

**LAPORAN PRAKTIKUM  
ALGORITMA DAN PEMROGRAMAN 2**

**LATIHAN SOAL 2**



**DISUSUN OLEH:  
RIZKINA AZIZAH  
103112400082  
S1 IF-12-01**

**PROGRAM STUDI S1 TEKNIK INFORMATIKA  
FAKULTAS INFORMATIKA  
TELKOM UNIVERSITY PURWOKERTO  
2025**

## TIPE A

### 1. Tipe A 1

Source Code:

```
// Rizkina Azizah_103112400082
package main

import "fmt"

func hitungBiayaSewa(jam int, menit int, member bool, voucher string) float64 {
    var harga float64
    if member {
        harga = 3500.0
    } else {
        harga = 5000.0
    }

    totalDurasiJam := float64(jam) + float64(menit)/60.0

    biayaSewa := totalDurasiJam * harga

    //diskon
    if totalDurasiJam >= 3 && (len(voucher) == 5 || len(voucher) == 6) {
        biayaSewa *= 0.9
    }

    return biayaSewa
}

func main() {
    var jam, menit int
    var member bool
    var voucher string
```

```

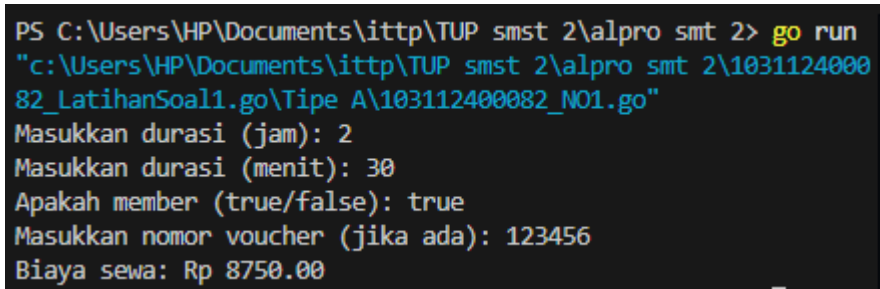
fmt.Print("Masukkan durasi (jam): ")
fmt.Scan(&jam)
fmt.Print("Masukkan durasi (menit): ")
fmt.Scan(&menit)
fmt.Print("Apakah member (true/false): ")
fmt.Scan(&member)
fmt.Print("Masukkan nomor voucher (jika ada): ")
fmt.Scan(&voucher)

biaya := hitungBiayaSewa(jam, menit, member, voucher)

fmt.Printf("Biaya sewa: Rp %.2f\n", biaya)
}

```

Output:



```

PS C:\Users\HP\Documents\ittp\TUP smst 2\alpro smt 2> go run
"c:\Users\HP\Documents\ittp\TUP smst 2\alpro smt 2\1031124000
82_LatihanSoal1.go\Tipe A\103112400082_NO1.go"
Masukkan durasi (jam): 2
Masukkan durasi (menit): 30
Apakah member (true/false): true
Masukkan nomor voucher (jika ada): 123456
Biaya sewa: Rp 8750.00

```

Deskripsi Program:

- Program ini digunakan untuk menghitung biaya sewa sesuai tarif, durasi, dan status membershipnya
- *func hitungBiayaSewa* menerima 4 parameter yaitu jam, menit, member, voucher. Fungsi ini menggunakan paradigma percabangan untuk menentukan harga per jam jika member 3.500 jika nonmember 5.000 dan juga untuk menghitung diskon jika memenuhi syarat yaitu durasi lebih dari 3 jam dan mempunyai voucher yang terdiri dari 5 atau 6 karakter
- *func main* berfungsi untuk:
  - input jam, menit, member, dan voucher
  - Memanggil *func hitungBiayaSewa()*
  - Menampilkan output *biaya* dengan format dua angka belakang koma

## 2. Tipe A 2

Source Code:

```
package main
import "fmt"

func PerfectNumber(n int) bool {
    cek := 0
    for i := 1; i < n; i++ {
        if n%i == 0 {
            cek += i
        }
    }
    return cek == n
}

func main() {
    var a, b int

    fmt.Print("Masukkan nilai a: ")
    fmt.Scan(&a)
    fmt.Print("Masukkan nilai b: ")
    fmt.Scan(&b)

    fmt.Printf("Perfect number antara %d dan %d: ", a, b)
    for i := a; i <= b; i++ {
        if PerfectNumber(i) {
            fmt.Print(i, " ")
        }
    }
    fmt.Println()
}
```

Output:

```

PS C:\Users\HP\Documents\ittp\TUP smst 2\alpro smt 2> go run
"c:\Users\HP\Documents\ittp\TUP smst 2\alpro smt 2\1031124000
82_LatihanSoal1.go\Tipe A\103112400082_No2.go"
Masukkan nilai a: 3
Masukkan nilai b: 13
Perfect number antara 3 dan 13: 6
PS C:\Users\HP\Documents\ittp\TUP smst 2\alpro smt 2>

```

Deskripsi Program:

- Program ini digunakan untuk mengimplementasikan penggunaan subprogram yang menampilkan barisan bilangan perfect number dalam rentang a sampai dengan b
- *func PerfectNumber* berfungsi untuk mengecek apakah n adalah bilangan sempurna, fungsi ini menggunakan paradigma perulangan dan percabangan
- *func main* berfungsi untuk:
  - input a dan b
  - Memanggil fungsi *PerfectNumber(i)* untuk mengecek setiap bilangan rentang a sampai b apakah merupakan perfect number atau bukan

### 3. Tipe A 3

Source Code

```

package main

import "fmt"

func main() {
    var x, y int

    fmt.Print("Masukkan nilai x: ")
    fmt.Scan(&x)
    fmt.Print("Masukkan nilai y: ")
    fmt.Scan(&y)

    pertemuan := 0
    for i := 1; i <= 365; i++ {
        if i%x == 0 && i%y != 0 {
            pertemuan++
        }
    }
}

```

```

    }
}

fmt.Printf("Jumlah pertemuan dalam setahun: %d\n", pertemuan)
}

```

Output:

```

PS C:\Users\HP\Documents\ittp\TUP smst 2\alpro smt 2> go run
"c:\Users\HP\Documents\ittp\TUP smst 2\alpro smt 2\1031124000
82_LatihanSoal1.go\Tipe A\tempCodeRunnerFile.go"
Masukkan nilai x: 2
Masukkan nilai y: 5
Jumlah pertemuan dalam setahun: 146
PS C:\Users\HP\Documents\ittp\TUP smst 2\alpro smt 2>

```

Deskripsi Program:

- Program ini digunakan untuk menghitung jumlah pertemuan rahasia antara dua agen mata-mata selama satu tahun (365 hari)
- Mendeklarasikan 2 variabel x dan y dengan tipe data integer dan berlaku juga sebagai input
- Menggunakan metode iteratif dengan cara:
  - Loop dari 1 hingga 365
  - Setiap angka dicek apakah merupakan kelipatan x dan bukan kelipatan y jika iya maka dianggap sebagai hari pertemuan
- Menampilkan output *pertemuan*

## TIPE B

### 1. Tipe B 1

Source Code:

```

package main

import "fmt"

func main() {
    var a, b int

```

```

    fmt.Print("Masukkan nilai a: ")
    fmt.Scan(&a)
    fmt.Print("Masukkan nilai b: ")
    fmt.Scan(&b)

    if a > b {
        fmt.Println("Nilai a harus kurang dari atau sama dengan b.")
        return
    }

    jumlahGanjil := 0
    for i := a; i <= b; i++ {
        if i%2 != 0 {
            jumlahGanjil++
        }
    }

    fmt.Print("Banyaknya angka ganjil : ", jumlahGanjil)
}

```

Output:

```

PS C:\Users\HP\Documents\ittp\TUP smst 2\alpro smt 2> go run
"c:\Users\HP\Documents\ittp\TUP smst 2\alpro smt 2\1031124000
82_LatihanSoal1.go\Tipe A\Tipe B\103112400082_No1B.go"
Masukkan nilai a: 1
Masukkan nilai b: 1000
Banyaknya angka ganjil : 500

```

Deskripsi Program:

- Program ini digunakan untuk penghitung angka ganjil dalam rentang
- Menggunakan dua variabel a dan b dengan tipe data integer yang merupakan variabel input
- *If a > b* berfungsi untuk memeriksa apakah a lebih dari b jika iya maka akan mencetak pesan error a harus kurang dari b

- Menggunakan paradigma perulangan untuk menghitung jumlah bilangan ganjil
- Output akan mencetak *jumlahGanjil*

## 2. Tipe B 2

Source Code:

```
package main

import "fmt"

func hitungBiaya(jumlahMenu, jumlahOrang, sisaMakanan int) int {
    var totalBiaya int

    if jumlahMenu > 50 {
        totalBiaya = 100000
    } else if jumlahMenu <= 3 {
        totalBiaya = 10000
    } else {
        totalBiaya = 10000 + (jumlahMenu-3)*2500
    }

    if sisaMakanan == 1 {
        totalBiaya *= jumlahOrang
    }

    return totalBiaya
}

func main() {
    var M int

    fmt.Print("Masukkan jumlah rombongan: ")
    fmt.Scan(&M)
```



```

    for i := 1; i <= M; i++ {
        var jumlahMenu, jumlahOrang int
        var sisaMakanan int

        fmt.Printf("Masukkan jumlah menu, jumlah orang, dan status makanan
yang tersisa (0 untuk tidak, 1 untuk iya): ")
        fmt.Scan(&jumlahMenu, &jumlahOrang, &sisaMakanan)
        totalBiaya := hitungBiaya(jumlahMenu, jumlahOrang, sisaMakanan)
        fmt.Printf("Total biaya rombongan %d: Rp %d\n", i, totalBiaya)
    }
}

```

Output:

```

PS C:\Users\HP\Documents\ittp\TUP smst 2\alpro smt 2> go run
"c:\Users\HP\Documents\ittp\TUP smst 2\alpro smt 2\1031124000
82_LatihanSoal1.go\Tipe A\Tipe B\tempCodeRunnerFile.go"
Masukkan jumlah rombongan: 2
Masukkan jumlah menu, jumlah orang, dan status makanan yang t
ersisa (0 untuk tidak, 1 untuk iya): 3 12 0
Total biaya rombongan 1: Rp 10000
Masukkan jumlah menu, jumlah orang, dan status makanan yang t
ersisa (0 untuk tidak, 1 untuk iya): 2 15 1
Total biaya rombongan 2: Rp 150000
PS C:\Users\HP\Documents\ittp\TUP smst 2\alpro smt 2>

```

Deskripsi program:

- Program ini digunakan untuk menghitung total biaya makan dari para pembeli
- *Func hitungBiaya* menerima tiga parameter jumlahMenu, jumlahOrang, sisaMakanan
- Menggunakan paradigma percabangan untuk menghitung biaya berdasarkan jumlah menu
- *Func main* berfungsi untuk:
  - Input M dan input jumlahMenu, jumlahOrang, sisaMakanan
  - Memanggil *hitungBiaya( )* untuk menghitung biaya rombongan
  - Menampilkan output total biaya untuk masing-masing rombongan

### 3. Tipe B 3

Source Code:

```
package main
```

```

import "fmt"

func faktor(n, i int) {
    if i > n {
        return
    }

    if n%i == 0 {
        fmt.Print(i, " ")
    }

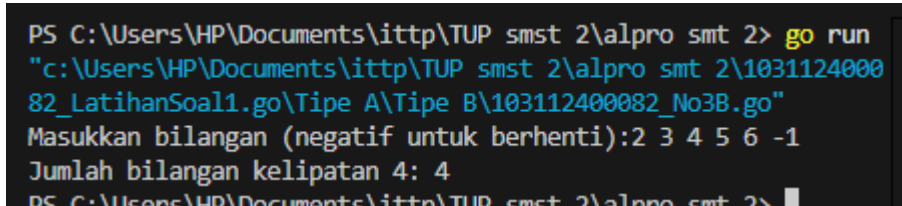
    faktor(n, i+1)
}

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

    faktor(n, 1)
    fmt.Println()
}

```

Output:



```

PS C:\Users\HP\Documents\ittp\TUP smst 2\alpro smt 2> go run
"c:\Users\HP\Documents\ittp\TUP smst 2\alpro smt 2\1031124000
82_LatihanSoal1.go\Tipe A\Tipe B\103112400082_No3B.go"
Masukkan bilangan (negatif untuk berhenti):2 3 4 5 6 -1
Jumlah bilangan kelipatan 4: 4
PS C:\Users\HP\Documents\ittp\TUP smst 2\alpro smt 2>

```

Deskripsi Program:

- Program ini digunakan untuk menghitung jumlah bilangan positif 4
- *Func jumlahKelipatan4()* berfungsi untuk :
  - Menerima input n.

- Pengecekan kelipatan 4: Jika n negatif maka fungsi mengembalikan 0 yang artinya perhitungan selesai, jika n kelipatan 4 maka n dijumlahkan dengan pemanggilan fungsi berikutnya, jika n bukan kelipatan 4 maka lanjut ke pemanggilan fungsi selanjutnya tanpa penjumlahan
- *Func main* berfungsi untuk:
  - Memanggil *jumlahKelipatan4()*
  - Menampilkan output total

## TIPE C

### 1. Tipe C 1

Source Code:

```
package main

import (
    "fmt"
    "strconv"
)

func potongBilangan(n int) (int, int) {
    inputStr := strconv.Itoa(n)
    length := len(inputStr)

    mid := (length + 1) / 2

    bilangan1, _ := strconv.Atoi(inputStr[:mid])
    bilangan2, _ := strconv.Atoi(inputStr[mid:])

    return bilangan1, bilangan2
}

func main() {
    var n int

    fmt.Print("Masukkan bilangan bulat positif (>10): ")
}
```

```

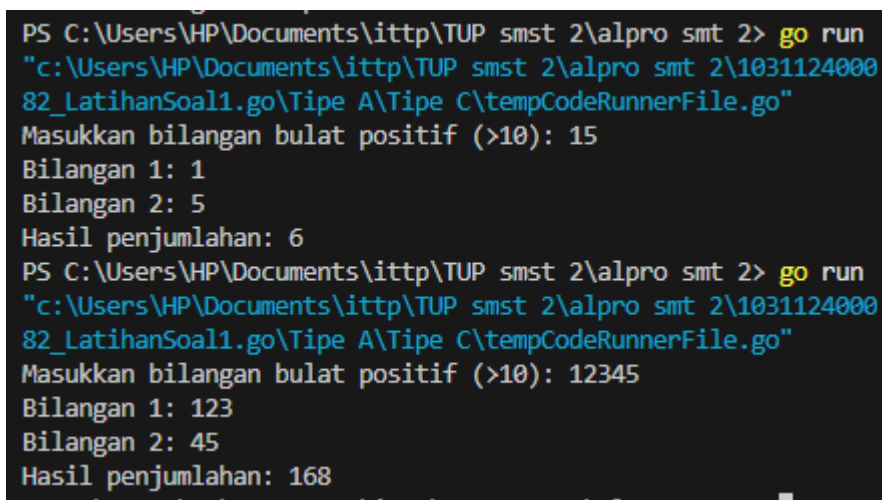
    fmt.Scan(&n)

    bilangan1, bilangan2 := potongBilangan(n)

    fmt.Printf("Bilangan 1: %d\n", bilangan1)
    fmt.Printf("Bilangan 2: %d\n", bilangan2)
    fmt.Printf("Hasil penjumlahan: %d\n", bilangan1+bilangan2)
}

```

Output:



```

PS C:\Users\HP\Documents\ittp\TUP smst 2\alpro smt 2> go run
"c:\Users\HP\Documents\ittp\TUP smst 2\alpro smt 2\1031124000
82_LatihanSoal1.go\Tipe A\Tipe C\tempCodeRunnerFile.go"
Masukkan bilangan bulat positif (>10): 15
Bilangan 1: 1
Bilangan 2: 5
Hasil penjumlahan: 6
PS C:\Users\HP\Documents\ittp\TUP smst 2\alpro smt 2> go run
"c:\Users\HP\Documents\ittp\TUP smst 2\alpro smt 2\1031124000
82_LatihanSoal1.go\Tipe A\Tipe C\tempCodeRunnerFile.go"
Masukkan bilangan bulat positif (>10): 12345
Bilangan 1: 123
Bilangan 2: 45
Hasil penjumlahan: 168

```

Deskripsi Program:

- Program ini digunakan untuk memotong suatu bilangan bulat positif tepat di posisi Tengah menjadi dua bagian sama panjang
- *Func hitungBiayaSewa* berfungsi menentukan:
  - Menentukan harga yang berbeda bagi member dan nonmember dengan menggunakan paradigma percabangan
  - Menghitung biaya sewa yaitu totalDurasijam x harga
  - Memberikan diskon 10% jika memenuhi syarat
  - Mengembalikan *biayaSewa*
- *Func main* berfungsi untuk :
  - Input jam, menit, member, voucher
  - Memanggil *func hitungBiayaSewa*
  - Output dicetak dengan 2 angka dibelakang koma

## 2. Tipe C 2

Source Code:

```
package main

import (
    "fmt"
)

func main() {
    var N int
    fmt.Print("Masukkan jumlah peserta: ")
    fmt.Scan(&N)

    // Variabel untuk menghitung jumlah hadiah
    hadiahA := 0
    hadiahB := 0
    hadiahC := 0

    for i := 1; i <= N; i++ {
        var nomorKartu int
        fmt.Printf("Masukkan nomor kartu peserta ke-%d: ", i)
        fmt.Scan(&nomorKartu)

        if semuaDigitSama(nomorKartu) {
            fmt.Println("Hadiah A")
            hadiahA++
        } else if semuaDigitBerbeda(nomorKartu) {
            fmt.Println("Hadiah B")
            hadiahB++
        } else {
            fmt.Println("Hadiah C")
            hadiahC++
        }
    }
}
```

```

    }
}

// Menampilkan jumlah peserta yang memperoleh masing-masing hadiah
fmt.Printf("\nJumlah yang memperoleh Hadiah A: %d\n", hadiahA)
fmt.Printf("Jumlah yang memperoleh Hadiah B: %d\n", hadiahB)
fmt.Printf("Jumlah yang memperoleh Hadiah C: %d\n", hadiahC)
}

func semuaDigitSama(M int) bool {
    var a, b, c int
    a = M / 100
    b = (M / 10) % 10
    c = M % 10
    return a == b && b == c

}

// Fungsi untuk memeriksa apakah semua digit berbeda
func semuaDigitBerbeda(M int) bool {
    var a, b, c int
    a = M / 100
    b = (M / 10) % 10
    c = M % 10

    return a != b && b != c && a != c
}

```

Output:

```

PS C:\Users\HP\Documents\ittp\TUP smst 2\alpro smt 2> go run
"c:\Users\HP\Documents\ittp\TUP smst 2\alpro smt 2\1031124000
82_LatihanSoal1.go\Tipe A\Tipe C\tempCodeRunnerFile.go"
Masukkan jumlah peserta: 3
Masukkan nomor kartu peserta ke-1: 333
Hadiah A
Masukkan nomor kartu peserta ke-2: 123
Hadiah B
Masukkan nomor kartu peserta ke-3: 898
Hadiah C

Jumlah yang memperoleh Hadiah A: 1
Jumlah yang memperoleh Hadiah B: 1
Jumlah yang memperoleh Hadiah C: 1
PS C:\Users\HP\Documents\ittp\TUP smst 2\alpro smt 2>

```

Deskripsi program:

- Program ini digunakan untuk menentukan jenis hadiah yang diperoleh setiap peserta
- Menggunakan paradigma percabangan untuk menentukan hadiah. Jika semua digit sama maka mendapat *Hadiah A*, jika semua digit berbeda maka mendapatkan *Hadiah B*, dan jika selain itu maka mendapatkan *Hadiah C*.
- *Func semuaDigitSama* berfungsi untuk mengecek digit apakah ketika digit sama
- *Func semuaDigitBerbeda* berfungsi untuk mengecek apakah ketiga digit berbeda

### 3. Tipe B 3

Source Code:

```

package main

import "fmt"

func perkalianIteratif(n, m int) int {
    hasil := 0
    for i := 0; i < n; i++ {
        hasil += m
    }
    return hasil
}

```

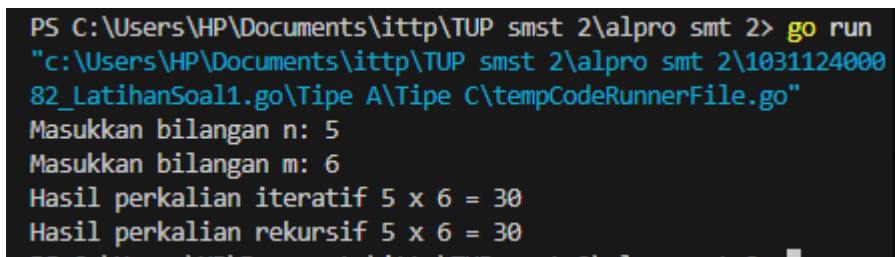
```
// Fungsi tail-recursive
func perkalianTailRekursif(n, m, akumulasi int) int {
    if n == 0 {
        return akumulasi
    }
    return perkalianTailRekursif(n-1, m, akumulasi+m)
}

func main() {
    var n, m int
    fmt.Print("Masukkan bilangan n: ")
    fmt.Scan(&n)
    fmt.Print("Masukkan bilangan m: ")
    fmt.Scan(&m)

    hasil1 := perkalianIteratif(n, m)
    hasil2 := perkalianTailRekursif(n, m, 0)
    fmt.Printf("Hasil perkalian iteratif %d x %d = %d\n", n, m, hasil1)
    fmt.Printf("Hasil perkalian rekursif %d x %d = %d\n", n, m, hasil2)
}

```

Output:



```
PS C:\Users\HP\Documents\ittp\TUP smst 2\alpro smt 2> go run
"c:\Users\HP\Documents\ittp\TUP smst 2\alpro smt 2\1031124000_
82_LatihanSoal1.go\Tipe A\Tipe C\tempCodeRunnerFile.go"
Masukkan bilangan n: 5
Masukkan bilangan m: 6
Hasil perkalian iteratif 5 x 6 = 30
Hasil perkalian rekursif 5 x 6 = 30
PS C:\Users\HP\Documents\ittp\TUP smst 2\alpro smt 2>

```

Deskripsi Program:



- Program ini digunakan untuk menghitung perkalian antara bilangan bulat  $n$  dan  $m$
- *Func perkalianIteratif( $n, m$  int)* berfungsi untuk melakukan penjumlahan  $m$  sebanyak  $n$  kali dengan menggunakan metode iteratif
- *Func perkalianTailRekursif* berfungsi untuk menentukan:
  - Basis: Jika  $n = 0$ , kembalikan hasil akumulasi
  - Rekurensi: kurangi  $n$  dan tambahkan  $m$  ke akumulasi
- Func main ( ):
  - Input bilangan  $n$  dan  $m$
  - Memanggil *Func perkalianIteratif( $n, m$  int)* dan *Func perkalianTailRekursif*
  - Menampilkan hasil output dari keduanya

