

LAPORAN PRAKTIKUM
ALGORITMA DAN PEMROGRAMAN 2
MODUL
“PROSEDUR”



DISUSUN OLEH:
SHEILA STEPHANIE ANINDYA
103112400086
S1 IF-12-01
S1 TEKNIK INFORMATIKA
TELKOM UNIVERSITY PURWOKERTO
2025

I. DASAR TEORI

Prosedur merupakan bagian dari suatu program yang dirancang secara terpisah untuk menjalankan tugas atau fungsi tertentu.

Secara umum, terdapat dua jenis prosedur, yaitu Subrutin (subprogram) dan Fungsi:

1. **Subrutin (subprogram)** adalah bagian dari kode program yang dibuat secara terpisah untuk menyelesaikan sebagian tugas dalam sebuah program. Istilah "Prosedur" yang sering digunakan sebenarnya mengacu pada "Subprogram."
2. **Fungsi** merupakan bagian dari program yang dibuat untuk menjalankan suatu tugas tertentu dan mengembalikan nilai kepada program utama.

Prosedur sering dimanfaatkan dalam pemrograman terstruktur karena beberapa alasan berikut:

1. Menerapkan konsep pemrograman modular, yang membagi program kompleks menjadi beberapa bagian lebih kecil dalam bentuk prosedur.
2. Untuk tugas yang sering berulang, cukup dibuat satu prosedur yang dapat dipanggil kapan saja diperlukan, sehingga mengurangi duplikasi kode.
3. Membantu meningkatkan keterbacaan kode program.
4. Memungkinkan penyembunyian detail implementasi program, sehingga lebih rapi dan mudah dikelola.

Ciri-Ciri Prosedur :

1. Memiliki satu tujuan utama (logical inherent). Sebuah prosedur sebaiknya hanya berfungsi untuk satu tujuan tertentu tanpa bercampur dengan tujuan lain, sehingga tetap fokus dan mudah dipahami.
2. Berukuran kecil (small size). Ukuran yang dimaksud mengacu pada panjang algoritma atau jumlah baris kode dalam prosedur. Prosedur yang lebih ringkas lebih mudah dibaca dan diperbaiki.
3. Bersifat mandiri (independent). Sebuah prosedur idealnya dapat berjalan sendiri tanpa bergantung pada prosedur lain. Selain itu, variabel dalam prosedur sebaiknya tidak mempengaruhi variabel di bagian lain dari program.

Keuntungan menggunakan Prosedur :

1. Merupakan penerapan konsep program modular, yaitu memecah-mecah program yang rumit menjadi program-program bagian yang lebih sederhana dalam bentuk prosedur-prosedur.
2. Untuk hal-hal yang sering dilakukan berulang-ulang, cukup dituliskan sekali saja dalam prosedur dan dapat dipanggil atau dipergunakan sewaktu-waktu bila diperlukan.

II. GUIDED

1. Source Code:

```
package main

import "fmt"

//prosedur untk menghitung total gaji karyawan
func hitungGaji(nama string, gajiPokok float64, jamLembur int) {
    bonusLembur := float64(jamLembur) * 50000
    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() { // main redeclared in this block
    var nama string
    var gajiPokok float64
    var jamLembur int

    fmt.Print("Nama : ")
    fmt.Scanln(&nama)

    fmt.Print("Gaji Pokok : ")
    fmt.Scanln(&gajiPokok)
```

```
fmt.Print("Jumlah Jam Lembur : ")
fmt.Scanln(&jamLembur)

// memanggil prosedur dengan data dr inputan

hitungGaji(nama, gajiPokok, jamLembur)

}
```

Output:

```
Nama : awan
Gaji Pokok : 5000000
Jumlah Jam Lembur : 12

=== Slip Gaji ===
Nama Karyawan : awan
Gaji Pokok : Rp 5000000.00
Bonus Lembur : Rp 600000.00 (12 jam x Rp 50.000)
Total Gaji : Rp 5600000.00
PS C:\Sheila Stephanie Anindya\SMT 2\Alpro 2\code> |
```

Penjelasan :

Program ini menghitung slip gaji karyawan dengan menginput nama karyawan, gaji pokok, dan jam lembur. Program akan menghitung bonus lembur dengan jumlah jam lembur dikalikan Rp 50.000, serta menghitung total gaji dengan menjumlahkan gaji pokok dengan bonus lembur.

2. Source Code:

```
package main
```

```

import "fmt"

//prosedur utk menghitung rata-rata dan menentukan kelulusan
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   : %.2f\n", nilai1)
    fmt.Printf("Nilai 2   : %.2f\n", nilai2)
    fmt.Printf("Nilai 3   : %.2f\n", nilai3)
    fmt.Printf("Rata-rata  : %.2f\n", rataRata)
    fmt.Println("Status    :", status)
}

func main() { // main redeclared
    var nama string
    var nilai1, nilai2, nilai3 float64

    fmt.Print("Nama Mahasiswa: ")
    fmt.Scanln(&nama)

    fmt.Print("Nilai 1: ")
    fmt.Scanln(&nilai1)

    fmt.Print("Nilai 2: ")

```

```
fmt.Scanln(&nilai2)

fmt.Print("Nilai 3: ")
fmt.Scanln(&nilai3)

//memanggil prosedur
hitungRataRata(nama, nilai1, nilai2, nilai3)
}
```

Output:

```
PS C:\Sheila Stephanie Anindya\SMT 2\Alpro 2\code> go run "c:\Sheila Stephanie Anindya\
Nama Mahasiswa: meng
Nilai 1: 70
Nilai 2: 60
Nilai 3: 80

=== Hasil Akademik ===
Nama Mahasiswa : meng
Nilai 1       : 70.00
Nilai 2       : 60.00
Nilai 3       : 80.00
Rata-rata     : 70.00
Status        : Lulus
PS C:\Sheila Stephanie Anindya\SMT 2\Alpro 2\code> |
```

Penjelasan :

Program ini menghitung rata-rata nilai serta menyatakan status dengan menginput nama mahasiswa, nilai 1, nilai 2, dan nilai 3. Jika rata-rata yang dihasilkan ≥ 60 maka dinyatakan lulus, namun jika rata-rata ≤ 60 dinyatakan tidak lulus.

III. UNGUIDED

1. Source Code:

```
package main

import "fmt"

var a, b, c, d int

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

func permutasi(n, r int) int {
    if n < r {
        return 0
    }
    return faktorial(n) / faktorial(n-r)
}

func kombinasi(n, r int) int {
    if n < r {
        return 0
    }
    return faktorial(n) / (faktorial(r) * faktorial(n-r))
}
```



```

func hitungPermutasiKombinasi(a, b, c, d int) {
    if a < c || b < d {
        fmt.Println("Input tidak memenuhi syarat :  $a \geq c$  dan  $b \geq d$ ")
        return
    }
    fmt.Println(permutasi(a, c), kombinasi(a, c))
    fmt.Println(permutasi(b, d), kombinasi(b, d))
}

func main() {
    fmt.Scan(&a, &b, &c, &d)

    hitungPermutasiKombinasi(a, b, c, d)
}

```

Output:

```

PS C:\Sheila Stephanie Anindya\SMT 2\Alpro 2\code> go run "c:\Sheila
0086_unguided1.go"
5 10 3 10
60 10
3628800 1
PS C:\Sheila Stephanie Anindya\SMT 2\Alpro 2\code>

```

Penjelasan :

Program ini menghitung permutasi dan kombinasi dari 2 pasang bilangan yang diinput. Lalu program akan mengecek bilangan $a \geq c$ atau $b \geq d$, kalau tidak program bakal mengeluarkan output bahwa inputan tidak memenuhi syarat.

2. Source Code:

```
package main

import "fmt"

func skor(durasi [8]int) (int, int) {
    jumlahSoal := 0
    totalWaktu := 0

    for _, waktu := range durasi {
        if waktu < 301 {
            jumlahSoal++
            totalWaktu = totalWaktu + waktu
        }
    }
    return jumlahSoal, totalWaktu
}

func main() {
    var winner string

    maxSoal := -1    // -1 biar nilai prtama yg dibaca iterasi lbh gede
    minWaktu := 999999 // dikasih angka bsar diawal biar nilai prtama yg
dibaca lbh kecil

    for {
        var nama string

        fmt.Scan(&nama)
```

```

        if nama == "Selesai" {
            break
        }

        var durasi [8]int

        for i := range durasi {
            fmt.Scan(&durasi[i])
        }

        jumlahSoal, totalWaktu := skor(durasi)

        if jumlahSoal > maxSoal || (jumlahSoal == maxSoal && totalWaktu
< minWaktu) {
            winner, maxSoal, minWaktu = nama, jumlahSoal, totalWaktu
        }
    }

    fmt.Println(winner, maxSoal, minWaktu)
}

```

Output:

```

o run "c:\Sheila Stephanie Anindya\SMT 2\Alpro 2\code\
Astuti 20 50 301 301 61 71 75 10
Bertha 25 47 301 26 50 60 65 21
Selesai
Bertha 7 294
PS C:\Sheila Stephanie Anindya\SMT 2\Alpro 2\code> 

```

Penjelasan :

Program ini menentukan pemenang berdasarkan jumlah soal yang dijawab dan total waktu yang dihabiskan, dengan menginput nama dan durasi waktunya.

3. Source Code:

```
package main

import "fmt"

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

func main() {
    var n int

    fmt.Scan(&n)

    cetakDeret(n)
}
```

Output:

```
PS C:\Sheila Stephanie Anindya\SMT 2\Alpro 2\code> go run "c:\Sheila  
"  
22  
22 11 34 17 52 26 13 40 20 10 5 16 8 4 2 1  
PS C:\Sheila Stephanie Anindya\SMT 2\Alpro 2\code> 
```

Penjelasan :

Program ini mencetak deret angka dengan aturan jika n (inputan) genap akan dibagi 2, jika n ganjil akan dikali 3 + 1. Perhitungan akan berjalan sampai akhirnya berhenti di angka 1.

IV. KESIMPULAN

Prosedur adalah bagian dari program yang dirancang secara terpisah untuk menjalankan tugas tertentu. Terdapat dua jenis utama, yaitu subrutin (subprogram) dan fungsi, yang masing-masing memiliki peran dalam modularisasi kode.

Penggunaan prosedur dalam pemrograman terstruktur memberikan berbagai keuntungan, seperti meningkatkan keterbacaan, mengurangi duplikasi kode, dan menyembunyikan detail implementasi agar lebih mudah dikelola.

Prosedur yang baik memiliki ciri-ciri seperti memiliki satu tujuan utama, berukuran kecil, dan bersifat mandiri. Dengan menerapkan prosedur, program menjadi lebih terorganisir dan efisien.

V. REFERENSI

1. Wisnu Agung. (n.d.). *Prosedur dan Fungsi dalam PHP*. Retrieved from <https://wisnuagung.github.io/mkdocs-fix/menu/php/prosedurFungsi/>
2. Kode Dasar. (n.d.). *Prosedur dan Fungsi*. Retrieved from <https://kodedasar.com/blog/prosedur-dan-fungsi/>