

ALGORITMA DAN PEMROGRAMAN 2
LATIHAN SOAL 2



Oleh:

ARIEL AHNAF KUSUMA

103112400050

IF-12-01

S1 TEKNIK INFORMATIKA
TELKOM UNIVERSITY PURWOKERTO
2025

1. TIPE A

- Sewa Sepeda

SOURCE CODE

```
1 package main
2
3 //103112400050
4 import "fmt"
5
6 func main() {
7     var jam, menit int
8     var member_103112400050 string
9     var kupon string
10    fmt.Print("masukkan Durasi (jam): ")
11    fmt.Scan(&jam)
12    fmt.Print("masukkan Durasi (menit): ")
13    fmt.Scan(&menit)
14    fmt.Print("apakah member? (true/false): ")
15    fmt.Scan(&member_103112400050)
16    fmt.Print("masukkan nomor kupon (jika ada): ")
17    fmt.Scan(&kupon)
18    member := member_103112400050 == "true"
19    durasi := hitungDurasi(jam, menit)
20    tarif := 5000.0
21    if member {
22        tarif = 3500.0
23    }
24    biaya := durasi * tarif
25    if durasi > 3 && (len(kupon) == 5 || len(kupon) == 6) {
26        biaya *= 0.9
27    }
28    fmt.Printf("biaya sewa setelah diskon (jika memenuhi syarat): Rp %.2f\n", biaya)
29 }
30 func hitungDurasi(jam, menit int) float64 {
31     totalMenit := jam*60 + menit
32     if totalMenit < 60 {
33         return 1.0
34     }
35     totalJam := totalMenit / 60
36     sisaMenit := totalMenit % 60
37     if sisaMenit >= 10 {
38         return float64(totalJam) + float64(sisaMenit)/60.0
39     }
40     return float64(totalJam)
41 }
42 }
```

OUTPUT CODE

```
PS C:\ARIEL AHNAF KUSUMA\103112400050_LATIHAN SOAL 2> go run "c:\ARIEL AHNAF
L A\103112400050_sewaspeda.go"
masukkan Durasi (jam): 2
masukkan Durasi (menit): 30
apakah member? (true/false): true
masukkan nomor kupon (jika ada): 123456
biaya sewa setelah diskon (jika memenuhi syarat): Rp 8750.00
PS C:\ARIEL AHNAF KUSUMA\103112400050_LATIHAN SOAL 2>
```

DESKRIPSI CODE

Kode Program soal tipe A nomor 1 ini berfungsi untuk menghitung biaya sewa berdasarkan duasi penggunaan, status keanggotaan dan penggunaan kupon (jika ada)

- Perfect Numbers

SOURCE CODE

```
1  package main
2
3  import "fmt"
4
5  //103112400050
6  func main() {
7      var a, b int
8      fmt.Print("Nilai awal: ")
9      fmt.Scan(&a)
10     fmt.Print("Nilai akhir: ")
11     fmt.Scan(&b)
12     fmt.Printf("Perfect number dari %d sampai %d:", a, b)
13     for i := a; i <= b; i++ {
14         s := 0
15         for j := 1; j <= i/2; j++ {
16             if i%j == 0 {
17                 s += j
18             }
19         }
20         if s == i {
21             fmt.Printf(" %d", i)
22         }
23     }
24 }
25
```

OUTPUT CODE

```
PS C:\ARIEL AHNAF KUSUMA\103112400050_LATIHAN SOAL 2> go run "c:\ARIEL
3112400050_LATIHAN SOAL 2\103112400050_SOAL A\103112400050_perfectnumbe
Nilai awal: 3
Nilai akhir: 13
Perfect number dari 3 sampai 13: 6
PS C:\ARIEL AHNAF KUSUMA\103112400050_LATIHAN SOAL 2> 
```

DESKRIPSI CODE

Code Program diatas adalah berfungsi untuk menampilkan *perfect number* dalam rentang yang diberikan oleh pengguna

- RANDEZVOUS

SOURCE CODE

```
1 package main
2
3 import (
4     "fmt"
5 )
6
7 func main() {
8     var x, y int
9     fmt.Print("Masukkan nilai x: ")
10    fmt.Scan(&x)
11    fmt.Print("Masukkan nilai y: ")
12    fmt.Scan(&y)
13
14    fmt.Printf("Jumlah pertemuan rahasia dalam setahun: %d\n", hitungPertemuan(x, y))
15 }
16
17 func hitungPertemuan(x, y int) (count int) {
18     for hari := 1; hari <= 365; hari++ {
19         if hari%x == 0 && hari%y != 0 {
20             count++
21         }
22     }
23     return
24 }
25
```

OUTPUT CODE

```
PS C:\ARIEL AHNAF KUSUMA\103112400050_LATIHAN SOAL 2> go run "c:\ARIEL AHNAF KUS
Masukkan nilai x: 2
Masukkan nilai y: 5
Jumlah pertemuan rahasia dalam setahun: 146
PS C:\ARIEL AHNAF KUSUMA\103112400050_LATIHAN SOAL 2> █
```

DESKRIPSI CODE

Code Program ini berfungsi menghitung berapa kali kedua orang itu bertemu dalam satu tahun (365 hari)

2. TIPE B

- VOUCHER

SOURCE CODE

```
1 package main
2
3 import "fmt"
4
5 //103112400050
6 func hitungAngkaGanjil(a, b int) (jumlah int) {
7     for i := a; i <= b; i++ {
8         if i%2 != 0 {
9             jumlah++
10        }
11    }
12    return jumlah
13 }
14
15 func main() {
16     var a, b int
17     fmt.Print("Masukkan nilai a: ")
18     fmt.Scan(&a)
19     fmt.Print("Masukkan nilai b: ")
20     fmt.Scan(&b)
21
22     fmt.Printf("Jumlah angka ganjil: %d\n", hitungAngkaGanjil(a, b))
23 }
24
```

OUTPUT CODE

```
PS C:\ARIEL AHNAF KUSUMA\103112400050_LATIHAN SOAL 2> go run "c:\103112400050_LATIHAN SOAL 2\103112400050_SOAL B\103112400050_voucher.go"
Masukkan nilai a: 1
Masukkan nilai b: 1000
Jumlah angka ganjil: 500
PS C:\ARIEL AHNAF KUSUMA\103112400050_LATIHAN SOAL 2>
```

DESKRIPSI CODE

Code Program ini berfungsi untuk digunakan menghitung banyaknya bilangan ganjil di antara dua bilangan

- RESTAURANT

SOURCE CODE

```
1 package main
2
3 //103112400050
4 import "fmt"
5
6 func main() {
7     var x int
8     fmt.Print("masukkan jumlah rombongan: ")
9     fmt.Scan(&x)
10    for i := 1; i <= x; i++ {
11        var menu, orang, sisa int
12        fmt.Printf("masukkan jumlah menu, jumlah orang, dan status sisa makanan (0 untuk tidak, 1 untuk iya)\n: ")
13        fmt.Scan(&menu, &orang, &sisa)
14        var tarif int
15        switch {
16        case menu > 50:
17            tarif = 100000
18        case menu <= 3:
19            tarif = 10000
20        default:
21            tarif = 10000 + (menu-3)*2500
22        }
23        if sisa == 1 {
24            tarif *= orang
25        }
26        fmt.Printf("total biaya untuk rombongan %d: Rp %d\n", i, tarif)
27    }
28 }
29
```

OUTPUT CODE

```
USUMA\103112400050_LATIHAN SOAL 2\103112400050_SOAL B\103112400050_restoran.go"
masukkan jumlah rombongan: 2
masukkan jumlah menu, jumlah orang, dan status sisa makanan (0 untuk tidak, 1 untuk iya)
: 3 12 0
total biaya untuk rombongan 1: Rp 10000
masukkan jumlah menu, jumlah orang, dan status sisa makanan (0 untuk tidak, 1 untuk iya)
: 2 15 1
total biaya untuk rombongan 2: Rp 150000
PS C:\ARIEL AHNAF KUSUMA\103112400050_LATIHAN SOAL 2>
```

DESKRIPSI CODE

Code Program diatas yaitu berfungsi untuk menghitung banyaknya *total biaya* makanan untuk beberapa rombongan berdasarkan jumlah menu, jumlah orang, dan status sisa makanan

- BILANGAN POSITIF KELIPATAN

SOURCE CODE

```
1 package main
2
3 import "fmt"
4
5 //103112400050
6 func main() {
7     total := jumlahKelipatan4(0)
8     fmt.Printf("Jumlah bilangan kelipatan 4: %d\n", total)
9 }
10
11 func jumlahKelipatan4(total int) int {
12     var bilangan int
13     fmt.Print("Masukkan bilangan (negatif untuk berhenti): ")
14     fmt.Scan(&bilangan)
15
16     if bilangan < 0 {
17         return total
18     }
19     if bilangan%4 == 0 {
20         total += bilangan
21     }
22
23     return jumlahKelipatan4(total)
24 }
25
```

OUTPUT CODE

```
PS C:\ARIEL AHNAF KUSUMA\103112400050_LATIHAN SOAL 2> go run "c:\ARIEL AHNAF KUSUMA\103112400050_LATIHAN SOAL 2\103112400050_SOAL 2\103112400050_bilanganpositif.go"
Masukkan bilangan (negatif untuk berhenti): 2
Masukkan bilangan (negatif untuk berhenti): 3
Masukkan bilangan (negatif untuk berhenti): 4
Masukkan bilangan (negatif untuk berhenti): 5
Masukkan bilangan (negatif untuk berhenti): 6
Masukkan bilangan (negatif untuk berhenti): -1
Jumlah bilangan kelipatan 4: 4
PS C:\ARIEL AHNAF KUSUMA\103112400050_LATIHAN SOAL 2>
```

DESKRIPSI CODE

Code Program diatas berfungsi untuk menghitung jumlah dari semua bilangan yang berkelipatan 4, yang dimasukkan secara berulang

3. TIPE C

- DIGIT

SOURCE CODE

```
1  package main
2
3  import "fmt"
4
5  //103112400050
6  func main() {
7      var bilangan int
8      fmt.Print("Masukkan bilangan bulat positif (>10): ")
9      fmt.Scanln(&bilangan)
10     digit := 0
11     for b := bilangan; b > 0; b /= 10 {
12         digit++
13     }
14     pangkat := 1
15     for i := 0; i < digit/2; i++ {
16         pangkat *= 10
17     }
18     angka1 := bilangan / pangkat
19     angka2 := bilangan % pangkat
20     hasil := angka1 + angka2
21
22     fmt.Println("Bilangan 1:", angka1)
23     fmt.Println("Bilangan 2:", angka2)
24     fmt.Println("Hasil penjumlahan:", hasil)
25 }
26
```

OUTPUT CODE


```
PS C:\ARIEL AHNAF KUSUMA\103112400050_LATIHAN SOAL 2> go run "c:\V
ARIEL AHNAF KUSUMA\103112400050_LATIHAN SOAL 2\103112400050_SOAL
C\103112400050_digit.go"
Masukkan bilangan bulat positif (>10): 15
Bilangan 1: 1
Bilangan 2: 5
Hasil penjumlahan: 6
PS C:\ARIEL AHNAF KUSUMA\103112400050_LATIHAN SOAL 2> go run "c:\V
ARIEL AHNAF KUSUMA\103112400050_LATIHAN SOAL 2\103112400050_SOAL
C\103112400050_digit.go"
Masukkan bilangan bulat positif (>10): 12345
Bilangan 1: 123
Bilangan 2: 45
Hasil penjumlahan: 168
PS C:\ARIEL AHNAF KUSUMA\103112400050_LATIHAN SOAL 2> █
```

DESKRIPSI CODE

Code Program ini dapat berfungsi untuk membagi sebuah bilangan bulat positif menjadi dua bagian yang berbeda berdasarkan jumlah digitnya, lalu menjumlahkan kedua bagian tersebut.

- BUKBER IF

SOURCE CODE

```

1  package main
2
3  import "fmt"
4
5  //103112400050
6  func main() {
7      var peserta, nomor int
8      var a, b, c int
9
10     fmt.Print("Masukkan jumlah peserta: ")
11     fmt.Scan(&peserta)
12
13     for i := 1; i <= peserta; i++ {
14         fmt.Printf("Masukkan nomor kartu peserta ke-%d: ", i)
15         fmt.Scan(&nomor)
16
17         d1 := nomor / 100
18         d2 := (nomor / 10) % 10
19         d3 := nomor % 10
20
21         switch {
22             case d1 == d2 && d2 == d3:
23                 fmt.Println("Hadiah A")
24                 a++
25             case d1 != d2 && d2 != d3 && d1 != d3:
26                 fmt.Println("Hadiah B")
27                 b++
28             default:
29                 fmt.Println("Hadiah C")
30                 c++
31         }
32     }
33
34     fmt.Println("Jumlah yang memperoleh Hadiah A:", a)
35     fmt.Println("Jumlah yang memperoleh Hadiah B:", b)
36     fmt.Println("Jumlah yang memperoleh Hadiah C:", c)
37 }
38

```

OUTPUT CODE

```
PS C:\ARIEL AHNAF KUSUMA\103112400050_LATIHAN SOAL 2> go run "c:\
ARIEL AHNAF KUSUMA\103112400050_LATIHAN SOAL 2\103112400050_SOAL
C\103112400050_bukber.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:\ARIEL AHNAF KUSUMA\103112400050_LATIHAN SOAL 2> █
```

DESKRIPSI CODE

Kode Program di atas yaitu berfungsi untuk mengidentifikasi jenis hadiah yang diberikan kepada peserta berdasarkan pola tiga digit yang ditemukan pada nomor kartu peserta.

Hadiah A akan diperoleh jika ketiga digit pada nomor kartu sama

Hadiah B akan diperoleh jika ketiga digit berbeda satu sama lain

Hadiah C akan diperoleh jika hanya dua digit yang sama.

- **PERKALIAN DENGAN CARA PENJUMLAHAN**

SOURCE CODE

```

1  package main
2
3  import "fmt"
4
5  //103112400050
6  func main() {
7      var n, m int
8      fmt.Print("Masukkan bilangan n: ")
9      fmt.Scan(&n)
10     fmt.Print("Masukkan bilangan m: ")
11     fmt.Scan(&m)
12
13     hasil := perkalianRekursif(n, m, 0)
14
15     fmt.Printf("Hasil dari %d x %d = %d\n", n, m, hasil)
16 }
17
18 func perkalianRekursif(n, m, total int) int {
19     if n == 0 || m == 0 {
20         return 0
21     }
22     if n < 0 {
23         return -perkalianRekursif(-n, m, 0)
24     }
25     if n == 1 {
26         return total + m
27     }
28     return perkalianRekursif(n-1, m, total+m)
29 }
30
31 func perkalianIteratif(n, m int) int {
32     if n == 0 || m == 0 {
33         return 0
34     }
35     negatif := false
36     if n < 0 {
37         n = -n
38         negatif = true
39     }
40     total := 0
41     for i := 0; i < n; i++ {
42         total += m
43     }
44     if negatif {
45         return -total
46     }
47     return total
48 }

```

OUTPUT CODE

```
PS C:\ARIEL AHNAF KUSUMA\103112400050_LATIHAN SOAL 2> go run "c:\ARIEL A
03112400050_LATIHAN SOAL 2\103112400050_SOAL C\103112400050_perkaliandgp
"
Masukkan bilangan n: 5
Masukkan bilangan m: 6
Hasil dari 5 x 6 = 30
```

DESKRIPSI CODE

Kode Program diatas berfungsi untuk menghitung perkalian dua bilangan bulat tanpa menggunakan kali (*), dan program diatas menggunakan **prosedur iteratif** yang lalu menggunakan **konversi pola tail-end recursion** agar mendapatkan solusi dalam bentuk **rekursif**

