

**LAPORAN PRAKTIKUM
ALGORITMA DAN PEMROGRAMAN 2**

**MODUL 4
PROSEDUR**



**Disusun Oleh :
Natasya Intan Sukma Jiwanti / 2311102279
S1-IF-11-05**

**Dosen Pengampu :
Arif Amrulloh, S.Kom., M.Kom.**

**PROGRAM STUDI S1 TEKNIK INFORMATIKA
FAKULTAS INFORMATIKA
TELKOM UNIVERSITY PURWOKERTO
2024**

DASAR TEORI

Prosedur hampir mirip dengan fungsi yakni merupakan sebuah sub program yang berisi kumpulan instruksi tanpa adanya pengembalian, atau kata kunci return. Selain itu, prosedur ditandai dengan tidak adanya tipe data nilai yang dikembalikan, hanya ada parameter. Prosedur dapat dideklarasikan dengan cara mengetikkan nama procedure yang diikuti oleh parameter. Prosedur harus dideklarasikan diluar fungsi utama, dan untuk memanggilnya dalam fungsi utama kita hanya perlu mendeklarasikan nama prosedur yang diikuti oleh pendeklarasian parameter informal.

Perbedaan yang utama dari pendeklarasian antara parameter formal dan informal, yakni dalam parameter formal tidak terdapat nilai yang disimpan, sedangkan parameter informal akan mempunyai nilai yang disimpan dalam pendeklarasiannya. Selain dibedakan dari cara pendeklarasiannya, parameter juga dibedakan dengan alokasi memorinya yakni, pass by value dan pass by reference (menggunakan pointer).

Pada prosedur disarankan untuk menggunakan parameter pass by reference karena dalam prosedur tidak bisa mengembalikan nilai ke pemanggil, dan apabila menggunakan parameter pass by reference maka prosedur akan seolah – olah dapat mengirimkan nilai kepada si pemanggil.

I. GUIDED

1. Guided 1

Soal Studi Case

Buatlah sebuah program yang didalamnya berisi prosedur untuk digunakan untuk menghitung nilai faktorial dan permutasi.

Sourcecode

```
package main
import "fmt"

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

func permutasi(n, r int) {
    var hasiln, hasiln_r int
    faktorial(n, &hasiln)
    faktorial(n-r, &hasiln_r)
    fmt.Println("Hasil permutasi:", hasiln/hasiln_r)
}

func main() {
    var a, b int
    fmt.Print("Masukkan Bilangan Pertama: ")
    fmt.Scan(&a)
    fmt.Print("Masukkan Bilangan Kedua: ")
    fmt.Scan(&b)
    if a >= b {
        permutasi(a, b)
    } else {
        permutasi(b, a)
    }
}
```

Screenshoot Output

```
PS D:\Praktikum Alpro\praktikum 3\modul 4> go run "d:\Praktikum Alpro\praktikum 3\modul 4\tempCodeRunnerFile.go"
Masukkan Bilangan Pertama: 10
Masukkan Bilangan Kedua: 5
Hasil permutasi: 30240
PS D:\Praktikum Alpro\praktikum 3\modul 4> █
```

Deskripsi Program

Program diatas merupakan program yang berisi dua prosedur, yakni:

- a) Prosedur Faktorial, dalam prosedur ini menggunakan parameter pass by reference untuk parameter hasil, supaya dapat dipanggil dalam prosedur permutasi.
- b) Prosedur Permutasi, dalam prosedur ini tidak digunakan parameter pass by reference, namun hanya digunakan parameter pass by value sehingga hasil dari perhitungan permutasi hanya dapat dipanggil di prosedur permutasi. Oleh karena itu, pada program diatas kita langsung mencetak nilai hasil permutasi dalam prosedur permutasi.

Pada program diatas, prosedur yang dipanggil dalam fungsi main atau fungsi utama hanyalah prosedur permutasi, karena prosedur faktorial digunakan hanya untuk perhitungan permutasi.

2. Guided 2

Soal Studi Case

Buatlah program yang berisi prosedur untuk menghitung luas dan keliling persegi!

Sourcecode

```
package main
import "fmt"

func luasPersegi(s int) {
    hasil := s * s
    fmt.Println("Luas Persegi: ", hasil)
}

func kelilingPersegi(s int) {
    hasil := 4 * s
    fmt.Print("Keliling Persegi: ", hasil)
}

func main() {
    var s int
```

```
    fmt.Print("Input Nilai Sisi: ")
    fmt.Scan(&s)

    luasPersegi(s)
    kelilingPersegi(s)
}
```

Screenshot Output

```
PS D:\Praktikum Alpro\praktikum 3\modul 4> go run "d:\Praktikum Alpro\praktikum 3\modul 4\tempCodeRunnerFile.go"
Input Nilai Sisi: 7
Luas Persegi: 49
Keliling Persegi: 28
PS D:\Praktikum Alpro\praktikum 3\modul 4> █
```

Deskripsi Program

Program diatas merupakan program yang berisikan dua buah prosedur, dimana prosedur yang pertama untuk menghitung keliling persegi dan yang kedua untuk menghitung luas persegi. Kedua prosedur tersebut menggunakan parameter pass by value, oleh karena itu pada program diatas hasil perhitungan tidak dapat dipanggil dalam fungsi utama, dan hanya dapat dipanggil dalam prosedur masing – masing sehingga pada setiap prosedur langsung menggunakan instruksi print untuk mencetak.

Sedangkan pada fungsi utama, hanya terdapat instruksi untuk menginputkan nilai sisi dan pemanggilan kedua prosedur.

I. UNGUIDED

1. Unguided 1

Soal Study Case

Minggu ini, mahasiswa fakultas informatika mendapatkan tugas dari mata kuliah matematika diskrit untuk mempelajari kombinasi dan permutasi. Jonas salah seorang mahasiswa iseng mengimplementasikannya ke dalam suatu program. Oleh karena itu bersediakah kalian membantu Jonas? Dengan ketentuan Program:

- 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.
- Gunakanlah persamaan berikut:

$$P(n, r) = \frac{n!}{(n-r)!}, \text{ sedangkan } C(n, r) = \frac{n!}{r!(n-r)!}$$

Sourcecode

```
package main
import "fmt"

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

func permutasi(n, r int){
    var hasiln, hasiln_r int
    faktorial(n, &hasiln)
    faktorial(n-r, &hasiln_r)
    fmt.Println("Hasil permutasi bilangan ",n," dan ",r,":",
    hasiln/hasiln_r)
}

func kombinatorial(n, r int){
```

```

    var faktorn, faktorn_r, faktor_r int
    faktorial(n, &faktorn)
    faktorial(n-r, &faktorn_r)
    faktorial(r, &faktor_r)
    fmt.Println("Hasil Kombinasi bilangan ",n," dan ",r,":",
faktorn/faktor_r*faktorn_r)
}

func main() {
    var a, b, c, d int
    fmt.Print("Masukkan Bilangan: ")
    fmt.Scan(&a, &b, &c, &d)
    if a >= c && b >= d {
        permutasi(a, c)
        kombinatorial(a, c)
        permutasi(b, d)
        kombinatorial(b, d)
    } else {
        fmt.Println("Kondisi Tidak Memenuhi Syarat")
    }
}

```

Screenshot Output

```

PS D:\Praktikum Alpro\praktikum 3\modul 4> go run "d:\Praktikum Alpro\praktikum 3\modul 4\unguided1.go"
Masukkan Bilangan: 10 9 8 7
Hasil permutasi bilangan 10 dan 8 : 1814400
Hasil Kombinasi bilangan 10 dan 8 : 180
Hasil permutasi bilangan 9 dan 7 : 181440
Hasil Kombinasi bilangan 9 dan 7 : 144
PS D:\Praktikum Alpro\praktikum 3\modul 4>

```

Deskripsi Program

Program diatas merupakan program yang berisi dua prosedur, yakni:

- Prosedur Faktorial, dalam prosedur ini menggunakan parameter pass by reference untuk parameter hasil, supaya dapat dipanggil dalam prosedur permutasi.
- Prosedur Permutasi, dalam prosedur ini tidak digunakan parameter pass by reference, namun hanya digunakan parameter pass by value sehingga hasil dari perhitungan permutasi hanya dapat dipanggil di prosedur permutasi. Oleh karena itu, pada program diatas kita langsung mencetak nilai hasil permutasi dalam prosedur permutasi.

- c) Prosedur kombinasi, dalam prosedur ini tidak digunakan parameter pass by reference, namun hanya digunakan parameter pass by value sehingga hasil dari perhitungan kombinasi hanya dapat dipanggil di prosedur kombinasi. Oleh karena itu, pada program diatas kita langsung mencetak nilai hasil kombinasi dalam prosedur kombinasi.

Pada program diatas, prosedur yang dipanggil dalam fungsi main atau fungsi utama hanyalah prosedur permutasi dan prosedur kombinatorial, karena prosedur faktorial digunakan hanya untuk perhitungan permutasi.

2. Unguided 2

Soal Studi Case

Buat program gema yang mencari pemenang dari daftar peserta yang diberikan. Program harus dibuat modular, yaitu dengan membuat prosedur hitungskor yang mengembalikan nilai dari parameter 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 diikuti 8 interger yang menyatakan berapa lama peserta tersebut menyelesaikan soal. Jika tidak berhasil atau tidak mengirimkan jawaban maka otomatis dianggap menyelesaikan dalam waktu 301 menit.

Sourcecode

```
package main

import (
    "fmt"
    "math"
)

type Peserta struct {
    Nama string
    Nilai [8]int
}
```



```

}

func hitungSkor(nilai [8]int, totalskor *int, totalsoal *int)
{
    *totalskor = 0
    *totalsoal = 0

    for _, n := range nilai {
        if n < 301 {
            *totalsoal++
            *totalskor += n
        }
    }
}

func cariPemenang(peserta []Peserta) Peserta {
    var pemenang Peserta
    maxsoal := -1
    minnilai := math.MaxInt32
    for _, m := range peserta {
        var totalNilai, soalDikerjakan int
        hitungSkor(m.Nilai, &totalNilai, &soalDikerjakan)

        if soalDikerjakan > maxsoal || (soalDikerjakan ==
maxsoal && totalNilai < minnilai) {
            maxsoal = soalDikerjakan
            minnilai = totalNilai
            pemenang = m
        }
    }
    return pemenang
}

func main() {
    var jumlahpeserta int
    fmt.Print("Masukkan jumlah peserta: ")
    fmt.Scan(&jumlahpeserta)
    Peserta := make([]Peserta, jumlahpeserta)
    for i := 0; i < jumlahpeserta; i++ {
        fmt.Printf("Masukkan nama peserta ke-%d: ", i+1)
        fmt.Scan(&Peserta[i].Nama)
        for j := 0; j < 8; j++ {
            fmt.Printf("Masukkan waktu pengerjaan soal ke-%d
untuk %s: ", j+1, Peserta[i].Nama)
            fmt.Scan(&Peserta[i].Nilai[j])
        }
    }
}

```

```

    }
}
pemenang := cariPemenang(Peserta)
var totalNilai, totalSoalDikerjakan int
hitungSkor(pemenang.Nilai, &totalNilai,
&totalSoalDikerjakan)

fmt.Printf("\nPemenang: %s\n", pemenang>Nama)
fmt.Printf("Total Nilai: %d\n", totalNilai)
fmt.Printf("Total Soal Dikerjakan: %d\n",
totalSoalDikerjakan)
}

```

Screenshoot Output

```

PS D:\Praktikum Alpro\praktikum 3\modul 4> go run "d:\Praktikum Alpro\praktikum 3\modul 4\ungid2.go"
Masukkan jumlah peserta: 2
Masukkan nama peserta ke-1: natasya
Masukkan waktu pengerjaan soal ke-1 untuk natasya: 79
Masukkan waktu pengerjaan soal ke-2 untuk natasya: 19
Masukkan waktu pengerjaan soal ke-3 untuk natasya: 20
Masukkan waktu pengerjaan soal ke-4 untuk natasya: 65
Masukkan waktu pengerjaan soal ke-5 untuk natasya: 301
Masukkan waktu pengerjaan soal ke-6 untuk natasya: 301
Masukkan waktu pengerjaan soal ke-7 untuk natasya: 67
Masukkan waktu pengerjaan soal ke-8 untuk natasya: 18
Masukkan nama peserta ke-2: intan
Masukkan waktu pengerjaan soal ke-1 untuk intan: 109
Masukkan waktu pengerjaan soal ke-2 untuk intan: 301
Masukkan waktu pengerjaan soal ke-3 untuk intan: 18
Masukkan waktu pengerjaan soal ke-4 untuk intan: 97
Masukkan waktu pengerjaan soal ke-5 untuk intan: 27
Masukkan waktu pengerjaan soal ke-6 untuk intan: 301
Masukkan waktu pengerjaan soal ke-7 untuk intan: 301
Masukkan waktu pengerjaan soal ke-8 untuk intan: 17

Pemenang: natasya
Total Nilai: 268
Total Soal Dikerjakan: 6
PS D:\Praktikum Alpro\praktikum 3\modul 4>

```

Deskripsi Program

Program diatas merupakan program untuk mencari pemenang, dengan memiliki prosedur dan fungsi:

- 1) Struct Peserta, berisi variabel nama bertipe data string, dan array nilai dengan jumlah isi array 8 buah bertipe data interger.
- 2) Prosedur hitungSkor

Pada prosedur hitung skor, kita akan menggunakan perulangan sejumlah isi array nilai yakni 8. Lalu untuk jumlah soal dan waktunya program hanya akan menghitung jumlah soal yang waktu penyelesaiannya kurang dari 301 menit.

3) Fungsi cariPemenang

Pada fungsi cari pemenang kita akan menggunakan perulangan dengan range atau ukuran perulangan sebanyak isi data dalam struct peserta. Setelah itu akan dilakukan pemanggilan prosedur hitung skor, untuk mengetahui total skor dan jumlah soal yang dikerjakan oleh peserta. Pemenang akan ditentukan dari jumlah soal terbanyak dan total waktu pengerjaan terkecil.

4) Fungsi Main

Pada fungsi utama program akan meminta jumlah peserta lalu menggunakan perulangan untuk mengisi struct peserta sebanyak jumlah yang diinputkan.

3. Unguided 3

Soal Studi Case

Buat program sklena yang akan mencetak setiap suku dari deret dengan ketentuan:

- a) Bilangan genap, maka n selanjutnya dirumuskan dengan $\frac{1}{2}n$
- b) Bilangan ganjil, maka n selanjutnya dirumuskan dengan $3n+1$
- c) Suku terakhir bernilai 1
- d) Suku awal deret merupakan bilangan bulat dibawah 1000000 yang diinputkan oleh user

Sourcecode

```
package main
import "fmt"

func cetakderet(n uint){
    fmt.Print(n)
    for n>1{
        if n%2 == 0{
            n = n/2
        }else{
            n = 3*n + 1
        }
        fmt.Print(" ", n)
    }
    fmt.Print()
}
```

```

}

func main(){
    var n uint
    fmt.Print("Masukkan Suku Pertama *tidak boleh melebihi
1000000: ")
    fmt.Scan (&n)
    if n >= 1 && n < 1000000 {
        cetakderet(n)
    } else {
        fmt.Println("Masukkan Suku Pertama Tidak Valid!")
    }
}

```

Screenshot Output

```

PS D:\Praktikum Alpro\praktikum 3\modul 4> go run "d:\Praktikum Alpro\praktikum 3\modul 4\unguided3.go"
Masukkan Suku Pertama *tidak boleh melebihi 1000000: 279
279 838 419 1258 629 1888 944 472 236 118 59 178 89 268 134 67 202 101 304 152 76 38 19 58 29 88 44 22 11 34 17 52 26 13 40 20 10 5 16 8 4 2 1
PS D:\Praktikum Alpro\praktikum 3\modul 4> █

```

Deskripsi Program

Program diatas merupakan program yang digunakan untuk mencari dan mencetak deret dengan suku awal merupakan bilangan bulat yang diinputkan oleh user dan memiliki suku akhir bernilai 1. Program diatas menggunakan prosedur dengan parameter formal pass by value sehingga langsung dicetak dalam prosedur.