

**LAPORAN PRAKTIKUM
ALGORITMA DAN PEMROGRAMAN 2**

**MODUL V
REKURSIF**



Disusun Oleh :

**Fahri Ramadhan
2311102024**

Kelas

IF- 11- 06

Dosen Pengampu :

Abednego Dwi Septiadi, S.Kom., M.Kom

PROGRAM STUDI S1 TEKNIK INFORMATIKA

FAKULTAS INFORMATIKA

TELKOM UNIVERSITY PURWOKERTO 2024

I. DASAR TEORI

Rekursif adalah teknik pemrograman di mana sebuah fungsi memanggil dirinya sendiri untuk menyelesaikan masalah yang lebih kecil dari masalah aslinya, hingga mencapai kondisi dasar (base case) yang menghentikan pemanggilan rekursif. Dengan menggunakan rekursi, masalah yang besar dan kompleks dapat dipecah menjadi sub-masalah yang lebih sederhana dan lebih mudah diselesaikan. Dalam rekursi, setiap kali fungsi memanggil dirinya sendiri, keadaan saat ini disimpan di "stack" memori hingga base case tercapai. Setelah itu, fungsi akan kembali menyelesaikan permasalahan secara bertahap dari tumpukan stack (proses ini disebut "unwinding").

II. GUIDED 1

Sourcecode

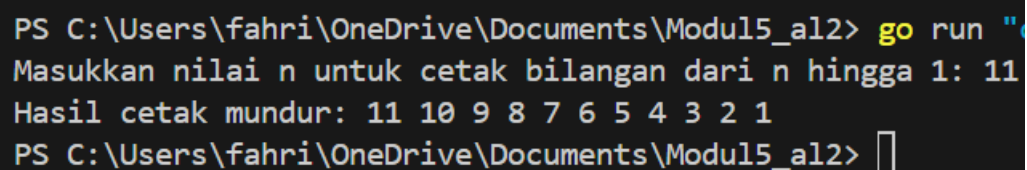
```
package main

import "fmt"

// Fungsi untuk mencetak bilangan dari n hingga 1
func cetakMundur(n int) {
    if n == 1 {
        fmt.Println(n)
        return
    }
    fmt.Print(n, " ")
    cetakMundur(n-1)
}

func main() {
    var n int
    fmt.Print("Masukkan nilai n untuk cetak
bilangan dari n hingga 1: ")
    fmt.Scanln(&n)
    fmt.Print("Hasil cetak mundur: ")
    cetakMundur(n)
}
```

Screenshoot Output



```
PS C:\Users\fahri\OneDrive\Documents\Modul5_a12> go run "
Masukkan nilai n untuk cetak bilangan dari n hingga 1: 11
Hasil cetak mundur: 11 10 9 8 7 6 5 4 3 2 1
PS C:\Users\fahri\OneDrive\Documents\Modul5_a12> █
```

Deskripsi Program

Program bertujuan untuk mencetak bilangan secara mundur dari angka yang dimasukkan oleh pengguna hingga 1. Di dalam fungsi main, program meminta pengguna untuk memasukkan nilai n, kemudian mencetak hasil dari fungsi cetakMundur. Fungsi cetakMundur menggunakan rekursi untuk menampilkan angka dari n hingga 1. Pada setiap langkah, fungsi ini akan

mencetak nilai n saat ini dan kemudian memanggil dirinya sendiri dengan nilai n-1 hingga mencapai angka 1, yang kemudian dicetak tanpa spasi, dan menghentikan rekursi dengan perintah return.

2. Guided 2 Sourcecode

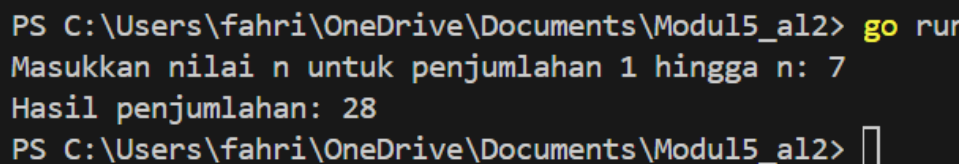
```
package main

import "fmt"

// Fungsi untuk menghitung penjumlahan 1 hingga n
func jumlahRekursi(n int) int {
    if n == 1 {
        return 1
    }
    return n + jumlahRekursi(n-1)
}

func main() {
    var n int
    fmt.Print("Masukkan nilai n untuk penjumlahan 1 hingga n: ")
    fmt.Scanln(&n)
    fmt.Println("Hasil penjumlahan:", jumlahRekursi(n))
}
```

Screenshoot Output



```
PS C:\Users\fahri\OneDrive\Documents\Modul5_al2> go run
Masukkan nilai n untuk penjumlahan 1 hingga n: 7
Hasil penjumlahan: 28
PS C:\Users\fahri\OneDrive\Documents\Modul5_al2> █
```

Deskripsi Program

Program berfungsi untuk menghitung jumlah bilangan dari 1 hingga n menggunakan rekursi. Di dalam fungsi main, program meminta pengguna untuk memasukkan nilai n, lalu mencetak hasil penjumlahan dari fungsi jumlahRekursif. Fungsi jumlahRekursif menggunakan konsep rekursi, di mana setiap pemanggilan fungsi akan menambahkan nilai n saat ini dengan hasil pemanggilan fungsi berikutnya dengan nilai n-1, hingga mencapai kondisi dasar di mana n sama dengan 1. Pada titik ini, fungsi mengembalikan 1, dan rekursi berakhir.

3. Guided 3 Sourcecode

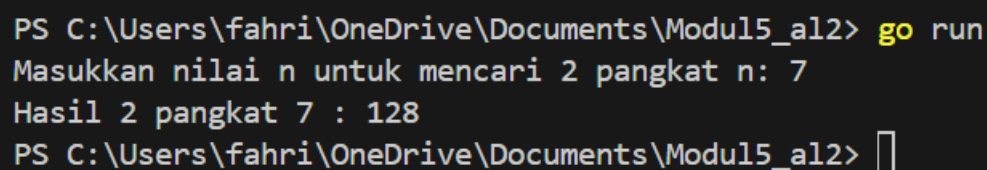
```
package main

import "fmt"

// Fungsi untuk mencari 2 pangkat n
func pangkatDua(n int) int {
    if n == 0 {
        return 1
    }
    return 2 * pangkatDua(n-1)
}

func main() {
    var n int
    fmt.Print("Masukkan nilai n untuk mencari 2 pangkat n: ")
    fmt.Scanln(&n)
    fmt.Println("Hasil 2 pangkat", n, ":", pangkatDua(n))
}
```

Screenshoot Output



```
PS C:\Users\fahri\OneDrive\Documents\Modul5_al2> go run
Masukkan nilai n untuk mencari 2 pangkat n: 7
Hasil 2 pangkat 7 : 128
PS C:\Users\fahri\OneDrive\Documents\Modul5_al2> █
```

Deskripsi Program

Program ini menghitung hasil dari 2 pangkat n menggunakan rekursi. Pada fungsi main, program meminta pengguna untuk memasukkan nilai n, kemudian mencetak hasil perhitungan dari fungsi pangkatDua. Fungsi pangkatDua bekerja dengan pendekatan rekursif: jika n bernilai 0, fungsi mengembalikan nilai 1 sebagai kondisi dasar (karena 2 pangkat 0 adalah 1). Untuk nilai n yang lebih besar, fungsi ini mengalikan 2 dengan hasil pemanggilan dirinya sendiri dengan nilai n-1, hingga rekursi mencapai kondisi dasar dan mengembalikan hasil perhitungan 2 pangkat n.

4. Guided 4 Sourcecode

```
package main

import "fmt"

// Fungsi untuk menghitung faktorial n!
func faktorial(n int) int {
    if n == 0 || n == 1 {
        return 1
    }
    return n * faktorial(n-1)
}

func main() {
    var n int
    fmt.Print("Masukkan nilai n untuk mencari faktorial n!: ")
    fmt.Scanln(&n)
    fmt.Println("Hasil faktorial dari", n, ":", faktorial(n))
}
```

Screenshoot Output

```
PS C:\Users\fahri\OneDrive\Documents\Modul5_al2> go run  
Masukkan nilai n untuk mencari faktorial n!: 7  
Hasil faktorial dari 7 : 5040  
PS C:\Users\fahri\OneDrive\Documents\Modul5_al2> █
```

Deskripsi Program

Program ini bertujuan untuk menghitung nilai faktorial dari bilangan n menggunakan rekursi. Di dalam fungsi main, program meminta pengguna untuk memasukkan nilai n , lalu mencetak hasil perhitungan faktorial yang didapat dari fungsi faktorial. Fungsi faktorial menggunakan pendekatan rekursif dengan kondisi dasar saat n bernilai 0 atau 1, di mana fungsi mengembalikan nilai 1 (karena faktorial dari 0 dan 1 adalah 1). Untuk nilai n yang lebih besar, fungsi ini mengalikan n dengan hasil pemanggilan dirinya sendiri dengan nilai $n-1$, dan proses ini berlanjut hingga mencapai kondisi dasar, memberikan hasil akhir faktorial dari 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 $S_2 = S_{n-1} + S_{n-2}$. Berikut ini adalah contoh nilai deret fibonacci hingga suku ke-10. Buatlah program yang mengimplementasikan fungsi rekursif pada deret fibonacci tersebut.

Sourcecode

```
package main

import (
    "fmt"
)

func fibonacci(n int) int {
    if n <= 1 {
        return n
    }
    return fibonacci(n-1) + fibonacci(n-2)
}

func main() {
    n := 10
    fmt.Printf("Deret Fibonacci hingga suku ke-
%d:\n", n)
    for i := 0; i <= n; i++ {
        fmt.Printf("Suku ke-%d: %d\n", i,
fibonacci(i))
    }
}
```


Screenshoot Output

```
PS C:\Users\fahri\OneDrive\Documents\Modul5_a12> go r
Deret Fibonacci hingga suku ke-10:
Suku ke-0: 0
Suku ke-1: 1
Suku ke-2: 1
Suku ke-3: 2
Suku ke-4: 3
Suku ke-5: 5
Suku ke-6: 8
Suku ke-7: 13
Suku ke-8: 21
Suku ke-9: 34
Suku ke-10: 55
PS C:\Users\fahri\OneDrive\Documents\Modul5_a12>
```

Deskripsi Program

Program menghitung dan menampilkan deret Fibonacci hingga suku ke-n yang diinputkan oleh pengguna. Dalam fungsi fibonacci, program menggunakan rekursi untuk menghitung nilai Fibonacci dari suatu bilangan n, di mana fibonacci(0) mengembalikan 0 dan fibonacci(1) mengembalikan 1. Nilai Fibonacci dari suku-suku berikutnya dihitung sebagai jumlah dari dua suku sebelumnya (fibonacci(n-1) + fibonacci(n-2)). Pada fungsi main, program meminta pengguna memasukkan nilai n, kemudian menggunakan loop for untuk memanggil fungsi fibonacci dan mencetak setiap suku dalam deret Fibonacci mulai dari suku ke-0 hingga suku ke-n.

UNGUIDED 2

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

Sourcecode

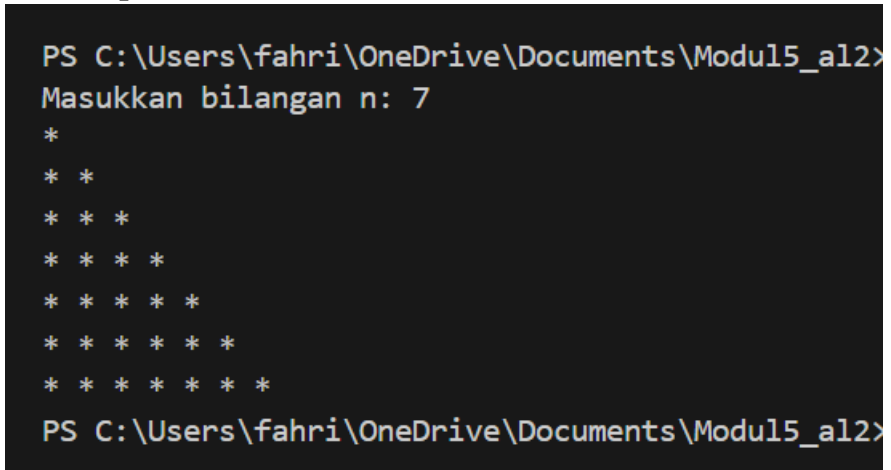
```
package main
import "fmt"

func cetakbintang(n int) {
    if n <= 0 {
        return
    }
    cetakbintang(n - 1)
    for i := 0; i < n; i++ {
        fmt.Print("* ")
    }
    fmt.Println()
}

func main() {
    var n int
    fmt.Print("Masukkan bilangan n: ")
    fmt.Scan(&n)

    fmt.Print("\n")
    cetakbintang(n)
}
```

Screenshoot Output



```
PS C:\Users\fahri\OneDrive\Documents\Modul5_al2>
Masukkan bilangan n: 7
*
* *
* * *
* * * *
* * * * *
* * * * *
* * * * *
* * * * *
PS C:\Users\fahri\OneDrive\Documents\Modul5_al2>
```

Deskripsi Program

Program menampilkan pola bintang segitiga yang berurutan dari satu baris hingga baris dengan jumlah bintang terbanyak sesuai dengan input n yang dimasukkan oleh pengguna. Di dalam fungsi main, program meminta

pengguna untuk memasukkan nilai n , kemudian mencetak pola bintang menggunakan fungsi `printStars`. Fungsi `printStars` adalah fungsi rekursif yang pertama-tama memanggil dirinya sendiri dengan $n-1$ hingga n mencapai 0, sehingga menunda pencetakan bintang hingga rekursi selesai. Setelah itu, untuk setiap pemanggilan dalam urutan balik (dari 1 hingga n), fungsi ini mencetak sejumlah bintang yang sesuai dengan nilai n saat ini, menghasilkan pola segitiga bintang yang terbalik.

UNGUIDED 3

3. Buatlah program yang mengimplementasikan rekursif untuk menampilkan faktor bilangan dari suatu N , atau bilangan yang apa saja yang habis membagi N . Masukan terdiri dari sebuah bilangan bulat positif N . Keluaran terdiri dari barisan bilangan yang menjadi faktor dari N (terurut dari 1 hingga N ya).

Sourcecode

```
package main

import "fmt"

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

func main() {
    var n int
    fmt.Print("Masukkan bilangan bulat positif N: ")
    fmt.Scanln(&n)
    fmt.Print("Faktor dari ", n, " adalah: ")
    printFactors(n, 1)
    fmt.Println()
}
```

Screenshoot Output

```
PS C:\Users\fahri\OneDrive\Documents\Modul5_al2> go ru
Masukkan bilangan bulat positif N: 11
Faktor dari 11 adalah: 1 11
PS C:\Users\fahri\OneDrive\Documents\Modul5_al2> █
```

Deskripsi Program

Program berfungsi untuk menampilkan semua faktor dari bilangan bulat positif n yang dimasukkan oleh pengguna. Dalam fungsi main, program meminta pengguna untuk memasukkan nilai n , kemudian memanggil fungsi `printFactors` untuk mencetak faktor-faktor dari n . Fungsi `printFactors` adalah fungsi rekursif yang menerima dua parameter: n (bilangan yang akan dicari faktornya) dan i (sebagai pembagi yang dimulai dari 1). Pada setiap pemanggilan, fungsi ini mengecek apakah n habis dibagi oleh i . Jika iya, maka i dicetak sebagai faktor. Setelah itu, fungsi memanggil dirinya sendiri dengan $i+1$, hingga i lebih besar dari n , yang menghentikan rekursi.

UNGUIDED 4

4. Buatlah program yang menimplementasikan rekursif untuk menampilkan barisan bilangan tertentu
Masukan terdiri dari sebuah bilangan bulat positif N .
Keluaran terdiri dari barisan bilangan dari N hingga 1 dan kembali ke N .

Sourcecode

```
package main
import (
    "fmt"
)

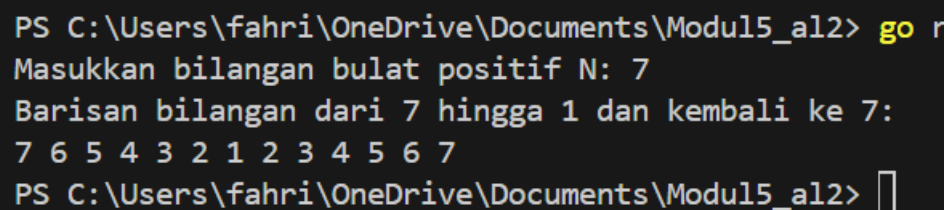
func printSeries(N, current int) {
    if current == 1 {
        fmt.Printf("%d ", current)
        return
    }

    fmt.Printf("%d ", current)
    printSeries(N, current-1)
    fmt.Printf("%d ", current)
}

func main() {
    var N int
    fmt.Print("Masukkan bilangan bulat positif N: ")
    fmt.Scan(&N)

    fmt.Printf("Barisan bilangan dari %d hingga 1 dan kembali ke %d:\n", N, N)
    printSeries(N, N)
    fmt.Println()
}
```

Screenshoot Output



```
PS C:\Users\fahri\OneDrive\Documents\Modul5_al2> go r
Masukkan bilangan bulat positif N: 7
Barisan bilangan dari 7 hingga 1 dan kembali ke 7:
7 6 5 4 3 2 1 2 3 4 5 6 7
PS C:\Users\fahri\OneDrive\Documents\Modul5_al2> █
```

Deskripsi Program

Program menampilkan bilangan bulat positif dari n hingga 1 secara menurun, kemudian kembali dari 1 hingga n secara menaik. Dalam fungsi

main, program meminta pengguna untuk memasukkan bilangan bulat positif n , lalu mencetak barisan bilangan tersebut menggunakan dua fungsi rekursif: `printDescending` dan `printAscending`. Fungsi `printDescending` mencetak bilangan dari n hingga 1 secara menurun, dengan memanggil dirinya sendiri dengan nilai $n-1$ hingga n kurang dari 1, yang menghentikan rekursi. Setelah selesai, fungsi `printAscending` mencetak bilangan dari 1 hingga n secara menaik, dimulai dengan `current` yang awalnya bernilai 1 dan bertambah 1 di setiap panggilan rekursif hingga mencapai n . Hasilnya adalah deretan bilangan dari n ke 1 dan kembali ke n .

UNGUIDED 5

5. Buatlah program yang menimplementasikan rekursif untuk menampilkan barisan bilangan ganjil.

Masukan terdiri dari sebuah bilangan bulat positif N . Keluaran terdiri dari barisan bilangan ganjil dari 1 hingga N .

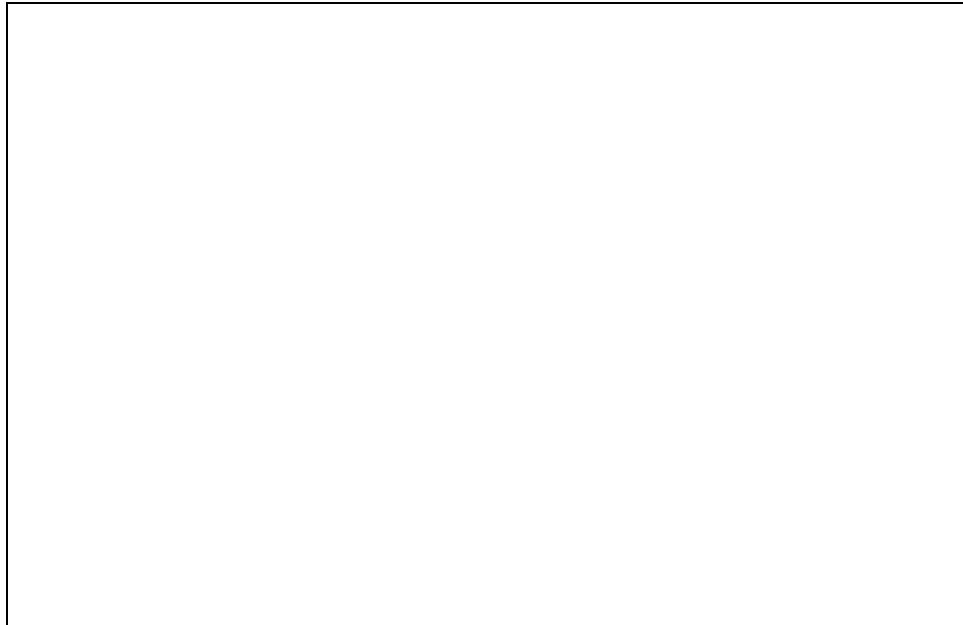
Sourcecode

```
package main
import (
    "fmt"
)

func printOddNumbers(N, current int) {
    if current > N {
        return
    }
    if current%2 != 0 {
        fmt.Printf("%d ", current)
    }
    printOddNumbers(N, current+1)
}

func main() {
    var N int
    fmt.Print("Masukkan bilangan bulat positif N: ")
    fmt.Scan(&N)

    fmt.Printf("Bilangan ganjil dari 1 hingga %d\n", N)
    printOddNumbers(N, 1)
    fmt.Println()
}
```



Screenshoot Output

```
PS C:\Users\fahri\OneDrive\Documents\Modul5_a12> go r
Masukkan bilangan bulat positif N: 11
Bilangan ganjil dari 1 hingga 11 adalah:
1 3 5 7 9 11
PS C:\Users\fahri\OneDrive\Documents\Modul5_a12> █
```

Deskripsi Program

Program berfungsi untuk mencetak semua bilangan ganjil dari 1 hingga bilangan n yang dimasukkan oleh pengguna. Pada fungsi main, program meminta pengguna untuk memasukkan nilai n, lalu memanggil fungsi printtambahangka untuk mencetak bilangan ganjil dalam rentang tersebut. Fungsi printtambahangka adalah fungsi rekursif yang menerima dua parameter: n (batas atas) dan current (bilangan yang diperiksa, dimulai dari 1). Setiap kali fungsi dipanggil, ia memeriksa apakah current adalah bilangan ganjil. Jika iya, nilai current dicetak. Fungsi ini kemudian memanggil dirinya sendiri dengan nilai current+1 hingga current melebihi n, yang menghentikan rekursi. Hasil akhir dari program ini adalah deretan bilangan ganjil dari 1 hingga n.

UNGUIDED 6

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 dipangkatkan y.

Catatan: diperbolehkan menggunakan asterik "*", tapi dilarang menggunakan import "math".

Sourcecode

```
package main
import "fmt"

func pangkat(x, y int) int {
    if y == 0 {
        return 1
    } else if y == 1 {
        return x
    } else {
        return x * pangkat(x, y-1)
    }
}

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

    fmt.Println(pangkat(x, y))
}
```


Screenshoot Output

```
PS C:\Users\fahri\OneDrive\Documents\Modul5_a12> go r
Masukkan x: 2
Masukkan y: 4
16
PS C:\Users\fahri\OneDrive\Documents\Modul5_a12> █
```

Deskripsi Program

Program ini menghitung hasil pangkat dari dua bilangan bulat, xxx sebagai basis dan yyy sebagai eksponen, menggunakan fungsi rekursif tanpa menggunakan pustaka matematika. Fungsi pangkat menerima dua parameter, basis dan eksponen. Jika eksponen bernilai 0, fungsi mengembalikan 1 karena setiap bilangan dipangkatkan 0 adalah 1 (ini adalah kasus dasar untuk menghentikan rekursi). Jika tidak, fungsi mengalikan basis dengan hasil dari pangkat(basis, eksponen-1), yang mengurangi nilai eksponen secara bertahap hingga mencapai 0. Dalam fungsi main, pengguna diminta memasukkan nilai x dan y, kemudian program memanggil fungsi pangkat dengan input tersebut dan mencetak hasilnya.