

**LAPORAN PRAKTIKUM  
ALGORITMA DAN PEMROGRAMAN 2**

**MODUL 4**

**Prosedur**



Oleh:

Achmad Zulvan Nur Hakim

103112400070

IF-12-01

**S1 TEKNIK INFORMATIKA  
TELKOM UNIVERSITY PURWOKERTO**

**2025**

## **I. DASAR TEORI**

Prosedur dapat dianggap sebagai potongan beberapa instruksi program menjadi suatu instruksi baru yang dibuat untuk mengurangi kerumitan dari kode program yang kompleks pada suatu program yang besar. Prosedur akan menghasilkan suatu akibat atau efek langsung pada program ketika dipanggil pada program utama. Suatu subprogram dikatakan prosedur apabila:

1. Tidak ada deklarasi tipe nilai yang dikembalikan, dan
2. Tidak terdapat kata kunci return dalam badan subprogram.

Kedudukannya prosedur sama seperti instruksi dasar yang sudah ada sebelumnya (assignment) dan/atau instruksi yang berasal dari paket (fmt), seperti fmt.Scan dan fmt.Print. Karena itu selalu pilih nama prosedur yang berbentuk kata kerja atau sesuatu yang merepresentasikan proses sebagai nama dari prosedur.

## II. GUIDED

Code 1:

```
package main

import "fmt"

//prosedur untuk menghitung total gaji karyawan
func hitungGaji(nama string, gajiPokok float64, jamLembur int) {
    bonusLembur := float64(jamLembur) * 50000
    totalGaji := gajiPokok * bonusLembur

    fmt.Println("\n==== Slip Gaji ====")
    fmt.Println("Nama Karyawan :", nama)
    fmt.Printf("Gaji Pokok : Rp%.2f\n", gajiPokok)
    fmt.Printf("Bonus Lembur : Rp%.2f (%d x Rp50.000)\n",
bonusLembur, jamLembur)
    fmt.Printf("Total Gaji : Rp%.2f\n", totalGaji)
}

func main() {
    var nama string
    var gajiPokok float64
    var jamLembur int

    //input dari pengguna
    fmt.Print("masukkan nama karyawan: ")
    fmt.Scanln(&nama)

    fmt.Print("masukkan gaji pokok: ")
    fmt.Scanln(&gajiPokok)

    fmt.Print("masukkan jumlah jam lembur: ")
    fmt.Scanln(&jamLembur)

    //memanggil prosedur dengan data dari pengguna
    hitungGaji(nama, gajiPokok, jamLembur)
}
```

Output:

```
PS D:\1031124000\0> go run "d:\1031124000\0\Modul4\Guided\1\1.g
masukkan nama karyawan: Ahmad
masukkan gaji pokok: 5000000
masukkan jumlah jam lembur: 4

=== Slip Gaji ===
Nama Karyawan : Ahmad
Gaji Pokok : Rp5000000.00
Bonus Lembur : Rp200000.00 (4 x Rp50.000)
Total Gaji : Rp10000000000.00
```

Penjelasan:

Program ini menghitung total gaji karyawan berdasarkan gaji pokok dan jumlah jam lembur, dengan bonus Rp50.000 per jam lembur. Hasilnya ditampilkan dalam bentuk slip gaji yang mencakup nama karyawan, gaji pokok, bonus lembur, dan total gaji.

Code 2:

```
package main

import "fmt"

func hitungRataRata(nama string, nilai1, nilai2, nilai3 float64) {

    ratarata := (nilai1 + nilai2 + nilai3) / 3
    status := "Tidak Lulus"

    if ratarata >= 60 {
        status = "Lulus"
    }

    fmt.Println("\n=== Hasil Akademik ===")
    fmt.Println("Nama Mahasiswa :", nama)
    fmt.Printf("Nilai 1          : %.2f\n", nilai1)
    fmt.Printf("Nilai 2          : %.2f\n", nilai2)
    fmt.Printf("Nilai 3          : %.2f\n", nilai3)
    fmt.Printf("Rata-rata   : %.2f\n", ratarata)
    fmt.Println("Status      :", status)
}

func main() {
    var nama string
    var nilai1, nilai2, nilai3 float64

    fmt.Print("Masukkan Nama Mahasiswa: ")
    fmt.Scanln(&nama)

    fmt.Print("Masukkan Nilai 1: ")
    fmt.Scanln(&nilai1)

    fmt.Print("Masukkan Nilai 2: ")
    fmt.Scanln(&nilai2)

    fmt.Print("Masukkan Nilai 3: ")
    fmt.Scanln(&nilai3)

    hitungRataRata(nama, nilai1, nilai2, nilai3)
}
```

Output:

```
PS D:\103112400070> go run "d:\103112400070\103112400070_Modul4
Masukkan Nama Mahasiswa: Ahmad
Masukkan Nilai 1: 95.00
Masukkan Nilai 2: 97.00
Masukkan Nilai 3: 96.00

=== Hasil Akademik ===
Nama Mahasiswa : Ahmad
Nilai 1       : 95.00
Nilai 2       : 97.00
Nilai 3       : 96.00
Rata-rata     : 96.00
Status        : Lulus
```

Penjelasan:

Program ini digunakan untuk menghitung rata-rata tiga nilai mahasiswa dan menentukan status kelulusan berdasarkan batas nilai 60.

### III. UNGUIDED

Code 1:

```
//Achmad Zulvan Nur Hakim 103112400070

package main

import "fmt"

func Faktorial(n int) int {
    if n == 0 {
        return 1
    }
    result := 1
    for i := 1; i <= n; i++ {
        result *= i
    }
    return result
}

func Permutasi(n, r int) int {
    return Faktorial(n) / Faktorial(n-r)
}

func Kombinasi(n, r int) int {
    return Faktorial(n) / (Faktorial(r) * Faktorial(n-r))
}

func main() {
    var a, b, c, d int

    fmt.Scan(&a, &b, &c, &d)

    if a < c || b < d {
        return
    }
    permA := Permutasi(a, c)
    kombA := Kombinasi(a, c)
    permB := Permutasi(b, d)
    kombB := Kombinasi(b, d)

    fmt.Println(permA, kombA)
    fmt.Println(permB, kombB)
}
```

Output:

```
PS D:\103112400070> go run "d:\103112400070\103112400070_Modul4\Unguided\1\1.go"
5 10 3 10
60 10
3628800 1
PS D:\103112400070> go run "d:\103112400070\103112400070_Modul4\Unguided\1\1.go"
8 0 2 0
56 28
1 1
```

Penjelasan:

Program ini digunakan untuk menghitung dan mencetak permutasi serta kombinasi dari dua pasangan bilangan input dengan faktorial.



Code 2:

```
// Achmad Zulvan Nur Hakim 103112400070
package main
import "fmt"

func main() {
    var maxDurasi = 301
    var pemenang string
    var maxSoal, minWaktu int
    maxSoal = 0
    minWaktu = maxDurasi * 8

    for {
        var peserta string
        var durasi [8]int
        fmt.Scan(&peserta)

        if peserta == "Selesai" {
            break
        }
        for i := 0; i < 8; i++ {
            fmt.Scan(&durasi[i])
        }
        var soalTerjawab, totalDurasi int
        hitungNilai(durasi, &soalTerjawab, &totalDurasi, maxDurasi)

        if soalTerjawab > maxSoal || (soalTerjawab == maxSoal && totalDurasi <
minWaktu) {
            pemenang = peserta
            maxSoal = soalTerjawab
            minWaktu = totalDurasi
        }
    }
    fmt.Println(pemenang, maxSoal, minWaktu)
}

func hitungNilai(durasi [8]int, soal *int, total *int, maxDurasi int) {
    *soal = 0
    *total = 0
    for _, waktu := range durasi {
        if waktu < maxDurasi {
            *soal++
            *total += waktu
        }
    }
}
```

```
PS D:\103112400070> go run "d:\103112400070\103112400070_Modul4\Unguided\2\2.  
Astuti 20 50 301 301 61 71 75 10  
Bertha 25 47 301 26 50 60 65 21  
Selesai  
Bertha 7 294
```

Penjelasan:

Program ini digunakan untuk mencari pemenang dari suatu kompetisi dengan syarat peserta yang berhasil menyelesaikan soal paling banyak dalam waktu paling singkat adalah pemenangnya.

Code 3:

```
//Achmad Zulvan Nur Hakim 103112400070
package main

import "fmt"

func cetakDeret(n int) {
    for {
        fmt.Print(n, " ")
        if n == 1 {
            break
        }
        if n%2 == 0 {
            n /= 2
        } else {
            n = 3*n + 1
        }
    }
    fmt.Println()
}

func main() {
    var n int
    fmt.Scan(&n)

    cetakDeret(n)
}
```

Output:

```
PS D:\103112400070> go run "d:\103112400070\103112400070_Modul14\
22
22 11 34 17 52 26 13 40 20 10 5 16 8 4 2 1
```

Penjelasan:

Program ini di gunakan untuk mencetak deret bilangan yang di mana jika bilangan genap dibagi 2, dan jika ganjil dikalikan 3 lalu ditambah 1, hingga mencapai nilai 1.

#### **IV. KESIMPULAN**

Praktikum ini menunjukkan bahwa penggunaan prosedur dalam pemrograman membantu menyusun kode yang lebih modular, terstruktur, dan mudah dipahami. Dengan membagi program menjadi prosedur-prosedur yang spesifik, kita dapat meningkatkan efisiensi, mengurangi redundansi kode, dan mempermudah proses debugging serta pengembangan program lebih lanjut.

## **V. REFERENSI**

*MODUL 4. Prosedur Algoritma dan Pemrograman 2*