

**LAPORAN PRAKTIKUM  
ALGORITMA DAN PEMROGRAMAN 4**

**MODUL 1V  
PROCEDURE**



**Disusun Oleh :**

**Shafa Adila Santoso / 2311102158**

**IF – 11 -05**

**Dosen Pengampu :**

**Arif Amrulloh, S.Kom., M.Kom.**

**PROGRAM STUDI S1 TEKNIK INFORMATIKA**

**FAKULTAS INFORMATIKA**

**TELKOM UNIVERSITY PURWOKERTO**

**2024**

## I. DASAR TEORI

### 1. Definisi Procedure

Prosedur adalah suatu blok kode program yang dirancang untuk melakukan tugas-tugas tertentu yang terkait dengan manipulasi atau pengolahan data dalam sebuah program. Prosedur biasanya berisi serangkaian instruksi atau perintah yang digunakan untuk memanipulasi data dalam suatu program, dan dapat dijalankan berkali-kali dengan input yang berbeda. Prosedur juga dapat mengambil parameter atau argumen sebagai input, melakukan operasi pada data yang diberikan, dan mengembalikan hasil atau output. Hal ini memungkinkan prosedur untuk digunakan kembali dalam berbagai bagian dari program yang berbeda-beda, sehingga memudahkan pengembangan dan pemeliharaan program secara keseluruhan. Prosedur akan menghasilkan suatu akibat atau efek langsung pada program ketika dipanggil pada program utama. Suatu subprogram dapat dikatakan sebagai prosedur ketika :

- Tidak terdapat deklarasi tipe nilai yang dikembalikan
- Tidak terdapat kata kunci return dalam badan subprogram

### 2. Deklarasi Prosedur

Notasi Algoritma	
1	procedure <nama procedure> (<params>)
2	kamus
3	{deklarasi variabel lokal dari procedure}
4	...
5	algoritma
6	{badan algoritma procedure}
7	...
8	endprocedure
Notasi dalam bahasa Go	
9	func <nama procedure> <(params)> {
10	/* deklarasi variabel lokal dari procedure */
11	...
12	/* badan algoritma procedure */
13	...
14	}

### 3. Cara Pemanggilan Procedure

Pemanggilan prosedur adalah pernyataan sederhana yang dibuat dengan menyebutkan nama prosedur, mencantumkan nama atau nilai parameter aktual dalam tanda kurung, dan menambahkan titik koma di akhir. Sebenarnya untuk pemanggilan prosedur itu cukup mudah yaitu hanya dengan menuliskan nama beserta parameter ataupun argumen yang diminta dari suatu prosedur.

### 4. Parameter

Suatu subprogram yang dipanggil dapat berkomunikasi dengan pemanggilnya melalui argumen yang diberikan melalui parameter yang

ideklarasikan pada subprogramnya. Parameter ini dibedakan menjadi 2 jenis yaitu parameter formal dan parameter aktual.

A. Parameter Formal

Parameter formal adalah variabel yang didefinisikan dalam kerangka kerja atau template sebuah fungsi atau prosedur. Parameter formal ditulis dalam prototipe fungsi dan header fungsi definisi. Parameter formal berbeda dengan parameter aktual, yaitu parameter yang digunakan untuk memanggil fungsi atau prosedur tersebut.

B. Parameter Aktual

Parameter aktual adalah variabel yang ditransfer ke dalam fungsi atau prosedur saat dipanggil. Parameter aktual harus sesuai dengan tipe data dan urutan parameter formal pada deklarasi fungsi atau prosedur yang bersangkutan

## II. GUIDED

### 1. Sourcecode

```
package main

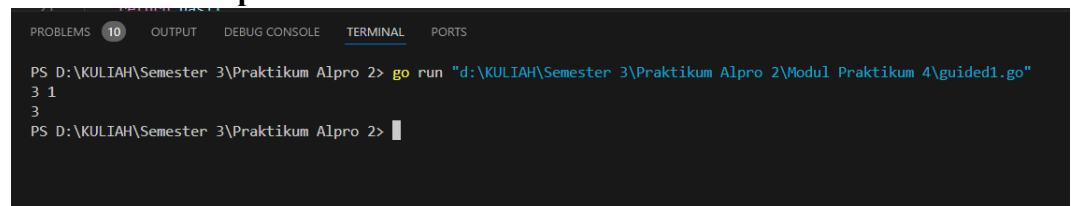
import "fmt"

func main() {
    var a, b int
    fmt.Scan(&a, &b)
    if a >= b {
        permutasi(a, b)
    } else {
        permutasi(b, a)
    }
}

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

func permutasi(n, r int) {
    hasil := faktorial(n) / faktorial(n-r)
    fmt.Println(hasil) // Mencetak hasil permutasi
}
```

### Screenshoot Output



```
PROBLEMS 10 OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS D:\KULIAH\Semester 3\Praktikum Alpro 2> go run "d:\KULIAH\Semester 3\Praktikum Alpro 2\Modul Praktikum 4\guided1.go"
3 1
3
PS D:\KULIAH\Semester 3\Praktikum Alpro 2> |
```

### Deskripsi Program

Program ini menghitung permutasi dari dua angka yang dimasukkan oleh pengguna. Pertama, pengguna memasukkan dua angka, lalu program memeriksa apakah angka pertama lebih besar atau sama dengan angka kedua. Kalau iya, program akan menghitung permutasi dengan angka pertama sebagai total elemen dan angka kedua sebagai jumlah yang dipilih. Kalau tidak, posisi kedua angka dibalik. Untuk menghitungnya, ada fungsi yang digunakan untuk mendapatkan faktorial dari sebuah

angka, lalu hasilnya dipakai buat hitung permutasi dengan rumus \yang ada disoal.

## 2. Sourcecode

```
package main

import "fmt"

// Fungsi untuk menghitung luas persegi
func hitungLuas(sisi float64) float64 {
    return sisi * sisi
}

// Fungsi untuk menghitung keliling persegi
func hitungKeliling(sisi float64) float64 {
    return 4 * sisi
}

// Prosedur untuk menghitung dan menampilkan hasil luas
dan keliling persegi
func tampilkanHasilPersegi(sisi float64) {
    luas := hitungLuas(sisi)
    keliling := hitungKeliling(sisi)

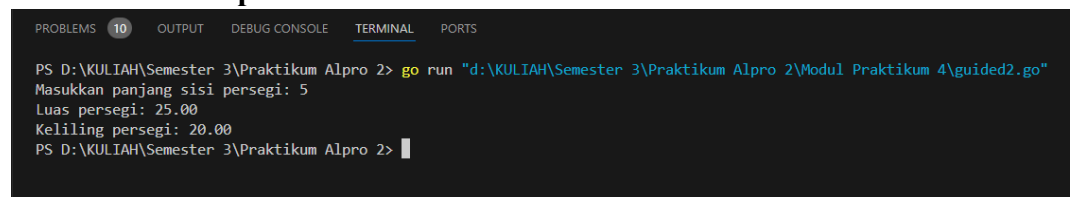
    fmt.Printf("Luas persegi: %.2f\n", luas)
    fmt.Printf("Keliling persegi: %.2f\n", keliling)
}

func main() {
    var sisi float64

    fmt.Print("Masukkan panjang sisi persegi: ")
    fmt.Scan(&sisi)

    // Memanggil prosedur untuk menampilkan hasil
    tampilkanHasilPersegi(sisi)
}
```

## Screenshoot Output



```
PROBLEMS 10 OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS D:\KULIAH\Semester 3\Praktikum Alpro 2> go run "d:\KULIAH\Semester 3\Praktikum Alpro 2\Modul Praktikum 4\guided2.go"
Masukkan panjang sisi persegi: 5
Luas persegi: 25.00
Keliling persegi: 20.00
PS D:\KULIAH\Semester 3\Praktikum Alpro 2> |
```

## Deskripsi Program

Program di atas merupakan program Go sederhana yang digunakan untuk menghitung dan menampilkan luas serta keliling dari sebuah persegi. Algoritmanya dimulai dengan meminta pengguna memasukkan panjang sisi persegi. Setelah itu, nilai sisi tersebut digunakan dalam dua fungsi: yang menghitung luas persegi dan keliling persegi. Hasil perhitungan dari kedua fungsi tersebut kemudian ditampilkan melalui prosedur di mana luas dan keliling persegi dicetak dalam format dua angka desimal. Dengan demikian, setelah pengguna memasukkan nilai sisi, program akan menghitung luas dan keliling persegi, lalu menampilkannya sebagai output.

### III. UNGUIDED

1. Minggu ini, mahasiswa Fakultas Informatika mendapatkan tugas dari mata kuliah matematika diskrit untuk mempelajari kombinasi dan permutasi. Jonas salah seorang mahasiswa, iseng untuk mengimplementasikannya ke dalam suatu program. Oleh karena itu bersediakah kalian membantu Jonas? (tidak tentunya ya :p) Masukan terdiri dari empat buah bilangan asli a, b, c, dan d yang dipisahkan oleh spasi, dengan syarat  $a \geq c$  dan  $b \geq d$ . Keluaran terdiri dari dua baris. Baris pertama adalah hasil permutasi dan kombinasi a terhadap c, sedangkan baris kedua adalah hasil permutasi dan kombinasi b terhadap d.

#### Sourcecode

```
package main

import "fmt"

// Prosedur untuk menghitung faktorial
func factorial_158(n int, result *int) {
    if n == 0 {
        *result = 1
        return
    }
    *result = 1
    for i := 1; i <= n; i++ {
        *result *= i
    }
}

// Prosedur untuk menghitung permutasi P(n, r)
func permutation_158(n, r int, result *int) {
    var factorialN, factorialNminR int
    factorial_158(n, &factorialN)
    factorial_158(n-r, &factorialNminR)
    *result = factorialN / factorialNminR
}

// Prosedur untuk menghitung kombinasi C(n, r)
func combination_158(n, r int, result *int) {
    var factorialN, factorialR, factorialNminR int
    factorial_158(n, &factorialN)
    factorial_158(r, &factorialR)
    factorial_158(n-r, &factorialNminR)
    *result = factorialN / (factorialR * factorialNminR)
}

// Prosedur untuk menampilkan hasil kombinasi dan permutasi
func hasilPerhitungan(a, b, c, d int) {
    var hasilKombinasiAC, hasilPermutasiAC int
    combination_158(a, c, &hasilKombinasiAC)
```

```

        permutation_158(a, c, &hasilPermutasiAC)

        var hasilKombinasiBD, hasilPermutasiBD int
        combination_158(b, d, &hasilKombinasiBD)
        permutation_158(b, d, &hasilPermutasiBD)

        // Menampilkan hasil
        fmt.Printf("%d %d\n", hasilPermutasiAC,
        hasilKombinasiAC)
        fmt.Printf("%d %d\n", hasilPermutasiBD,
        hasilKombinasiBD)
    }

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

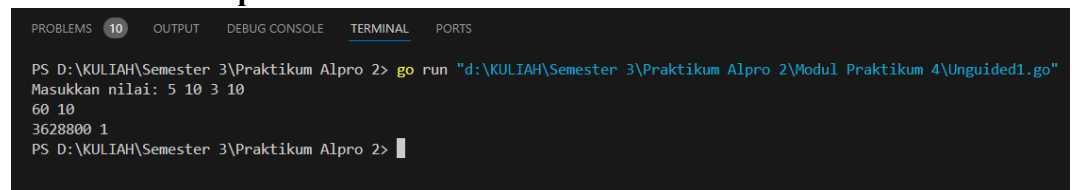
        // Input nilai n dan r
        for {
            fmt.Print("Masukkan nilai: ")
            fmt.Scanf("%d %d %d %d", &a, &b, &c, &d)

            if a >= c && b >= d {
                break
            } else {
                fmt.Println("Syarat tidak terpenuhi, coba
                lagi.")
            }
        }

        // Menampilkan hasil
        hasilPerhitungan(a, b, c, d)
    }

```

## Screenshoot Output



```

PROBLEMS 10 OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS D:\KULIAH\Semester 3\Praktikum Alpro 2> go run "d:\KULIAH\Semester 3\Praktikum Alpro 2\Modul Praktikum 4\Unguided1.go"
Masukkan nilai: 5 10 3 10
60 10
3628800 1
PS D:\KULIAH\Semester 3\Praktikum Alpro 2>

```

## Deskripsi Program

Program ini merupakan program Go sederhana yang digunakan untuk menghitung permutasi dan kombinasi dari dua pasang angka yang dimasukkan pengguna. Pengguna memasukkan empat angka (a, b, c, d), dan program memeriksa apakah nilai a lebih besar atau sama dengan c, serta b lebih besar atau sama dengan d. Jika syarat terpenuhi, program menghitung permutasi dan kombinasi untuk pasangan a, c dan b, d dengan menggunakan prosedur yang memanfaatkan perhitungan faktorial. Hasil



perhitungan permutasi dan kombinasi kemudian ditampilkan dalam dua baris.

2. Kompetisi pemrograman tingkat nasional berlangsung ketat. Setiap peserta diberikan 8 soal yang harus dapat diselesaikan dalam waktu 5 jam saja. Peserta yang berhasil menyelesaikan soal paling banyak dalam waktu paling singkat adalah pemenangnya.

Buat program gema yang mencari pemenang dari daftar peserta yang diberikan. Program harus dibuat modular, yaitu dengan membuat prosedur `hitungSkor` yang mengembalikan total soal dan total skor yang dikerjakan oleh seorang peserta, melalui parameter formal. Pembacaan nama peserta dilakukan di program utama, sedangkan waktu pengerjaan dibaca di dalam prosedur. Setiap baris masukan dimulai dengan satu string nama peserta tersebut diikuti dengan adalah 8 integer yang menyatakan berapa lama (dalam menit) peserta tersebut menyelesaikan soal. Jika tidak berhasil atau tidak mengirimkan jawaban, maka otomatis dianggap menyelesaikan dalam waktu 5 jam 1 menit (301 menit).

Satu baris keluaran berisi nama pemenang, jumlah soal yang diselesaikan, dan nilai yang diperoleh. Nilai adalah total waktu yang dibutuhkan untuk menyelesaikan soal yang berhasil diselesaikan.

### Sourcecode

```
package main

import (
    "fmt"
    "strings"
)

const batasWaktu = 301

// Fungsi untuk menghitung jumlah soal yang selesai
dikerjakan dan waktu total
func kalkulasiSkor_158(soal [8]int, jumlahSoal *int,
waktuTotal *int) {
    *jumlahSoal = 0
    *waktuTotal = 0
    for _, durasi := range soal {
        if durasi < batasWaktu {
            *jumlahSoal++
            *waktuTotal += durasi
        }
    }
}

func main() {
```

```

var peserta, juara string
var soal [8]int
var jumlahSoal, waktuTotal int
maxJumlahSoal := -1
minWaktuTotal := batasWaktu * 8 // Inisialisasi
dengan nilai maksimum yang mungkin

for {
    // Input nama peserta
    fmt.Print("Masukkan nama peserta: ")
    fmt.Scan(&peserta)
    if strings.ToLower(peserta) == "selesai" {
        break
    }

    // Input waktu pengerjaan soal
    fmt.Println("Masukkan waktu pengerjaan untuk 8
soal: ")
    for i := 0; i < 8; i++ {
        fmt.Scan(&soal[i])
    }

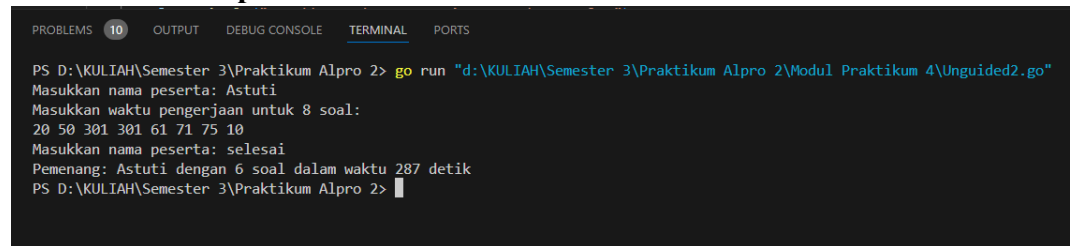
    // Menghitung jumlah soal yang dikerjakan dan
waktu totalnya
    kalkulasiSkor_158(soal, &jumlahSoal,
&waktuTotal)

    // Membandingkan hasil dengan peserta sebelumnya
untuk menentukan pemenang
    if jumlahSoal > maxJumlahSoal || (jumlahSoal ==
maxJumlahSoal && waktuTotal < minWaktuTotal) {
        maxJumlahSoal = jumlahSoal
        minWaktuTotal = waktuTotal
        juara = peserta
    }
}

// Output pemenang dengan jumlah soal yang
diselesaikan dan waktu totalnya
fmt.Printf("Pemenang: %s dengan %d soal dalam waktu
%d detik\n", juara, maxJumlahSoal, minWaktuTotal)
}

```

## Screenshoot Output

A screenshot of a terminal window showing the execution of a Go program. The terminal has tabs for PROBLEMS (10), OUTPUT, DEBUG CONSOLE, TERMINAL (selected), and PORTS. The command executed is 'go run "d:\KULIAH\Semester 3\Praktikum Alpro 2\Modul Praktikum 4\Unguided2.go"'. The program prompts for a participant's name, which is 'Astuti', and the number of questions, which is 8. It then displays a list of times: 20 50 301 301 61 71 75 10. The program prompts for a participant's name, which is 'selesai'. Finally, it displays the winner: 'Pemenang: Astuti dengan 6 soal dalam waktu 287 detik'.

## Deskripsi Program

Program ini merupakan program Go sederhana yang digunakan untuk menentukan pemenang dari beberapa peserta berdasarkan jumlah soal yang dikerjakan dan waktu total pengerjaan. Algoritmanya dimulai dengan meminta input nama peserta dan waktu yang dibutuhkan untuk mengerjakan 8 soal. Jika waktu untuk setiap soal kurang dari 301 detik, soal dianggap selesai, dan waktu tersebut ditambahkan ke total waktu pengerjaan. Program kemudian membandingkan jumlah soal yang diselesaikan dan waktu total peserta tersebut dengan peserta sebelumnya. Pemenang adalah peserta yang menyelesaikan soal terbanyak dalam waktu tercepat. Proses ini terus berulang hingga pengguna memasukkan "selesai" sebagai nama peserta, dan akhirnya program menampilkan pemenang, jumlah soal yang diselesaikan, serta waktu total pengerjaannya.

3. Skiena dan Revilla dalam Programming Challenges mendefinisikan sebuah deret bilangan. Deret dimulai dengan sebuah bilangan bulat  $n$ . Jika bilangan  $n$  saat itu genap, maka suku berikutnya adalah  $\frac{1}{2}n$ , tetapi jika ganjil maka suku berikutnya bernilai  $3n+1$ . Rumus yang sama digunakan terus menerus untuk mencari suku berikutnya. Deret berakhir ketika suku terakhir bernilai 1. Untuk suku awal sampai dengan 1000000, diketahui deret selalu mencapai suku dengan nilai 1.  
Buat 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. Lalu Masukan berupa satu bilangan integer positif yang lebih kecil dari 1000000.  
Keluaran terdiri dari satu baris saja. Setiap suku dari deret tersebut dicetak dalam baris yang sama dan dipisahkan oleh sebuah spasi.

## Sourcecode

```
package main

import "fmt"

// Prosedur untuk mencetak deret
func cetakDeret_158(n int) {
    fmt.Print(n)

    // Terus mencetak deret hingga nilainya menjadi 1
    for n != 1 {
        if n%2 == 0 {
            n = n / 2
        } else {
            n = 3*n + 1
        }
        fmt.Print(" ", n)
    }
    fmt.Println()
}

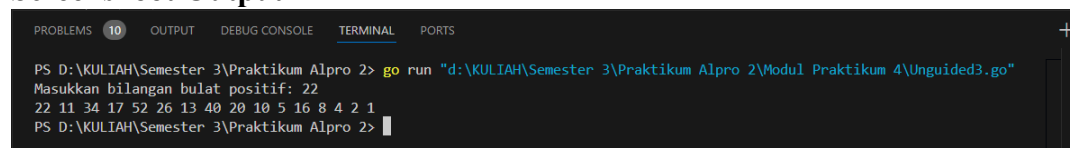
func main() {
    var n int

    // Input dari pengguna
    fmt.Print("Masukkan bilangan bulat positif: ")
    fmt.Scan(&n)

    // Validasi input agar lebih kecil dari 1000000
    if n <= 0 || n >= 1000000 {
        fmt.Println("Input harus bilangan positif lebih
kecil dari 1000000")
        return
    }

    // Panggil prosedur untuk mencetak deret
    cetakDeret_158(n)
}
```

## Screenshoot Output



```
PS D:\KULIAH\Semester 3\Praktikum Alpro 2> go run "d:\KULIAH\Semester 3\Praktikum Alpro 2\Modul Praktikum 4\Unguided3.go"
Masukkan bilangan bulat positif: 22
22 11 34 17 52 26 13 40 20 10 5 16 8 4 2 1
PS D:\KULIAH\Semester 3\Praktikum Alpro 2>
```

## Deskripsi Program

Program ini merupakan program Go sederhana untuk mencetak deret berdasarkan algoritma Collatz untuk bilangan bulat positif yang dimasukkan pengguna. Algoritmanya dimulai dengan meminta pengguna memasukkan sebuah angka positif yang harus lebih kecil dari 1.000.000. Jika input tidak memenuhi syarat tersebut, program akan menampilkan pesan kesalahan dan berhenti. Setelah menerima input yang valid, program menjalankan prosedur `cetakDeret\_158`, yang mencetak nilai awal dan kemudian melanjutkan untuk mencetak deret hingga nilai tersebut mencapai 1. Jika angka genap, program membagi angka tersebut dengan 2; jika angka ganjil, program menghitung  $(3n + 1)$ . Proses ini diulang sampai mencapai nilai 1, dan seluruh deret dicetak dalam satu baris.

## **DAFTAR PUSTAKA**

[1] Kusuma, A. A. A. (2022). PEMBAHASAN PROSEDUR DAN FUNGSI PADA PEMROGRAMAN. *Jurnal Portal Data*, 2(11).

[1] Asisten Praktikum (2024). PROSEDUR.