

**LAPORAN PRAKTIKUM
ALGORITMA DAN PEMROGRAMAN 2**

MODUL 4

MATERI



Oleh:

ABISAR FATHIR

103112400068

12-IF-01

**S1 TEKNIK INFORMATIKA
TELKOM UNIVERSITY PURWOKERTO**

2025

I. DASAR TEORI

Prosedur adalah bagian dari kode program yang dirancang untuk menjalankan tugas tertentu tanpa mengembalikan nilai. Penggunaannya membantu menyederhanakan program dengan memisahkan logika ke dalam bagian yang lebih kecil. Dengan prosedur, kode menjadi lebih mudah dibaca, dikelola, dan diperbaiki.

Dalam kompetisi pemrograman, setiap peserta diberikan beberapa soal dengan batasan waktu tertentu. Penyelesaian soal dihitung berdasarkan waktu yang digunakan oleh peserta. Jika peserta tidak menyelesaikan soal atau melewati batas waktu, maka soal tersebut dianggap tidak berhasil dikerjakan.

Pemenang ditentukan berdasarkan jumlah soal yang berhasil diselesaikan. Jika ada peserta yang menyelesaikan jumlah soal yang sama, maka peserta dengan waktu penyelesaian paling cepat akan menjadi pemenang.

Prosedur dalam program ini digunakan untuk menghitung jumlah soal yang berhasil diselesaikan oleh peserta serta total waktu yang dihabiskan. Dengan pemisahan ini, program utama hanya berfokus pada membaca data dan menentukan pemenang, sementara logika perhitungan dilakukan dalam prosedur.

II. GUIDED

1. Program ke 1

Source Code:

```
package main

import "fmt"

func hitunggaji(nama string, gajipokok float64, jamlembur int) {

    bonusLembur := float64(jamlembur) * 5000

    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 jam x Rp.50,000)\n", bonusLembur,
jamlembur)

    fmt.Printf("total gaji: Rp. %2f\n", totalgaji)

}

func main() {

    var nama string

    var gajipokok float64

    var jamlembur int
```

```

fmt.Print("Masukkan nama karyawan:")

fmt.Scanln(&nama)

fmt.Print("Masukkan gaji Pokok:")

fmt.Scanln(&gajipokok)

fmt.Print("Masukkan jumlah jam lembur")

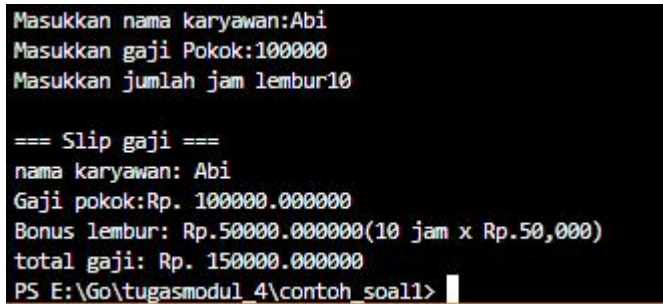
fmt.Scanln(&jamlembur)


hitunggaji(nama, gajipokok, jamlembur)

}

```

Output Program:



```

Masukkan nama karyawan:Abi
Masukkan gaji Pokok:100000
Masukkan jumlah jam lembur10

=== Slip gaji ===
nama karyawan: Abi
Gaji pokok:Rp. 100000.000000
Bonus lembur: Rp.50000.000000(10 jam x Rp.50,000)
total gaji: Rp. 150000.000000
PS E:\Go\tugasmodul_4\contoh_soal1>

```

Deskripsi Program:menghitung total gaji karyawan, termasuk gaji pokok dan bonus lembur, berdasarkan input dari pengguna. Program akan mencetak slip gaji setelah perhitungan selesai.

2. Program ke 2

Source Code:

```
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 : %2\n", nilai1)
    fmt.Printf("Nilai 2 : %.2f\n", nilai2)
    fmt.Printf("Nilai 3 : %.2f\n", nilai3)
    fmt.Printf("Rata rata : %.2f\n")
    fmt.Println("Status :", status)
}

func main() {
```

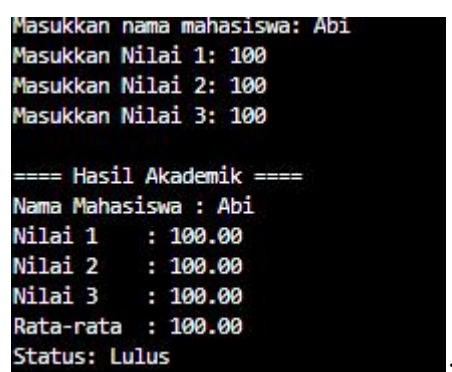
```
var nama string

var nilai1, nilai2, nilai3 float64

fmt.Print("masukkan nama mahasiswa: ")

fmt.Scanln(&nama)
```

Output Program

A screenshot of a terminal window showing the output of a Go program. The text is as follows:

```
Masukkan nama mahasiswa: Abi
Masukkan Nilai 1: 100
Masukkan Nilai 2: 100
Masukkan Nilai 3: 100

==== Hasil Akademik ====
Nama Mahasiswa : Abi
Nilai 1      : 100.00
Nilai 2      : 100.00
Nilai 3      : 100.00
Rata-rata    : 100.00
Status: Lulus
```

Deskripsi Program: Program ini digunakan untuk menghitung rata-rata nilai mahasiswa berdasarkan tiga nilai yang diberikan. Selain itu, program juga menentukan status kelulusan, di mana mahasiswa

dianggap **lulus** jika rata-rata nilai ≥ 60 dan **tidak lulus** jika rata-rata di bawah 60.

III. UNGUIDED

1. Program ke 1

Source Code:

```
package main

import (
    "fmt"
)

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

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 {
fmt.Println(permutasi(a, c), kombinasi(a, c))
fmt.Println(permutasi(b, d), kombinasi(b, d))
} else {
fmt.Println("Input tidak valid, pastikan a >= c dan b >= d")
}
}

```

Output Program:

```

5 10
3 10
60 10
3628800 1

```

Deskripsi Program: Program ini membantu menghitung Permutasi (P) dan Kombinasi (C) untuk dua pasangan bilangan (a, c) dan (b, d).

Program akan meminta pengguna memasukkan empat bilangan bulat positif a, b, c, d dengan syarat:

$a \geq c$ dan $b \geq d$

2. Program ke 2

Source Code:

```
package main

import (
    "fmt"
)

func hitungSkor(waktu [8]int) (int, int) {
    jumlahSoal, totalWaktu := 0, 0
    for _, w := range waktu {
        if w <= 300 {
            jumlahSoal++
            totalWaktu += w
        }
    }
    return jumlahSoal, totalWaktu
}
```

```
func main() {  
  
    var nama, pemenang string  
  
    var waktu [8]int  
  
    maxSoal, minWaktu := 0, 999999  
  
  
    for {  
  
        fmt.Print("Masukkan nama peserta (atau 'Selesai' untuk berhenti): ")  
  
        fmt.Scan(&nama)  
  
        if nama == "Selesai" {  
  
            break  
  
        }  
  
  
  
        for i := 0; i < 8; i++ {  
  
            fmt.Scan(&waktu[i])  
  
        }  
  
  
  
        soal, total := hitungSkor(waktu)  
  
  
  
        if soal > maxSoal || (soal == maxSoal && total < minWaktu) {  
  
            pemenang, maxSoal, minWaktu = nama, soal, total  
  
        }  
  
    }  
}
```

```

if pemenang != "" {

fmt.Println(pemenang, maxSoal, minWaktu)

} else {

fmt.Println("Tidak ada peserta yang valid.")

}

}

```

Output Program:

```

Masukkan nama peserta (atau 'Selesai' untuk berhenti): Astuti
20 50 301 301 61 71 75 10
Masukkan nama peserta (atau 'Selesai' untuk berhenti): Bertha
25 47 301 26 50 50 60 65 21
Masukkan nama peserta (atau 'Selesai' untuk berhenti): Selesai
'Selesai
Masukkan nama peserta (atau 'Selesai' untuk berhenti): Bertha 7 323

```

Deskripsi Program: Program ini adalah sebuah aplikasi sederhana untuk menentukan pemenang dari sebuah kompetisi pemrograman nasional. Setiap peserta diberikan 8 soal yang harus diselesaikan dalam waktu maksimal 5 jam (300 menit per soal). Jika peserta tidak menyelesaikan soal atau tidak mengirimkan jawaban, maka soal tersebut dihitung dengan waktu 301 menit (dianggap gagal)..

3. Program ke 3

Source Code:

```

package main

import (

"fmt"

```

```
)

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

func main() {
    var n int
    fmt.Scan(&n)
    if n > 0 && n < 1000000 {
        cetakDeret(n)
    }
}
```

Output Program:

```
22
22 11 34 17 52 26 13 40 20 10 5 16 8 4 2 1
```

Deskripsi Program:program skiena yang akan mencetak setiap suku dari deret yang dijelaskan di atas untuk

nilai suku awal yang diberikan. Pencetakan deret harus dibuat dalam prosedur cetakDeret yang

mempunyai 1 parameter formal, yaitu nilai dari suku awal.

IV. KESIMPULAN

Modul 4 membahas penggunaan prosedur, program menjadi lebih terstruktur dan mudah dipahami. Pembagian tugas antara program utama dan prosedur membuat kode lebih rapi serta mempermudah pengujian dan pemeliharaan.

Pendekatan ini memastikan bahwa program dapat menangani banyak peserta tanpa mengalami kesulitan dalam membaca dan mengelola data. Selain itu, program dapat diperluas atau dimodifikasi dengan lebih mudah sesuai kebutuhan.

V. REFERENSI