukkan pangkat: 2
5^2 = 25
Masukkan angka untuk faktorial: 10
10! = 3628800
PS C:\Users\User\Documents\103112400028_MODUL6>

```

Deskripsi: program di atas merupakan bahasa Go yang berfungsi untuk menghitung pangkat dari sebuah bilangan dan factorial dari sebuah angka dan menggunakan metode rekursif.

III. UNGUIDED

Source Code + Screenshot hasil program beserta penjelasan

```
// MUHAMMAD GAMEL AL GHIFARI
//103112400028
package main
import "fmt"
func main() {
    var n int
    fmt.Print("n:")
    fmt.Scanln(&n)

    fmt.Print("Sn: ")
    for i := 0; i <= 10; i++ {
        fmt.Printf("%d ", fibonacci(i))
    }
    fmt.Println()
}
func fibonacci(n int) int {
    if n <= 1 {
        return n
    }
    return fibonacci(n-1) + fibonacci(n-2)
}
```

```
PS C:\Users\User\Documents\103112400028_MODUL6> go run "c:\Users\User\Do
n:10
Sn: 0 1 1 2 3 5 8 13 21 34 55
PS C:\Users\User\Documents\103112400028_MODUL6> |
```

Deskripsi: program ini merupakan Bahasa Go yang berfungsi untuk mengimplementasikan fungsi rekursif untuk menghasilkan deret Fibonacci.

```
// MUHAMMAD GAMEL AL GHIFARI
//103112400028
package main

import "fmt"

func main() {
    var n int
    fmt.Print("N: ")
    fmt.Scan(&n)
    printPattern(n, 1)
}

func printStars(n int) {
    if n == 0 {
        return
    }
    fmt.Print("*")
    printStars(n - 1)
}

func printPattern(n, current int) {
    if current > n {
        return
    }
    printStars(current)
    fmt.Println()
    printPattern(n, current+1)
}
```

```
PS C:\Users\User\Documents\103112400028_MODULE6> go run "c:\Users\User\Documents
N: 3
*
**
***
I
PS C:\Users\User\Documents\103112400028_MODULE6> go run "c:\Users\User\Documents
N: 4
*
**
***
****
PS C:\Users\User\Documents\103112400028_MODULE6> 
```


Deskripsi: program di atas merupakan Bahasa Go yang berfungsi untuk menghasilkan atau membuat pola bintang dengan menggunakan fungsi rekursif. Pola bintang yang dihasilkan memiliki jumlah baris yang diinputkan.

```
//MUHAMMAD GAMEL AL GHIFARI
//103112400028
package main
import "fmt"
func main() {
    var n int
    fmt.Print("Masukkan n: ")
    fmt.Scan(&n)

    fmt.Printf("Faktor dari %d: ", n)
    printFactors(n, 1)
    fmt.Println()
}

func printFactors(n, i int) {
    if i > n {
        return
    }
    if n%i == 0 {
        fmt.Printf("%d ", i)
    }
    printFactors(n, i+1)
}
```

```
PS C:\Users\User\Documents\103112400028_MODUL6> go run "c:\Users\User\Documents\
Masukkan n: 5
Faktor dari 5: 1 5
PS C:\Users\User\Documents\103112400028_MODUL6> go run "c:\Users\User\Documents\
Masukkan n: 12
Faktor dari 12: 1 2 3 4 6 12
PS C:\Users\User\Documents\103112400028_MODUL6> |
```

Deskripsi: program ini merupakan bahasa Go yang dibuat untuk menemukan atau menampilkan faktor dari bilangan bulat positif yang diinputkan. Program ini menggunakan fungsi rekursif.

IV. KESIMPULAN

Dari praktikum ini, kita bisa tau bahwa rekursi adalah cara yang efektif untuk menyelesaikan masalah dengan mudah dipahami. Namun, jika tidak digunakan dengan benar, rekursi bisa menyebabkan program berjalan terlalu lama atau bahkan eror. Oleh karena itu, penting untuk selalu menentukan base case agar program bisa berhenti pada saat yang tepat. Secara keseluruhan, rekursi sangat berguna dalam banyak situasi, terutama dalam pemrograman berbasis matematika.

V. REFERENSI

Modul 5 Praktikum Algoritma dan Pemrograman 2