

**LAPORAN PRAKTIKUM  
ALGORITMA DAN PEMROGRAMAN 2**

**MODUL IV**

**PROSEDUR**



**Disusun Oleh :**

**Dimas Akal Hernanda/2311102249**

**IF-11-06**

**Dosen Pengampu :**

**ABEDNEGO DWI SEPTIADI**

**PROGRAM STUDI S1 TEKNIK INFORMATIKA**

**FAKULTAS INFORMATIKA**

**TELKOM UNIVERSITY PURWOKERTO**

**2024**

## **I. DASAR TEORI**

### Dasar Teori

Prosedur merupakan suatu dari program yang digunakan untuk menjalankan suatu tugas tertentu yang letaknya terpisah dari bagian program yang menggunakannya. Suatu prosedur digunakan dengan tujuan khusus, yaitu untuk mengerjakan suatu tugas tertentu. Keuntungan menggunakan prosedur yaitu dapat memudahkan dalam menulis dan menemukan kesalahan pada program, dan juga menghemat ukuran program.

Parameter dibagi menjadi 2 yaitu parameter formal dan actual

1. Parameter Formal Parameter formal merupakan suatu parameter yang ditulis ketika deklarasi suatu subprogram yang berfungsi sebagai petunjuk argumen
2. Parameter Aktual Parameter aktual merupakan argumen yang digunakan pada bagian parameter ketika melakukan pemanggilan subprogram

## II. GUIDED

1. Buatlah sebuah program beserta prosedur yang digunakan untuk menampilkan suatu pesan error, warning atau Informasi berdasarkan masukan dari user. Masukan terdiri dari sebuah bilangan bulat flag (0 s.d. 2) dan sebuah string pesan M. Keluaran berupa string pesan M beserta jenis pesannya, yaitu error, warning atau informasi berdasarkan nilai flag 0, 1 dan 2 secara berturut-turut.

### Sourcecode

```
package main

import (
    "fmt"
)

func main() {
    var bilangan int
    var pesan string
    fmt.Scan(&bilangan, &pesan)
    cetakPesan(pesan, bilangan)
}

func cetakPesan(x string, flag int) {
    var jenis string = ""
    if flag == 0 {
        jenis = "Error Alert!!"
    } else if flag == 1 {
        jenis = "Warning"
    } else if flag == 2 {
        jenis = "Informasi"
    }

    fmt.Println(x, jenis)
}
```

### Screenshoot Output

### Deskripsi Program

Program di atas menerima input berupa bilangan bulat (bilangan) dan string (pesan). Berdasarkan nilai bilangan, program mengategorikan pesan ke dalam tiga jenis:

Jika bilangan adalah 0, pesan di-labeli "Error Alert!!".

Jika 1, di-labeli "Warning".

Jika 2, di-labeli "Informasi".

2.

## Sourcecode

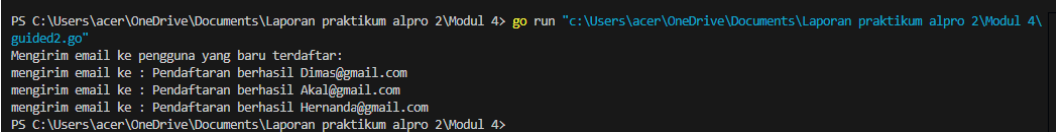
```
package main

import "fmt"

func sendEmailNotification(email string) {
    fmt.Println("mengirim email ke : Pendaftaran berhasil", email)
}

func main() {
    emails := []string{"jawir@gmail.com", "jawa@gmail.com",
"sumatran@gmail.com"}
    fmt.Println("Mengirim email ke pengguna yang baru terdaftar: ")
    for _, email := range emails {
        sendEmailNotification(email)
    }
}
```

## Screenshoot Output



```
PS C:\Users\acer\OneDrive\Documents\Laporan praktikum alpro 2\Modul 4> go run "c:\Users\acer\OneDrive\Documents\Laporan praktikum alpro 2\Modul 4\
guided2.go"
Mengirim email ke pengguna yang baru terdaftar:
mengirim email ke : Pendaftaran berhasil Dimas@gmail.com
mengirim email ke : Pendaftaran berhasil Akal@gmail.com
mengirim email ke : Pendaftaran berhasil Hernanda@gmail.com
PS C:\Users\acer\OneDrive\Documents\Laporan praktikum alpro 2\Modul 4>
```

## Deskripsi Program

Program di atas mensimulasikan pengiriman email notifikasi pendaftaran ke beberapa pengguna.

- Fungsi sendEmailNotification mencetak pesan bahwa email telah dikirim.
- Fungsi main mendefinisikan slice email, lalu menggunakan loop untuk mengirim notifikasi ke setiap alamat email dalam slice.

3. Suatu subprogram yang dipanggil dapat berkomunikasi dengan pemanggilnya melalui argumen yang diberikan melalui parameter yang dideklarasikan pada subprogramnya. Berikut ini jenis atau pembagian dari parameter. Berdasarkan letak penulisannya pada program, maka parameter dapat dikelompokkan menjadi dua, yaitu parameter formal dan parameter aktual.

### Source code

```
package main

import "fmt"

func f1(x, y int) float64 {
    var hasil float64
    hasil = float64(2*x) - 0.5*float64(y) + 3.0
    return hasil
}

func f2(x, y int, hasil *float64) {
    *hasil = float64(2*x) - 0.5*float64(y) + 3.0
}

func main() {
    var a, b int
    var c float64

    fmt.Print("Masukkan 2 bilangan : ")
    fmt.Scan(&a, &b)

    f2(a, b, &c)

    fmt.Println("Result from f2 (stored in c): ", c)

    resultF1 := f1(b, a)
    fmt.Println("Result from f1 : ", resultF1)
}
```

### Screenshoot Output

```
PS C:\Users\acer\OneDrive\Documents\Laporan praktikum alpro 2\Modul 4> go run "c:\Users\acer\OneDrive\Documents\Laporan praktikum alpro 2\Modul 4\
guided3.go"
Masukkan 2 bilangan : 4 20
Result from f2 (stored in c): 1
Result from f1 : 41
PS C:\Users\acer\OneDrive\Documents\Laporan praktikum alpro 2\Modul 4> █
```

## Deskripsi Program

Program di atas melakukan perhitungan matematika menggunakan dua fungsi: `f1` dan `f2`.

- Fungsi `f1` mengembalikan hasil perhitungan:  $2x - 0.5y + 3.0$ .
- Fungsi `f2` melakukan perhitungan yang sama tapi menyimpan hasilnya melalui pointer.

Dalam `main`, program meminta input dua bilangan, lalu:

1. Menggunakan `f2` untuk menghitung hasil dan menyimpannya di variabel `c`.
2. Menggunakan `f1` untuk menghitung dengan urutan parameter yang berbeda.

Output menampilkan hasil dari kedua fungsi.

### 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)

#### Sourcecode

```
package main

import "fmt"

// Fungsi untuk menghitung faktorial
func faktorial(n int, hasil *int) {
    *hasil = 1
    for i := 1; i <= n; i++ {
        *hasil = *hasil * i
    }
}

// Fungsi untuk menghitung permutasi
func permutasi(n, r int, hasil *int) {
    var nfaktorial, nrfaktorial int
    faktorial(n, &nfaktorial)
    faktorial(r, &nrfaktorial)
    faktorial(n-r, &nrfaktorial)
    *hasil = nfaktorial / nrfaktorial
}

// Fungsi untuk menghitung kombinasi
func kombinasi(n, r int, hasil *int) {
    var nfaktorial, rfaktorial, nrfaktorial int
    faktorial(n, &nfaktorial)
    faktorial(r, &rfaktorial)
    faktorial(n-r, &nrfaktorial)
    *hasil = nfaktorial / (rfaktorial * nrfaktorial)
}

// Fungsi utama
func main() {
    var a, b, c, d, p1, p2, c1, c2 int
    fmt.Print("Masukkan 4 bilangan: ")
}
```

```

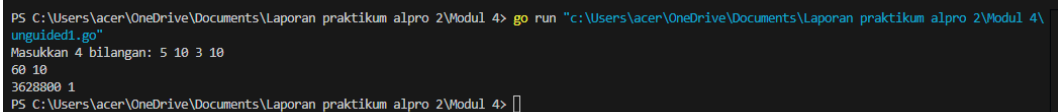
fmt.Scan(&a, &b, &c, &d)

// Menghitung permutasi dan kombinasi untuk a dan c
permutasi(a, c, &p1)
kombinasi(a, c, &c1)
fmt.Printf("%d %d\n", p1, c1)

// Menghitung permutasi dan kombinasi untuk b dan d
permutasi(b, d, &p2)
kombinasi(b, d, &c2)
fmt.Printf("%d %d\n", p2, c2)
}

```

## Screenshoot Output



```

PS C:\Users\acer\OneDrive\Documents\Laporan praktikum alpro 2\Modul 4> go run "c:\Users\acer\OneDrive\Documents\Laporan praktikum alpro 2\Modul 4\unguided1.go"
Masukkan 4 bilangan: 5 10 3 10
60 10
3628800 1
PS C:\Users\acer\OneDrive\Documents\Laporan praktikum alpro 2\Modul 4>

```

## Deskripsi Program

program ini digunakan untuk menghitung dan menampilkan hasil permutasi dan kombinasi berdasarkan input pengguna.

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.

## Sourcecode

```

package main

import (
    "fmt"
    "strings"
)

```



```

func hitungSkor(waktu []int, totalSoal *int, totalSkor *int) {
    *totalSoal = 0
    *totalSkor = 0
    for _, t := range waktu {
        if t < 301 { // Waktu penyelesaian kurang dari 301 menit
            *totalSoal++
            *totalSkor += t
        }
    }
}

func main() {
    var pemenang string
    var maxSoal, minSkor int
    maxSoal = 0
    minSkor = 999999

    for {
        var nama string
        var waktu [8]int

        fmt.Print("Masukkan nama peserta (ketik 'Selesai' untuk
menghentikan pendaftaran): ")
        fmt.Scan(&nama)

        if strings.ToLower(nama) == "selesai" {
            break
        }

        fmt.Println("Masukkan waktu penyelesaian untuk 8 soal (dalam
menit, satu per satu):")
        for i := 0; i < 8; i++ {
            _, err := fmt.Scan(&waktu[i])
            if err != nil {
                fmt.Println("Terjadi kesalahan dalam membaca input. Pastikan
Anda memasukkan angka yang benar.")
                i-- // Mengulangi pembacaan waktu untuk soal yang sama
            }
        }

        var soal, skor int
        hitungSkor(waktu[:], &soal, &skor)

        if soal > maxSoal || (soal == maxSoal && skor < minSkor) {
            pemenang = nama
        }
    }
}

```

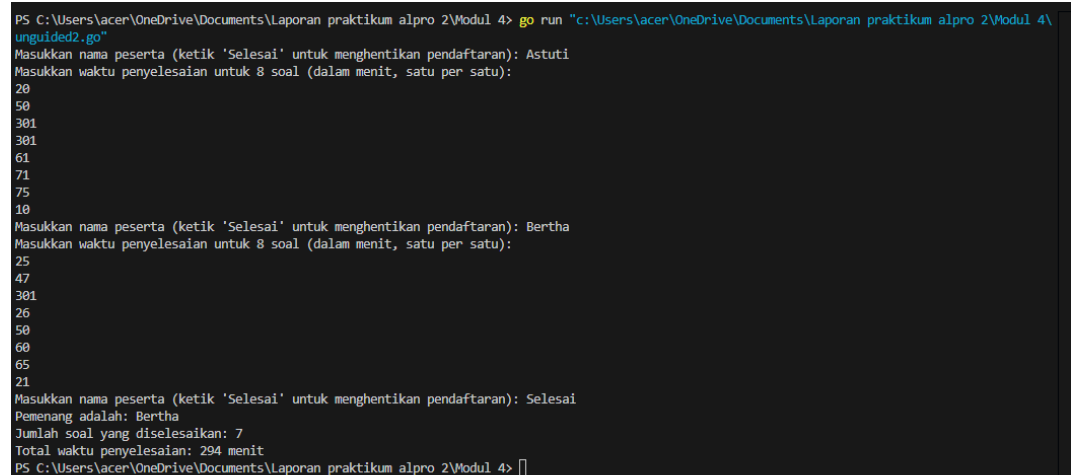
```

        maxSoal = soal
        minSkor = skor
    }
}

if pemenang != "" {
    fmt.Printf("Pemenang adalah: %s\n", pemenang)
    fmt.Printf("Jumlah soal yang diselesaikan: %d\n", maxSoal)
    fmt.Printf("Total waktu penyelesaian: %d menit\n", minSkor)
} else {
    fmt.Println("Tidak ada peserta.")
}
}

```

## Screenshoot Output



```

PS C:\Users\acer\OneDrive\Documents\Laporan praktikum alpro 2\Modul 4> go run "c:\Users\acer\OneDrive\Documents\Laporan praktikum alpro 2\Modul 4\
unguided2.go"
Masukkan nama peserta (ketik 'Selesai' untuk menghentikan pendaftaran): Astuti
Masukkan waktu penyelesaian untuk 8 soal (dalam menit, satu per satu):
20
50
301
301
61
71
75
10
Masukkan nama peserta (ketik 'Selesai' untuk menghentikan pendaftaran): Bertha
Masukkan waktu penyelesaian untuk 8 soal (dalam menit, satu per satu):
25
47
301
26
50
60
65
21
Masukkan nama peserta (ketik 'Selesai' untuk menghentikan pendaftaran): Selesai
Pemenang adalah: Bertha
Jumlah soal yang diselesaikan: 7
Total waktu penyelesaian: 294 menit
PS C:\Users\acer\OneDrive\Documents\Laporan praktikum alpro 2\Modul 4> 

```

## Deskripsi Program

Program ini digunakan untuk menentukan pemenang dalam sebuah kompetisi penyelesaian soal.

-Fungsi Utama: Mengumpulkan nama peserta dan waktu penyelesaian untuk 8 soal. Program menghitung jumlah soal yang diselesaikan dalam waktu kurang dari 301 menit dan total waktu penyelesaian.

- Pemenang: Ditentukan berdasarkan jumlah soal yang diselesaikan; jika ada peserta dengan jumlah soal yang sama, pemenang dipilih berdasarkan total waktu penyelesaian yang lebih sedikit.

- Input: Nama peserta dan waktu penyelesaian untuk setiap soal. Input berhenti jika pengguna mengetik "Selesai".
- Output: Menampilkan nama pemenang, jumlah soal yang diselesaikan, dan total waktu penyelesaian.

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  $1/2n$ , 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

### Sourcecode

```
package main

import "fmt"

func deret(n int) {
    fmt.Print(n)
    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
    fmt.Print("Masukkan sebuah bilangan positif: ")
    fmt.Scan(&n)

    // Memastikan bilangan yang dimasukkan adalah positif
    if n > 0 {
        deret(n)
    } else {
        fmt.Println("Silakan masukkan bilangan positif.")
    }
}
```

## Screenshoot Output

```
PS C:\Users\acer\OneDrive\Documents\Laporan praktikum alpro 2\Modul 4> go run "c:\Users\acer\OneDrive\Documents\Laporan praktikum alpro 2\Modul 4\unguided3.go"
Masukkan sebuah bilangan positif: 22
22 11 34 17 52 26 13 40 20 10 5 16 8 4 2 1
PS C:\Users\acer\OneDrive\Documents\Laporan praktikum alpro 2\Modul 4> 
```

## Deskripsi Program

Program tersebut menggunakan prosedur yang berfungsi untuk menghitung deret yang di inputkan oleh user, dengan deret dimulai dengan bilangan bulan  $n$ , jika  $n$  genap maka suku berikutnya setengah  $n$ , namun jika ganjil maka suku berikutnya  $3n+1$ , Rumus tersebut digunakan hingga suku berikut bernilai 1