

**LAPORAN PRAKTIKUM  
ALGORITME DAN PEMROGRAMAN 2**

**MODUL 4  
PROSEDUR**



**Oleh:**

**RAIFANKA RAISA RAMADHAN**

**2311102205**

**IF - 11 - 07**

**PROGRAM STUDI S1 TEKNIK INFORMATIKA  
FAKULTAS INFORMATIKA  
INSTITUT TEKNOLOGI TELKOM PURWOKERTO  
2024**

## **I. Dasar Teori**

Di Golang, tidak ada pembedaan khusus antara prosedur dan fungsi, karena keduanya didefinisikan menggunakan kata kunci `func`. Namun, prosedur biasanya merujuk pada fungsi yang tidak mengembalikan nilai. Prosedur di Go dideklarasikan sama seperti fungsi lainnya, hanya saja tanpa tipe pengembalian. Secara umum, baik prosedur maupun fungsi di Go memiliki sintaks yang sama, dan perbedaannya hanya terletak pada apakah mereka mengembalikan nilai atau tidak.

## II. Unguided Unguided 1

```
package main

import (
    "fmt"
)

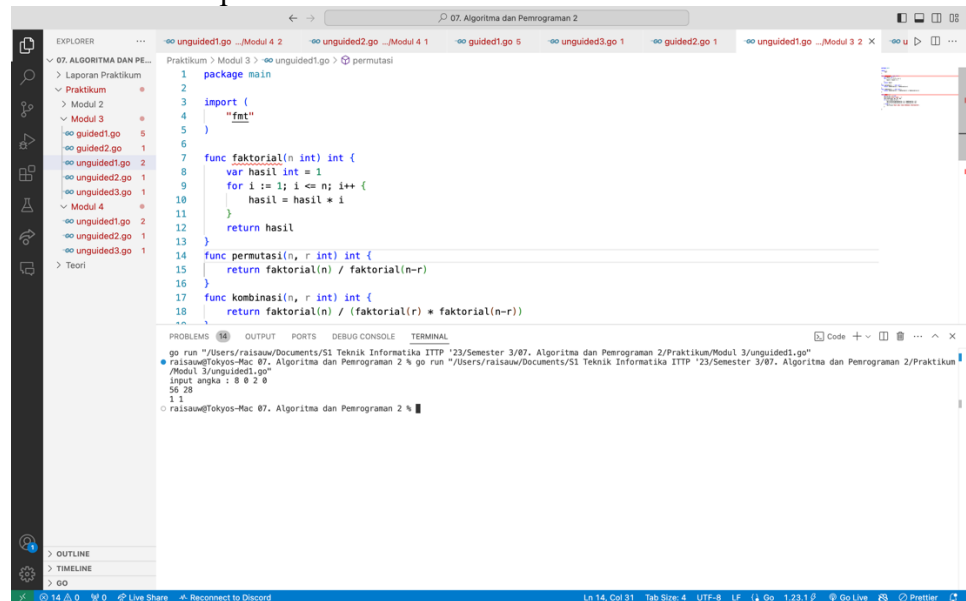
func faktorial(n int) int {
    var hasil int = 1
    for i := 1; i <= n; i++ {
        hasil = hasil * i
    }
    return hasil
}

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.Print("input angka : ")
    fmt.Scanln(&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.Print("MAAF ANDA TIDAK MEMENUHI PERSYARATAN")
    }
}
```

## Screenshots Output



The screenshot shows a Go IDE with a file explorer on the left, a code editor in the center, and a terminal at the bottom. The code editor displays a Go program with functions for factorial, permutation, and combination. The terminal shows the execution of the program with input values 8, 8, 2, and 0, resulting in the output 56 28 1 1.

```
1 package main
2
3 import (
4     "fmt"
5 )
6
7 func faktorial(n int) int {
8     var hasil int = 1
9     for i := 1; i <= n; i++ {
10         hasil = hasil * i
11     }
12     return hasil
13 }
14
15 func permutasi(n, r int) int {
16     return faktorial(n) / faktorial(n-r)
17 }
18
19 func kombinasi(n, r int) int {
20     return faktorial(n) / (faktorial(r) * faktorial(n-r))
21 }
```

PROBLEMS OUTPUT PORTS DEBUG CONSOLE TERMINAL

```
go run "/Users/raisaaw/Documents/S1 Teknik Informatika ITPP '23/Semester 3/07. Algoritma dan Pemrograman 2/Praktikum/Modul 3/unguided1.go"
raisaaw@Tokyo-Mac 07. Algoritma dan Pemrograman 2 % go run "/Users/raisaaw/Documents/S1 Teknik Informatika ITPP '23/Semester 3/07. Algoritma dan Pemrograman 2/Praktikum/Modul 3/unguided1.go"
input angka : 8 8 2 0
56 28
1 1
raisaaw@Tokyo-Mac 07. Algoritma dan Pemrograman 2 %
```

## Deskripsi:

Program ini adalah program untuk menghitung permutasi dan kombinasi dari dua pasang bilangan. Fungsi faktorial menghitung nilai faktorial dari bilangan yang diberikan. Fungsi permutasi menghitung permutasi  $P(n, r)$  menggunakan rumus  $P(n, r) = n! / (n - r)!$ , sementara fungsi kombinasi menghitung kombinasi  $C(n, r)$  dengan rumus  $C(n, r) = n! / (r!(n - r)!)$ . Di dalam fungsi main(), program menerima empat input bilangan dari pengguna (a, b, c, dan d). Jika  $a \geq c$  dan  $b \geq d$ , maka program akan menghitung dan menampilkan permutasi serta kombinasi untuk kedua pasangan bilangan tersebut. Jika tidak memenuhi syarat, program akan menampilkan pesan "MAAF ANDA TIDAK MEMENUHI PERSYARATAN".

## Unguided 2

```
package main

import "fmt"

func hitungSkor(waktu [8]int) (soal int, total int) {
    for i := 0; i < 8; i++ {
        if waktu[i] <= 300 {
            soal = soal + 1
            total = total + waktu[i]
        }
    }
    return soal, total
}

func main() {
    var nama string
    var waktu [8]int
    var pemenangNama string
    var pemenangSoal int = 0
    var pemenangWaktu int = 0

    for {
        fmt.Scanln(&nama, &waktu[0], &waktu[1], &waktu[2],
&waktu[3], &waktu[4], &waktu[5], &waktu[6], &waktu[7])

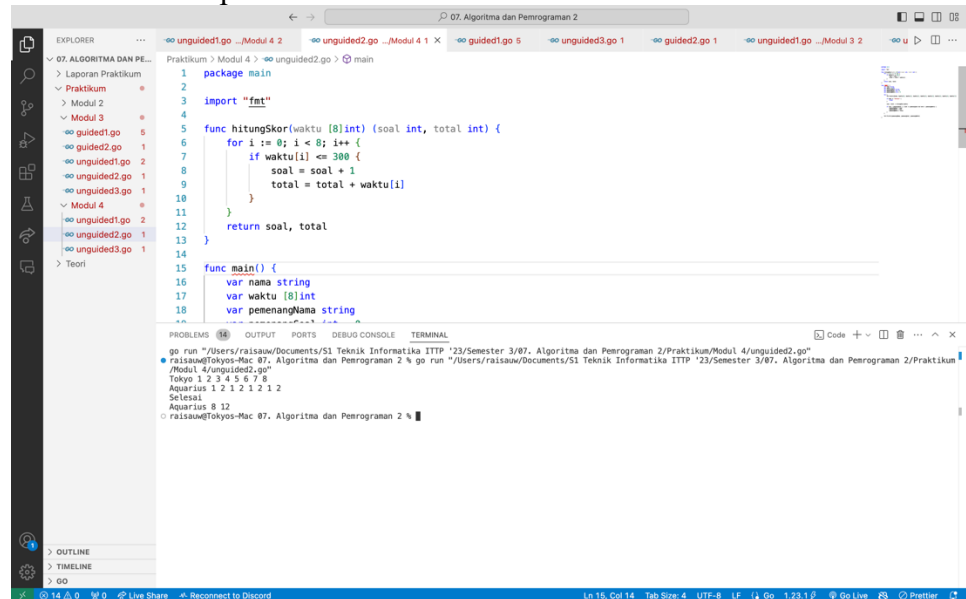
        if nama == "Selesai" {
            break
        }

        soal, total := hitungSkor(waktu)

        if soal > pemenangSoal || (soal == pemenangSoal &&
total < pemenangWaktu) {
            pemenangNama = nama
            pemenangSoal = soal
            pemenangWaktu = total
        }
    }

    fmt.Println(pemenangNama, pemenangSoal, pemenangWaktu)
}
```

## Screenshots Output



```
1 package main
2
3 import "fmt"
4
5 func hitungSkor(waktu [8]int) (soal int, total int) {
6     for i := 0; i < 8; i++ {
7         if waktu[i] <= 300 {
8             soal = soal + 1
9             total = total + waktu[i]
10        }
11    }
12    return soal, total
13 }
14
15 func main() {
16     var nama string
17     var waktu [8]int
18     var pemenangNama string
19
20     for i := 0; i < 8; i++ {
21         fmt.Print("Masukkan nama peserta ke-") + fmt.Sprint(i+1) + ": "
22         fmt.Scan(&nama)
23
24         fmt.Print("Masukkan waktu penyelesaian soal ke-") + fmt.Sprint(i+1) + ": "
25         for j := 0; j < 8; j++ {
26             fmt.Scan(&waktu[j])
27         }
28
29         soal, total := hitungSkor(waktu)
30         fmt.Println("Peserta:", nama, "Jumlah soal diselesaikan:", soal, "Total waktu:", total)
31
32         if i == 0 {
33             pemenangNama = nama
34         } else {
35             if total < waktuTotal {
36                 pemenangNama = nama
37                 waktuTotal = total
38             }
39         }
40     }
41
42     fmt.Println("Pemenang:", pemenangNama)
43 }
```

Terminal Output:

```
go run "/Users/raisaaw/Documents/S1 Teknik Informatika ITTP '23/Semester 3/07. Algoritma dan Pemrograman 2/Praktikum/Modul 4/unguided2.go"
raisaaw@Tokyo-Mac 07. Algoritma dan Pemrograman 2 % go run "/Users/raisaaw/Documents/S1 Teknik Informatika ITTP '23/Semester 3/07. Algoritma dan Pemrograman 2/Praktikum/Modul 4/unguided2.go"
Tokyo 1 2 3 4 5 6 7 8
Aquarius 1 2 1 2 1 2 1 2
Selesai
Aquarius 8 12
raisaaw@Tokyo-Mac 07. Algoritma dan Pemrograman 2 %
```

## Deskripsi:

Program ini adalah program untuk menghitung dan menentukan pemenang berdasarkan skor dari beberapa peserta dalam sebuah kompetisi. Fungsi `hitungSkor` menerima array waktu yang berisi waktu untuk menyelesaikan 8 soal. Fungsi ini menghitung jumlah soal yang diselesaikan dalam waktu 300 detik atau kurang, serta total waktu yang dihabiskan untuk soal-soal tersebut. Di dalam fungsi `main()`, program terus meminta input dari pengguna berupa nama peserta dan waktu untuk setiap soal sampai pengguna memasukkan "Selesai". Setelah setiap input, program memeriksa apakah peserta baru tersebut memiliki jumlah soal yang diselesaikan lebih banyak atau jika sama, total waktu yang lebih sedikit dibandingkan dengan pemenang sebelumnya. Jika ya, peserta tersebut menjadi pemenang baru. Setelah semua input selesai, program mencetak nama pemenang, jumlah soal yang diselesaikan, dan total waktu yang dihabiskan.

### Unguided 3

```
package main

import "fmt"

func hitungSukuBerikutnya(n int) int {
    if n%2 == 0 {
        return n / 2
    }
    return 3*n + 1
}

func cetakDeret(n int) {
    fmt.Printf("%d", n)

    for n != 1 {
        n = hitungSukuBerikutnya(n)
        fmt.Printf(" %d", n)
    }
    fmt.Println()
}

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

    if n <= 0 || n >= 1000000 {
        fmt.Println("Masukkan harus bilangan positif dan kurang
dari 1000000")
        return
    }

    cetakDeret(n)
}
```

## Screenshots Output



The screenshot shows a Go IDE with a file explorer on the left, a code editor in the center, and a terminal at the bottom. The code in the editor is as follows:

```
1 package main
2
3 import "fmt"
4
5 func hitungSukuBerikutnya(n int) int {
6     if n%2 == 0 {
7         return n / 2
8     }
9     return 3*n + 1
10 }
11
12 func cetakDeret(n int) {
13     fmt.Printf("%d", n)
14
15     for n != 1 {
16         n = hitungSukuBerikutnya(n)
17         fmt.Printf(" %d", n)
18     }
19 }
20
21 func main() {
22     n := 22
23     cetakDeret(n)
24 }
```

The terminal output shows the execution of the program:

```
go run "/Users/raisaaw/Documents/S1 Teknik Informatika ITPP '23/Semester 3/07. Algoritma dan Pemrograman 2/Praktikum/Modul 4/unguided3.go"
raisaaw@Tokyo-Mac 07. Algoritma dan Pemrograman 2 % go run "/Users/raisaaw/Documents/S1 Teknik Informatika ITPP '23/Semester 3/07. Algoritma dan Pemrograman 2/Praktikum/Modul 4/unguided3.go"
22 11 34 17 52 26 13 40 20 10 5 16 8 4 2 1
raisaaw@Tokyo-Mac 07. Algoritma dan Pemrograman 2 %
```

## Deskripsi:

Program ini adalah program yang mencetak deret bilangan. Fungsi `hitungSukuBerikutnya` menerima bilangan bulat `n` dan menentukan suku berikutnya dalam deret: jika `n` genap, suku berikutnya adalah  $n / 2$ ; jika `n` ganjil, suku berikutnya adalah  $3n + 1$ . Fungsi `cetakDeret` mencetak bilangan awal `n`, lalu terus menghitung dan mencetak suku-suku berikutnya hingga mencapai 1. Di dalam fungsi `main()`, program meminta pengguna untuk memasukkan bilangan awal, memastikan bahwa bilangan tersebut positif dan kurang dari 1.000.000. Jika input valid, program akan memanggil fungsi `cetakDeret` untuk mencetak deret dari bilangan tersebut.