

**LAPORAN PRAKTIKUM  
ALGORITMA DAN PEMROGRAMAN 2**

**MODUL 5  
REKRUSIF**



**Disusun Oleh :**

**Aby Hakim Al Yasiry Faozi/2311102208**

**IF-11-06**

**Dosen Pengampu :**

**Abednego Dwi Septiadi**

**PROGRAM STUDI S1 TEKNIK INFORMATIKA**

**FAKULTAS INFORMATIKA**

**TELKOM UNIVERSITY PURWOKERTO**

**2024**

## **I. DASAR TEORI**

Rekursif adalah konsep dalam pemrograman di mana suatu fungsi memanggil dirinya sendiri secara langsung atau tidak langsung untuk menyelesaikan suatu permasalahan. Pada dasarnya, rekursif berfungsi untuk memecah masalah besar menjadi sub-masalah yang lebih kecil hingga mencapai kondisi paling dasar (base case) di mana pemanggilan fungsi dapat dihentikan.

### **1. Base Case (Kondisi Dasar)**

Base case adalah kondisi di mana fungsi tidak akan memanggil dirinya lagi. Kondisi ini sangat penting untuk mencegah terjadinya pemanggilan fungsi yang berulang tanpa henti atau infinite recursion, yang dapat menyebabkan program mengalami error atau kehabisan memori. Misalnya, dalam menghitung faktorial, base case-nya adalah saat nilai yang dimasukkan adalah 0 atau 1, di mana nilai faktorialnya sudah diketahui (1).

### **2. Recursive Case (Kondisi Rekursif)**

Recursive case adalah kondisi di mana fungsi akan memanggil dirinya sendiri dengan parameter yang lebih kecil atau lebih sederhana dibandingkan dengan parameter awal. Misalnya, dalam perhitungan faktorial, nilai faktorial dari sebuah bilangan  $n$

### **3. Stack (Tumpukan)**

Pemanggilan rekursif biasanya ditangani oleh stack pemanggilan (call stack) di dalam memori. Setiap kali fungsi memanggil dirinya sendiri, komputer akan menyimpan informasi mengenai pemanggilan tersebut dalam stack. Ketika base case tercapai, komputer akan mulai kembali ke pemanggilan sebelumnya (proses ini disebut unwinding), sehingga hasil dari pemanggilan fungsi dapat diperoleh secara berurutan dari bawah ke atas.

## II. GUIDED

### 1. Soal Studi Case

#### Sourcecode

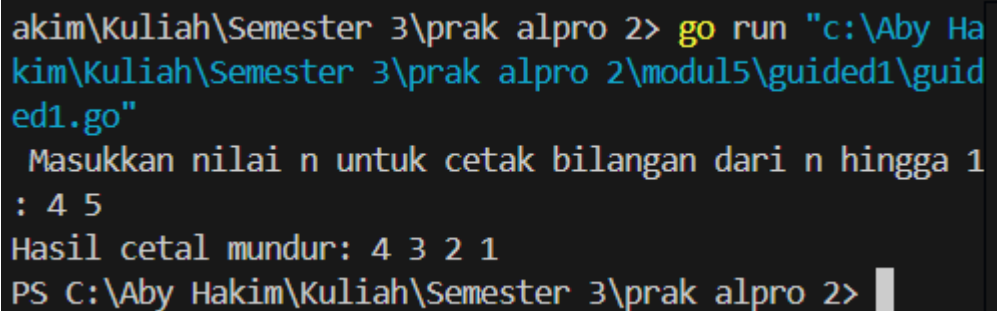
```
package main

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



```
akim\Kuliah\Semester 3\prak alpro 2> go run "c:\Aby Ha
kim\Kuliah\Semester 3\prak alpro 2\modul5\guided1\guid
ed1.go"
Masukkan nilai n untuk cetak bilangan dari n hingga 1
: 4 5
Hasil cetak mundur: 4 3 2 1
PS C:\Aby Hakim\Kuliah\Semester 3\prak alpro 2> |
```

#### Deskripsi Program

##### Fungsi cetakMundur:

- Mencetak angka dari n hingga 1 secara rekursif.
- Jika  $n == 1$ , cetak 1 dan selesai.
- Jika tidak, cetak n diikuti dengan panggilan rekursif cetakMundur( $n - 1$ ).

**Fungsi main:**

- Membaca nilai n dari input pengguna.
- Memanggil fungsi cetakMundur untuk mencetak angka dari n hingga 1.

**2. Soal Studi Case****Sourcecode**

```
package main

import "fmt"

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

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

### Screenshoot Output

```
PS C:\Aby Hakim\Kuliah\Semester 3\prak alpro 2>
go run "c:\Aby Hakim\Kuliah\Semester 3\prak al
pro 2\modul5\guided2\guided2.go"
Masukkan nilai n untuk penjumlahan 1 hingga n:
1 9
Hasil ppenjumlahan: 1
PS C:\Aby Hakim\Kuliah\Semester 3\prak alpro 2>
```

### Deskripsi Program

Fungsi jumlahRekursif:

- Fungsi jumlahRekursif menghitung penjumlahan dari 1 hingga n menggunakan rekursi.
- Jika n sama dengan 1, fungsi mengembalikan nilai 1.
- Jika n lebih dari 1, fungsi mengembalikan n ditambah hasil panggilan rekursif jumlahRekursif(n-1).

Fungsi utama main:

- Mendeklarasikan variabel n untuk menyimpan nilai input dari pengguna.
- Menampilkan pesan untuk meminta pengguna memasukkan nilai n.
- Membaca input pengguna dan menyimpannya di variabel n.
- Memanggil fungsi jumlahRekursif(n) dan mencetak hasil penjumlahannya.

### 3. Soal Studi Case

#### Sourcecode

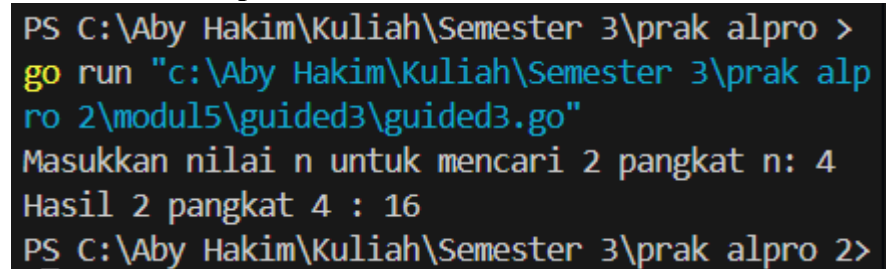
```
package main

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:\Aby Hakim\Kuliah\Semester 3\prak alpro >
go run "c:\Aby Hakim\Kuliah\Semester 3\prak alpro 2\modul5\guided3\guided3.go"
Masukkan nilai n untuk mencari 2 pangkat n: 4
Hasil 2 pangkat 4 : 16
PS C:\Aby Hakim\Kuliah\Semester 3\prak alpro 2>
```

#### Deskripsi Program

Fungsi pangkatDua:

- Fungsi pangkatDua menghitung 2 pangkat n menggunakan rekursi.
- Jika n sama dengan 0, fungsi mengembalikan nilai 1 (karena  $2^0 = 1$ ).
- Jika n lebih dari 0, fungsi mengembalikan 2 dikali hasil panggilan rekursif pangkatDua(n-1).

Fungsi utama main:

- Mendeklarasikan variabel n untuk menyimpan nilai input dari pengguna.
- Menampilkan pesan untuk meminta pengguna memasukkan nilai n.
- Membaca input pengguna dan menyimpannya di variabel n.

- Memanggil fungsi pangkatDua(n) dan mencetak hasil perhitungan 2 pangkat n.

#### 4. Soal Studi Case

##### 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:\Aby Hakim\Kuliah\Semester 3\prak alpro >
go run "c:\Aby Hakim\Kuliah\Semester 3\prak alpro 2\modul5\guided4\guided4.go"
Masukkan nilai n untuk mencari faktorial n!: 5
Hasil faktorial dari 5 : 120
PS C:\Aby Hakim\Kuliah\Semester 3\prak alpro 2>
```

##### Deskripsi Program

Fungsi faktorial:

- Fungsi faktorial menghitung faktorial dari n menggunakan rekursi.
- Jika n sama dengan 0 atau 1, fungsi mengembalikan nilai 1 (karena  $0! = 1$  dan  $1! = 1$ ).

- Jika n lebih dari 1, fungsi mengembalikan n dikalikan hasil panggilan rekursif faktorial(n-1).

Fungsi utama main:

- Mendeklarasikan variabel n untuk menyimpan nilai input dari pengguna.
- Menampilkan pesan untuk meminta pengguna memasukkan nilai n.
- Membaca input pengguna dan menyimpannya di variabel n.
- Memanggil fungsi faktorial(n) dan mencetak hasil faktorial dari n.

### III. UNGUIDED

#### 1. Soal Studi Case

##### Sourcecode

```
package main

import "fmt"

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

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

##### Screenshoot Output

```
PS C:\Aby Hakim\Kuliah\Semester 3\prak alpro >
go run "c:\Aby Hakim\Kuliah\Semester 3\prak alp
ro 2\modul5\unguided1\unguided1.go"
Masukan nilai n untuk mencari faktorial n: 4
Hasil faktorial dari 4 : 24
PS C:\Aby Hakim\Kuliah\Semester 3\prak alpro 2>
```



## Deskripsi Program

Fungsi faktorial:

- Fungsi faktorial menghitung faktorial dari n menggunakan rekursi.
- Jika n sama dengan 0 atau 1, fungsi mengembalikan nilai 1 (karena  $0! = 1$  dan  $1! = 1$ ).
- Jika n lebih dari 1, fungsi mengembalikan n dikalikan hasil panggilan rekursif faktorial(n-1).

Fungsi utama main:

- Menampilkan nama "Aby Hakim Al Yasiry Faozi".
- Mendeklarasikan variabel n untuk menyimpan nilai input dari pengguna.
- Menampilkan pesan untuk meminta pengguna memasukkan nilai n.
- Membaca input pengguna dan menyimpannya di variabel n.
- Memanggil fungsi faktorial(n) dan mencetak hasil faktorial dari n.

## 2. Soal Studi Case

### Sourcecode

```
package main

import "fmt"

// Procedure untuk mencetak pola bintang secara rekursif
func pola(i, N int, s string) {
    if i <= N {
        s = "*" + s
        fmt.Println(s)
        pola(i+1, N, s)
    }
}

func main() {
    var N int
    fmt.Print("Masukkan jumlah baris N untuk pola bintang: ")
    fmt.Scanln(&N)
    fmt.Println("Hasil Pola Bintang:")
    pola(1, N, "")
}
```

## Screenshoot Output

```
PS C:\Aby Hakim\Kuliah\Semester 3\prak alpro 2>
go run "c:\Aby Hakim\Kuliah\Semester 3\prak al
pro 2\modul5\unguided2\tempCodeRunnerFile.go"
Masukkan jumlah baris N untuk pola bintang: 4
Hasil Pola Bintang:
*
**
***
****
PS C:\Aby Hakim\Kuliah\Semester 3\prak alpro 2>
```

## Deskripsi Program

Fungsi pola:

- Fungsi pola digunakan untuk mencetak pola bintang secara rekursif.
- Fungsi ini menerima tiga parameter: i (nilai awal baris), N (jumlah baris), dan s (string yang menyimpan pola bintang).
- Jika i kurang dari atau sama dengan N, string s akan ditambahkan satu bintang di depannya ( $s = "*" + s$ ), kemudian mencetak string s, dan memanggil fungsi pola dengan nilai i yang ditambah 1.

Fungsi utama main:

- Mendeklarasikan variabel N untuk menyimpan nilai input dari pengguna yang menentukan jumlah baris untuk pola bintang.
- Menampilkan pesan untuk meminta pengguna memasukkan nilai N.
- Membaca input pengguna dan menyimpannya di variabel N.
- Menampilkan pesan "Hasil Pola Bintang:".
- Memanggil fungsi pola dengan parameter awal 1, N, dan string kosong untuk mulai mencetak pola bintang.

### 3. Soal Studi Case

#### Sourcecode

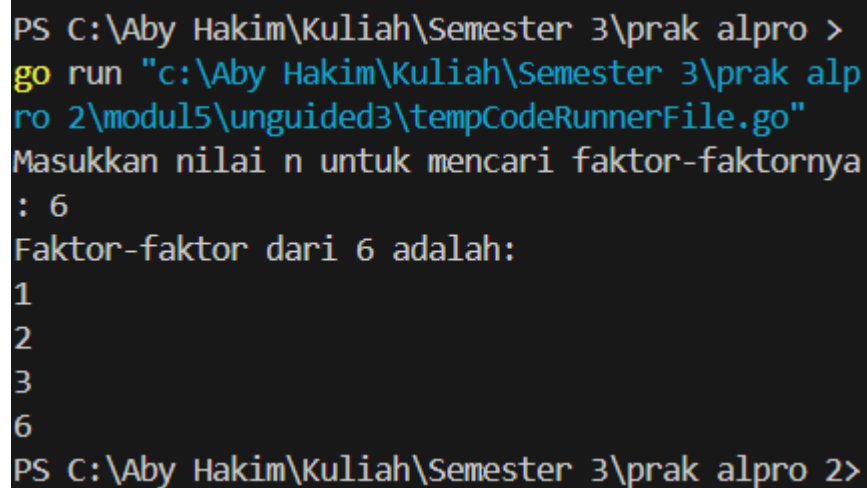
```
package main

import "fmt"

func faktor(n, i int) {
    if i <= n {
        if n%i == 0 {
            fmt.Println(i)
        }
        faktor(n, i+1)
    }
}

func main() {
    var n int
    fmt.Print("Masukkan nilai n untuk mencari faktor-faktornya: ")
    fmt.Scanln(&n)
    fmt.Println("Faktor-faktor dari", n, "adalah:")
    faktor(n, 1)
}
```

#### Screenshoot Output



```
PS C:\Aby Hakim\Kuliah\Semester 3\prak alpro >
go run "c:\Aby Hakim\Kuliah\Semester 3\prak alpro 2\modul5\unguided3\tempCodeRunnerFile.go"
Masukkan nilai n untuk mencari faktor-faktornya
: 6
Faktor-faktor dari 6 adalah:
1
2
3
6
PS C:\Aby Hakim\Kuliah\Semester 3\prak alpro 2>
```

#### Deskripsi Program

Fungsi faktor:

- Fungsi faktor digunakan untuk mencetak semua faktor dari sebuah bilangan n.

- Fungsi ini menerima dua parameter:  $n$  (bilangan yang faktor-faktornya akan dicari) dan  $i$  (nilai yang digunakan sebagai pembagi).
- Fungsi ini memeriksa apakah  $i$  (dimulai dari 1) adalah faktor dari  $n$  dengan mengecek apakah  $n \% i == 0$ .
- Jika  $i$  adalah faktor dari  $n$ , maka  $i$  dicetak.
- Fungsi kemudian memanggil dirinya sendiri dengan nilai  $i$  yang ditambah 1 untuk melanjutkan pencarian faktor berikutnya.

Fungsi utama main:

- Mendeklarasikan variabel  $n$  untuk menyimpan nilai input dari pengguna.
- Menampilkan pesan untuk meminta pengguna memasukkan nilai  $n$ .
- Membaca input pengguna dan menyimpannya di variabel  $n$ .
- Menampilkan pesan "Faktor-faktor dari  $n$  adalah:".
- Memanggil fungsi `faktor( $n$ , 1)` untuk mulai mencetak faktor-faktor dari  $n$ .

#### 4. Soal Studi Case

##### Sourcecode

```
package main

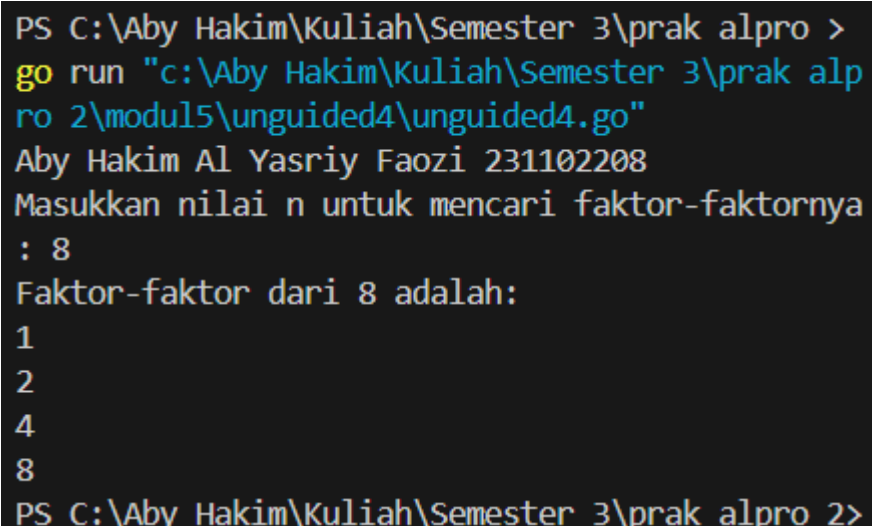
import "fmt"

func faktor(n, i int) {
    if i <= n {
        if n%i == 0 {
            fmt.Println(i)
        }

        faktor(n, i+1)
    }
}

func main() {
    var n int
    fmt.Print("Masukkan nilai n untuk mencari faktor-faktornya: ")
    fmt.Scanln(&n)
    fmt.Println("Faktor-faktor dari", n, "adalah:")
    faktor(n, 1)
}
```

##### Screenshoot Output



```
PS C:\Aby Hakim\Kuliah\Semester 3\prak alpro >
go run "c:\Aby Hakim\Kuliah\Semester 3\prak alpro 2\modul5\unguided4\unguided4.go"
Aby Hakim Al Yasriy Faozi 231102208
Masukkan nilai n untuk mencari faktor-faktornya
: 8
Faktor-faktor dari 8 adalah:
1
2
4
8
PS C:\Aby Hakim\Kuliah\Semester 3\prak alpro 2>
```

## Deskripsi Program

Fungsi faktor:

- Fungsi faktor digunakan untuk mencetak semua faktor dari sebuah bilangan n.
- Fungsi ini menerima dua parameter: n (bilangan yang faktor-faktornya akan dicari) dan i (nilai yang digunakan sebagai pembagi).
- Fungsi ini memeriksa apakah i (dimulai dari 1) adalah faktor dari n dengan mengecek apakah  $n \% i == 0$ .
- Jika i adalah faktor dari n, maka i dicetak.
- Fungsi kemudian memanggil dirinya sendiri dengan nilai i yang ditambah 1 untuk melanjutkan pencarian faktor berikutnya.

Fungsi utama main:

- Mendeklarasikan variabel n untuk menyimpan nilai input dari pengguna.
- Menampilkan pesan untuk meminta pengguna memasukkan nilai n.
- Membaca input pengguna dan menyimpannya di variabel n.
- Menampilkan pesan "Faktor-faktor dari n adalah:".
- Memanggil fungsi faktor(n, 1) untuk mulai mencetak faktor-faktor dari n.

## 5. Soal Studi Case

### Sourcecode

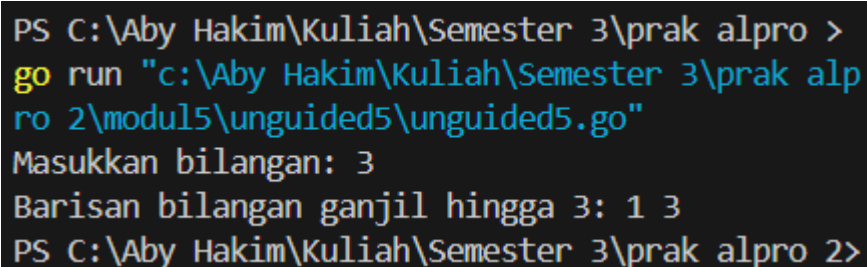
```
package main

import "fmt"

func printOddNumbers(n int) {
    if n < 1 {
        return
    }
    printOddNumbers(n - 2)
    fmt.Print(n, " ")
}

func main() {
    var number int
    fmt.Print("Masukkan bilangan: ")
    fmt.Scan(&number)
    fmt.Printf("Barisan bilangan ganjil hingga %d: ",
number)
    if number%2 == 0 {
        number--
    }
    printOddNumbers(number)
    fmt.Println()
}
```

### Screenshoot Output



```
PS C:\Aby Hakim\Kuliah\Semester 3\prak alpro >
go run "c:\Aby Hakim\Kuliah\Semester 3\prak alpro 2\modul5\unguided5\unguided5.go"
Masukkan bilangan: 3
Barisan bilangan ganjil hingga 3: 1 3
PS C:\Aby Hakim\Kuliah\Semester 3\prak alpro 2>
```

### Deskripsi Program

Fungsi printOddNumbers:

- Fungsi printOddNumbers digunakan untuk mencetak bilangan ganjil dari 1 hingga n secara rekursif.
- Fungsi ini memeriksa apakah n kurang dari 1. Jika iya, fungsi berhenti (return).
- Jika tidak, fungsi memanggil dirinya sendiri dengan n - 2 dan kemudian mencetak nilai n.

Fungsi utama main:

- Mendeklarasikan variabel number untuk menyimpan nilai input dari pengguna.
- Menampilkan pesan untuk meminta pengguna memasukkan bilangan.
- Membaca input pengguna dan menyimpannya di variabel number.
- Menampilkan pesan "Barisan bilangan ganjil hingga number".
- Jika number adalah bilangan genap, dikurangi 1 untuk memastikan mulai dari bilangan ganjil terbesar yang kurang dari atau sama dengan number.
- Memanggil fungsi printOddNumbers untuk mencetak bilangan ganjil dari 1 hingga number.

## 6. Soal Studi Case

### Sourcecode

```
package main

import "fmt"

func power(base, exponent int) int {
    if exponent == 0 {
        return 1
    } else if exponent < 0 {
        return 1 / power(base, -exponent)
    } else {
        return base * power(base, exponent-1)
    }
}

func main() {
    var base, exponent int
    fmt.Print("Masukkan bilangan pokok: ")
    fmt.Scan(&base)
    fmt.Print("Masukkan pangkat: ")
    fmt.Scan(&exponent)

    result := power(base, exponent)
    fmt.Printf("%d pangkat %d adalah %d\n", base,
exponent, result)
}
```



### Screenshoot Output

```
akim\Kuliah\Semester 3\prak alpro 2> go run "c:\Aby Hakim\Kuliah\Semester 3\prak alpro 2\modul5\unguided6\tempCodeRunnerFile.go"
Masukkan bilangan pokok: 2
Masukkan pangkat: 3
2 pangkat 3 adalah 8
PS C:\Aby Hakim\Kuliah\Semester 3\prak alpro 2> |
```

### Deskripsi Program

#### Fungsi power:

- Fungsi power menghitung pangkat dari sebuah bilangan base dengan eksponen exponent menggunakan rekursi.
- Jika exponent sama dengan 0, fungsi mengembalikan nilai 1 (karena setiap bilangan pangkat 0 adalah 1).
- Jika exponent kurang dari 0, fungsi mengembalikan hasil kebalikan dari pangkat positif (yaitu  $1 / \text{power}(\text{base}, -\text{exponent})$ ).
- Jika exponent lebih dari 0, fungsi mengembalikan base dikalikan dengan hasil panggilan rekursif  $\text{power}(\text{base}, \text{exponent}-1)$ .

#### Fungsi utama main:

- Mendeklarasikan variabel base dan exponent untuk menyimpan nilai input dari pengguna.
- Menampilkan pesan untuk meminta pengguna memasukkan nilai base.
- Membaca input pengguna dan menyimpannya di variabel base.
- Menampilkan pesan untuk meminta pengguna memasukkan nilai exponent.
- Membaca input pengguna dan menyimpannya di variabel exponent.
- Memanggil fungsi  $\text{power}(\text{base}, \text{exponent})$  dan menyimpan hasilnya di variabel result.
- Mencetak hasil pangkat dari base dengan eksponen exponent.