

LAPORAN PRAKTIKUM
ALGORITMA DAN PEMROGRAMAN 2
MODUL 5
REKURSIF



Oleh :

FAHRUR RIZQI

2311102059

S1 IF-11-02

S1 TEKNIK INFORMATIKA
TELKOM UNIVERSITY PURWOKERTO
2024

I. DASAR TEORI

a) Pengertian Rekursif

Rekursif adalah teknik pemrograman di mana suatu subprogram (baik fungsi atau prosedur) dapat memanggil dirinya sendiri untuk menyelesaikan masalah. Ini memungkinkan penyelesaian masalah yang kompleks dengan membaginya menjadi sub-masalah yang lebih kecil dan identik.

b) Struktur Dasar Rekursif

Setiap algoritma rekursif terdiri dari dua komponen utama:

- **Base-case (Basis):** Ini adalah kondisi yang menghentikan proses rekursif. Tanpa adanya base-case, proses rekursif akan terus berlanjut tanpa henti, yang dapat menyebabkan kesalahan seperti stack overflow.
- **Recursive-case:** Ini adalah bagian di mana subprogram memanggil dirinya sendiri. Kondisi ini merupakan negasi dari base-case, yang berarti bahwa pemanggilan akan terus dilakukan selama kondisi base-case tidak terpenuhi.

c) Contoh Implementasi

Modul ini memberikan beberapa contoh implementasi rekursif, seperti:

- **Fungsi cetak:** Menampilkan angka dari nilai awal hingga nilai tertentu, dengan menambahkan kondisi base-case untuk menghentikan pemanggilan ketika mencapai angka tertentu.
- **Faktorial:** Menghitung nilai faktorial dari suatu bilangan, di mana base-case adalah ketika n sama dengan 0 atau 1.
- **Deret Fibonacci:** Menghitung nilai deret Fibonacci menggunakan fungsi rekursif, di mana setiap suku adalah hasil penjumlahan dari dua suku sebelumnya.

d) Proses Rekursif

Proses rekursif dapat dibagi menjadi dua fase:

- **Forward:** Di mana pemanggilan subprogram dilakukan hingga mencapai base-

case.

- **Backward:** Setelah mencapai base-case, proses kembali ke pemanggilan sebelumnya, menyelesaikan setiap langkah yang tertunda.

e) Pentingnya Base-case

Base-case adalah elemen terpenting dalam algoritma rekursif. Tanpa pemahaman yang jelas tentang kondisi ini, sulit untuk merancang algoritma rekursif yang efektif. Base-case harus ditentukan terlebih dahulu sebelum membuat program rekursif.

f) Alternatif Rekursif

Setiap algoritma rekursif memiliki padanan dalam bentuk algoritma iteratif. Ini menunjukkan bahwa rekursif adalah salah satu cara untuk menyelesaikan masalah yang juga dapat diselesaikan dengan menggunakan loop.

Kesimpulan

Rekursif adalah teknik yang kuat dalam pemrograman yang memungkinkan penyelesaian masalah kompleks dengan cara yang lebih sederhana. Memahami komponen dasar seperti base-case dan recursive-case sangat penting untuk mengimplementasikan algoritma rekursif yang efektif dan efisien.

Modul ini memberikan pemahaman yang mendalam tentang rekursif, serta contoh-contoh praktis yang dapat digunakan sebagai referensi dalam pemrograman.

II. GUIDED

I. Guided 1

Sourcode

```
package main

import "fmt"

func main() {
    var n int
    fmt.Print("Masukan Angka: ")
    fmt.Scan(&n)
    fmt.Println("Hasilnya")
    baris(n)
}

func baris(bilangan int) {
    if bilangan == 1 {
        fmt.Println(1)
    } else {
        fmt.Println(bilangan)
        baris(bilangan - 1)
    }
}
```

Output :

```
PS C:\Users\ASUS\Documents\File Fahrur\SEMESTER 3\PRAKTIKUM ALPRO 2\MODUL 5\Fahrur Rizqi_2311102059> go
rs\ASUS\Documents\File Fahrur\SEMESTER 3\PRAKTIKUM ALPRO 2\MODUL 5\Fahrur Rizqi_2311102059\Guided1.go"
Masukan Angka: 5
Hasilnya
5
4
3
2
1
PS C:\Users\ASUS\Documents\File Fahrur\SEMESTER 3\PRAKTIKUM ALPRO 2\MODUL 5\Fahrur Rizqi_2311102059> █
```

Deskripsi Program

Program ini ditulis dalam bahasa Go untuk mencetak angka dari `n` hingga 1. Setelah meminta pengguna memasukkan sebuah angka (`n`), program menggunakan fungsi rekursif bernama `baris` yang mencetak nilai `bilangan`. Jika `bilangan` sama dengan 1, fungsi mencetak 1; jika tidak, ia mencetak nilai saat ini dan memanggil dirinya sendiri dengan `bilangan - 1`. Hasilnya adalah urutan angka yang dicetak secara menurun berdasarkan input yang diberikan.

II. Guided 2

Sourcode

```
package main

import "fmt"

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

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

Output :

```
PS C:\Users\ASUS\Documents\File Fahrur\SEMESTER 3\PRAKTIKUM ALPRO 2\MODUL 5\Fahrur Rizqi_2311102059> go
rs\ASUS\Documents\File Fahrur\SEMESTER 3\PRAKTIKUM ALPRO 2\MODUL 5\Fahrur Rizqi_2311102059\tempCodeRunn
Masukan Angka: 5
Hasilnya
15
PS C:\Users\ASUS\Documents\File Fahrur\SEMESTER 3\PRAKTIKUM ALPRO 2\MODUL 5\Fahrur Rizqi_2311102059> █
```

Deskripsi Program

Program ini menghitung jumlah bilangan bulat dari 1 hingga n menggunakan pendekatan rekursif. Setelah pengguna memasukkan angka n, fungsi `penjumlahan` dipanggil untuk menjumlahkan semua bilangan dari n hingga 1. Jika n sama dengan 1, fungsi mengembalikan 1; jika tidak, fungsi menjumlahkan n dengan hasil penjumlahan dari n-1. Hasil akhirnya ditampilkan di layar.

III. UNGUIDED

a. 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

Sourcode

```
package main

import "fmt"

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

func fibonacciSeries(n int) {
    for i := 0; i <= n; i++ {
        fmt.Print(fibonacci(i), " ")
    }
    fmt.Println()
}

func fibonacci(n int) int {
    if n <= 1 {
```

```

    return n
}

return fibonacci(n-1) + fibonacci(n-2)
}

```

Output :

```

PS C:\Users\ASUS\Documents\File Fahrur\SEMESTER 3\PRAKTIKUM ALPRO 2\MODUL 5\Fahrur Rizqi_2311102059> go
rs\ASUS\Documents\File Fahrur\SEMESTER 3\PRAKTIKUM ALPRO 2\MODUL 5\Fahrur Rizqi_2311102059\Unguided1.go'
Masukkan nilai n: 10
0 1 1 2 3 5 8 13 21 34 55
PS C:\Users\ASUS\Documents\File Fahrur\SEMESTER 3\PRAKTIKUM ALPRO 2\MODUL 5\Fahrur Rizqi_2311102059>

```

Deskripsi Program

Program ini menghitung deret Fibonacci hingga suku ke-10 menggunakan metode rekursif. Fungsi `fibonacci` mengembalikan nilai Fibonacci untuk bilangan bulat n, di mana $Fibonacci(0) = 0$ dan $Fibonacci(1) = 1$. Untuk n lebih besar dari 1, fungsi menghitung nilai dengan menjumlahkan dua suku sebelumnya. Di dalam fungsi `main`, program mencetak deret Fibonacci dari suku ke-0 hingga suku ke-10 ke layar.

b. Unguided 2

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

Contoh masukan dan keluaran:

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

Sourcode

```
package main

import "fmt"

func main() {
    var n int
    //fahrur059

    fmt.Print("Masukkan jumlah baris: ")
    fmt.Scanln(&n)

    printStars(n)
}

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

Output :

```
PS C:\Users\ASUS\Documents\File Fahrur\SEMESTER 3\PRAKTIKUM ALPRO 2\MODUL 5\Fahrur Rizqi_2311102059> go run rs\ASUS\Documents\File Fahrur\SEMESTER 3\PRAKTIKUM ALPRO 2\MODUL 5\Fahrur Rizqi_2311102059\tempCodeRunn
Masukkan jumlah baris: 5
*****
****
***
**
*
PS C:\Users\ASUS\Documents\File Fahrur\SEMESTER 3\PRAKTIKUM ALPRO 2\MODUL 5\Fahrur Rizqi_2311102059>
```

Deskripsi Program

Program ini mencetak pola bintang berdasarkan jumlah baris yang dimasukkan oleh pengguna. Setelah pengguna memasukkan angka n , fungsi `printStars` dipanggil untuk mencetak n bintang di satu baris. Fungsi ini menggunakan rekursi, di mana setelah mencetak bintang untuk baris saat ini, ia memanggil dirinya sendiri untuk mencetak bintang pada baris berikutnya dengan jumlah yang berkurang satu. Proses ini berlanjut hingga n mencapai 0.

c. Unguided 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).

Contoh masukan dan keluaran:

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

Sourcode

```
package main

import "fmt"

func main() {
    fmt.Print("Masukkan bilangan bulat positif: ")
    var n int
    fmt.Scanln(&n)
    //fahrur059
    fmt.Println("Faktor dari", n, "adalah:", faktor(n))
}
```

```
func faktor(n int) []int {
    var faktor []int
    for i := 1; i <= n; i++ {
        if n%i == 0 {
            faktor = append(faktor, i)
        }
    }
    return faktor
}
```

Output :

```
PS C:\Users\ASUS\Documents\File Fahrur\SEMESTER 3\PRAKTIKUM ALPRO 2\MODUL 5\Fahrur Rizqi_2311102059> go run .
Masukkan bilangan bulat positif: 5
Faktor dari 5 adalah: [1 5]
PS C:\Users\ASUS\Documents\File Fahrur\SEMESTER 3\PRAKTIKUM ALPRO 2\MODUL 5\Fahrur Rizqi_2311102059>
```

Deskripsi Program

Program ini berfungsi untuk menghitung faktor dari bilangan bulat positif yang dimasukkan oleh pengguna. Setelah pengguna memasukkan angka n, fungsi `faktor` dipanggil untuk mencari semua bilangan yang dapat membagi n tanpa sisa. Fungsi ini menggunakan loop untuk memeriksa setiap bilangan dari 1 hingga n, dan jika bilangan tersebut adalah faktor, ia menambahkannya ke dalam slice. Hasil akhirnya adalah daftar faktor yang dicetak di layar.

d. Unguided 4

Buatlah program yang mengimplementasikan 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.

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

Sourcode

```
package main

import "fmt"

func printSequence(n int) {
    if n == 1 {
        fmt.Print(n, " ")
        return
    }
    fmt.Print(n, " ")
    //fahrur059
    printSequence(n - 1)
    fmt.Print(n, " ")
}

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

    fmt.Println("Keluaran:")
    printSequence(N)
    fmt.Println()
}
```

Output :

```
PS C:\Users\ASUS\Documents\File Fahrur\SEMESTER 3\PRAKTIKUM ALPRO 2\MODUL 5\Fahrur Rizqi_2311102059>
rs\ASUS\Documents\File Fahrur\SEMESTER 3\PRAKTIKUM ALPRO 2\MODUL 5\Fahrur Rizqi_2311102059\tempCodeRu
Masukkan bilangan bulat positif N: 5
Keluaran:
5 4 3 2 1 2 3 4 5
PS C:\Users\ASUS\Documents\File Fahrur\SEMESTER 3\PRAKTIKUM ALPRO 2\MODUL 5\Fahrur Rizqi_2311102059>
```

Deskripsi Program

Program ini mencetak urutan angka dari N hingga 1, kemudian kembali mencetak angka dari 1 hingga N. Setelah pengguna memasukkan bilangan bulat positif N, fungsi `printSequence` dipanggil secara rekursif. Saat fungsi dipanggil, ia mencetak angka saat ini, kemudian memanggil dirinya sendiri dengan nilai N yang berkurang satu. Setelah mencapai 0, fungsi kembali mencetak angka saat ini saat proses rekursi kembali ke atas. Hasil akhirnya adalah urutan angka yang dicetak di layar.

e. Unguided 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

Sourcode

```
package main

import "fmt"

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

    fmt.Println("Keluaran:")
    printOddNumbers(n)
}
```

```

func printOddNumbers(n int) {
    if n >= 1 {
        printOddNumbers(n - 2)
        //fahrur059
        fmt.Print(n, " ")
    }
}

```

Output :

```

PS C:\Users\ASUS\Documents\File Fahrur\SEMESTER 3\PRAKTIKUM ALPRO 2\MODUL 5\Fahrur Rizqi_2311102059> g
rs\ASUS\Documents\File Fahrur\SEMESTER 3\PRAKTIKUM ALPRO 2\MODUL 5\Fahrur Rizqi_2311102059\tempCodeRun
Masukkan bilangan bulat positif: 5
Keluaran:
1 3 5
PS C:\Users\ASUS\Documents\File Fahrur\SEMESTER 3\PRAKTIKUM ALPRO 2\MODUL 5\Fahrur Rizqi_2311102059>

```

Deskripsi Program

Program ini mencetak angka ganjil dari 1 hingga N secara terbalik. Setelah pengguna memasukkan bilangan bulat positif N, fungsi `printOddNumbers` dipanggil secara rekursif. Fungsi ini mengurangi n sebesar 2 pada setiap panggilan hingga n kurang dari 1, dan kemudian mencetak angka ganjil yang tersisa saat kembali dari setiap panggilan. Hasilnya adalah deret angka ganjil yang dicetak di layar.

f. Unguided 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".

Contoh masukan dan keluaran:

No	Masukan	Keluaran
1	2 2	4
2	5 3	125

Sourcode

```
package main

import "fmt"

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 nilai x: ")
    //fahrur059
    fmt.Scan(&x)
    fmt.Print("Masukkan nilai y: ")
    fmt.Scan(&y)

    result := power(x, y)

    fmt.Printf("%d pangkat %d = %d\n", x, y, result)
}
```

Output :

```
PS C:\Users\ASUS\Documents\File Fahrur\SEMESTER 3\PRAKTIKUM ALPRO 2\MODUL 5\Fahrur Rizqi_2311102059>
rs\ASUS\Documents\File Fahrur\SEMESTER 3\PRAKTIKUM ALPRO 2\MODUL 5\Fahrur Rizqi_2311102059\tempCodeRu
Masukkan nilai x: 2
Masukkan nilai y: 2
2 pangkat 2 = 4
PS C:\Users\ASUS\Documents\File Fahrur\SEMESTER 3\PRAKTIKUM ALPRO 2\MODUL 5\Fahrur Rizqi_2311102059>
```

Deskripsi Program

Program ini menghitung hasil pangkat dari bilangan bulat x yang dipangkatkan dengan bilangan bulat y . Setelah pengguna memasukkan nilai x dan y , fungsi `power` dipanggil secara rekursif untuk menghitung x pangkat y . Jika y sama dengan 0, fungsi mengembalikan 1, sebagai dasar pangkat. Jika tidak, fungsi mengalikan x dengan hasil panggilan `power` untuk $y-1$. Hasil pangkat kemudian dicetak di layar.