

## Modul O6 Terbimbing

### Tipe Bentuk

1. **[Bea parkir]** Bea parkir kendaraan di suatu swalayan dihitung sesuai durasi parkir dan tarif per jam per jenis kendaraan. Berikut aturannya:

- Untuk kendaraan jenis mobil: jam pertama Rp 5.000, per satu jam berikutnya Rp 2.500.
- Untuk kendaraan jenis motor: jam pertama Rp 2.000, per satu jam berikutnya Rp 1.000.

Buatlah program dalam bahasa Go untuk mengimplementasikan perhitungan bea parkir tersebut. Untuk penyelesaiannya gunakan tipe bentuk struct **waktu** dengan field jam, menit, dan detik yang masing-masing bertipe integer. Definisikan pula tipe bentuk struct **kendaraan** dengan field jenis kendaraan (jke), nopol (no) dengan tipe string; jam masuk (jm) dan jam keluar (jk) dengan tipe waktu; serta bea parkir (bea) dan durasi parkir (durasi) dengan tipe integer.

```
package main

import "fmt"

// tipe bentuk struct waktu dengan field jam, menit, detik semuanya bertipe integer
type waktu struct {
    --- int
}

// tipe bentuk struct kendaraan dengan field jenis kendaraan (jke), nopol (no),
// jam masuk (jm) dan jam keluar (jk) bertipe waktu
// bea parkir (bea) dan durasi parkir (durasi) bertipe integer
type kendaraan struct {
    --- string
    --- waktu
    --- int
}

func main() {
    var k01, k02, k03 kendaraan
    var jenis, nopol string
    var j1, m1, d1, j2, m2, d2 int

    fmt.Scan(&jenis, &nopol, &j1, &m1, &d1, &j2, &m2, &d2)
    // Membuat kendaraan k01
    k01 = ---
```

```

    fmt.Scan(&jenis, &nopol, &j1, &m1, &d1, &j2, &m2, &d2)
    // Membuat kendaraan k02
    k02 = ___

    fmt.Scan(&jenis, &nopol, &j1, &m1, &d1, &j2, &m2, &d2)
    // Membuat kendaraan k03
    k03 = ___

    // hitung bea parkir 3 kendaraan
    ___
    ___
    ___

    // cetak data parkir per kendaraan
    cetak_data(k01)
    cetak_data(k02)
    cetak_data(k03)
}

func kendaraan_baru(j, n string, a, b, c, d, e, f int) kendaraan {
    /* Mengembalikan nilai kendaraan dengan data jenis kendaraan (j), nopol (n), jam masuk
       dalam jam (a), menit (b), detik (c) dan jam keluar dalam jam (d), menit (e),
       detik (f). */
    var k kendaraan
    ___
    return k
}

func durasi_parkir(k *kendaraan) {
    /* IS: k terdefinisi, kecuali durasi parkir (k.durasi) dan bea parkir (k.bea)
       Proses : Ubah durasi tot_detik menjadi k.durasi dalam satuan jam
       FS: Durasi parkir (k.durasi) berisi nilai dalam satuan jam
       */
    var tot_detik int
    ___
}

func bea_parkir(k *kendaraan) {
    /* IS: k terdefinisi, kecuali durasi parkir (k.durasi) dan bea parkir (k.bea)
       Proses: Memanggil prosedur durasi_parkir dan menghitung k.bea
       Jika kendaraan mobil, biaya parkir jam pertama Rp 5000, per jam selanjutnya Rp
       2500. Jika kendaraan motor, biaya parkir jam pertama Rp 2000, per jam selanjutnya
       Rp 1000
       FS: Durasi parkir (k.durasi) berisi nilai dalam satuan jam
       */
    ___
}

func cetak_data(k kendaraan) {
    /* IS: k terdefinisi
       FS: Tercetak data parkir kendaraan dengan format:
       "Bea parkir <jke> nopol <no> dengan durasi <durasi> jam: Rp <bea>."
       */
    fmt.Printf("")
}

```

Masukan dan keluaran program adalah sebagai berikut:

**Masukan** terdiri 3 baris. masing-masing baris berisi jenis kendaraan (jke), nomor polisi (no), waktu masuk parkir dalam jam (j1), menit (m1), dan detik (d1), waktu keluar parkir dalam jam (j2), menit (m2), dan detik (d2).

**Keluaran** berupa data parkir 3 kendaraan sesuai format pencetakan pada contoh.

### Contoh Masukan dan Keluaran

No	Masukan	Keluaran
1	mobil D123D 8 0 0 10 0 0	Bea parkir mobil nopol D123D dengan durasi 2 jam: Rp 7500.
	mobil D123E 8 0 0 10 0 1	Bea parkir mobil nopol D123E dengan durasi 3 jam: Rp 10000.
	mobil D124D 8 0 0 10 1 0	Bea parkir mobil nopol D124D dengan durasi 3 jam: Rp 10000.
2	motor D123A 8 0 0 10 0 0	Bea parkir motor nopol D123A dengan durasi 2 jam: Rp 3000.
	motor D123B 8 0 0 10 0 1	Bea parkir motor nopol D123B dengan durasi 3 jam: Rp 4000.
	motor D124C 8 0 0 10 1 0	Bea parkir motor nopol D124C dengan durasi 3 jam: Rp 4000.
3	mobil D123G 9 0 0 10 59 0	Bea parkir mobil nopol D123G dengan durasi 2 jam: Rp 7500.
	motor D123B 8 0 59 10 0 1	Bea parkir motor nopol D123B dengan durasi 2 jam: Rp 3000.
	mobil D124F 8 0 0 10 1 59	Bea parkir mobil nopol D124F dengan durasi 3 jam: Rp 10000.

2. **[Nilai mahasiswa]** Buatlah program untuk membaca data, mencetak data, menghitung rata-rata nilai, dan mencetak nilai tertinggi dari 3 orang mahasiswa. Data mahasiswa dibuat dengan tipe bentukan struct dengan field nama (string), NIM (string), dan nilai (real). Program harus mengimplementasikan pemrograman modular dengan masukan dan keluaran sebagai berikut:

**Masukan** terdiri dari 3 baris data mahasiswa dengan masing-masing barisnya terdiri atas data nama, NIM, dan nilai.

**Keluaran** terdiri dari beberapa baris sebagaimana terlihat pada contoh, yaitu data 3 mahasiswa, rata-rata nilai kelas, dan nilai tertinggi

```
package main

import "fmt"

// Tipe bentukan struct mahasiswa dengan field nama, NIM, dan nilai
type mahasiswa struct {
    ---
    ---
}

func main() {
    var mhs1, mhs2, mhs3 mahasiswa
```

```
// Panggil prosedur baca
---
// Panggil prosedur cetak
---
// Panggil fungsi rata_rata_nilai
---
// Panggil prosedur mhs_max_nilai
---
}

func baca(m1, m2, m3 *mahasiswa) {
/* IS: m1, m2, m3 terdefinisi sembarang
Proses: membaca dari piranti masukan data nama, NIM, dan nilai dari m1, m2, dan m3
FS: m1, m2, m3 berisi nilai
*/
---
}

func cetak(m1, m2, m3 mahasiswa) {
/* IS: m1, m2, m3 terdefinisi
FS: Semua field dari m1, m2, m3 tercetak di layar
*/
---
}

func rata_rata_nilai(m1, m2, m3 mahasiswa) float64 {
/* Mengembalikan rata-rata nilai dari 3 mahasiswa m1, m2, dan m3 */
---
}

func mhs_max_nilai(m1, m2, m3 mahasiswa) {
/* IS: m1, m2, dan m3 terdefinisi
FS: Tercetak di layar nama mahasiswa dengan nilai tertinggi dengan format
"Mahasiswa dengan nilai tertinggi <nilai tertinggi>: <nama mahasiswa> <NIM
mahasiswa>"
Asumsi nilai mahasiswa bersifat unik, sehingga hanya ada 1 nilai tertinggi
*/
---
}
```

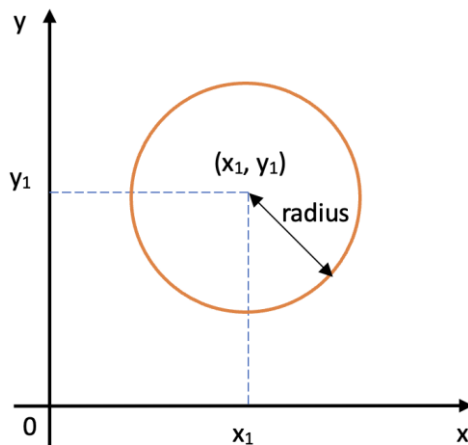
### Contoh Masukan dan Keluaran

No	Masukan	Keluaran
1	Budi 102 90 Cintya 101 80 Deni 103 88	Budi 102 90 Cintya 101 80 Deni 103 88 Rata-rata nilai: 86 Mahasiswa dengan nilai tertinggi 90: Budi 102
2	John 102 50 Jim 103 100 Hillary 101 75	John 102 50 Jim 103 100 Hillary 101 75 Rata-rata nilai: 75 Mahasiswa dengan nilai tertinggi 100: Jim 103
3	Bob 102 50 Deasy 103 95 Brian 101 98	Bob 102 50 Deasy 103 95 Brian 101 98 Rata-rata nilai: 81 Mahasiswa dengan nilai tertinggi 98: Brian 101

## Modul O6 Mandiri

### Tipe Bentukan

1. **[Lingkaran]** Sebuah lingkaran dalam koordinat kartesius dua dimensi memiliki atribut titik pusat  $(x_1, y_1)$  dan radius.



Buatlah program dalam bahasa Go untuk menghitung luas dan keliling lingkaran itu. Untuk menyelesaikan problem ini implementasikan penggunaan tipe bentukan struct dan pemrograman modular. Untuk titik pusat lingkaran gunakan tipe bentukan struct **titik** yang memiliki field  $x$  dan  $y$  bertipe real. Sedangkan tipe bentukan struct **lingkaran** memiliki field titik pusat bertipe titik dan radius, luas, keliling bertipe real. Program memiliki masukan dan keluaran sebagai berikut:

**Masukan** terdiri dari 2 baris bilangan real yang menyatakan titik  $x$  dan  $y$ , serta radius.

**Keluaran** berupa teks berisi data 2 lingkaran, yaitu masing-masing titik pusat, radius, luas dan keliling lingkarannya. Perhatikan format pencetakannya pada contoh.

```
package main

import "fmt"

// Tipe bentukan titik dengan field x dan y
type titik struct {
    _ float64
}

/* tipe bentukan struct lingkaran dengan field titik_pusat bertipe titik dan
radius, luas, keliling bertipe float64 */
```

```

type lingkaran struct {
    ---- titik
    ---- float64
}

func main() {
    var c1, c2 lingkaran
    var x, y, radius float64

    fmt.Scan(&x, &y, &radius)
    // membuat lingkaran baru c1
    c1 = ---

    fmt.Scan(&x, &y, &radius)
    // membuat lingkaran baru c2
    c2 = ---

    // Menghitung luas c1
    ---

    // Menghitung luas c2
    ---

    // Menghitung keliling c1
    ---

    // Menghitung keliling c2
    ---

    // cetak data
    cetak_data(c1)
    cetak_data(c2)
}

func lingkaran_baru(x, y, r float64) lingkaran {
    /* Mengembalikan lingkaran dengan titik pusat x, y dan radius r */
    var l lingkaran
    var tp titik
    ---
    return l
}

func hitung_luas(l *lingkaran) {
    /* Mengembalikan luas lingkaran (l.luas) dengan pi = 3.14 */
    const pi float64 = 3.14
    l.luas = ---
}

func hitung_keliling(l *lingkaran) {
    /* Mengembalikan keliling lingkaran (l.keliling) dengan pi = 3.14 */
    const pi float64 = 3.14
    l.keliling = ---
}

func cetak_data(l lingkaran) {
    /* IS: Lingkaran l terdefinisi
    FS: Tercetak data lingkaran l dengan format:

```

```

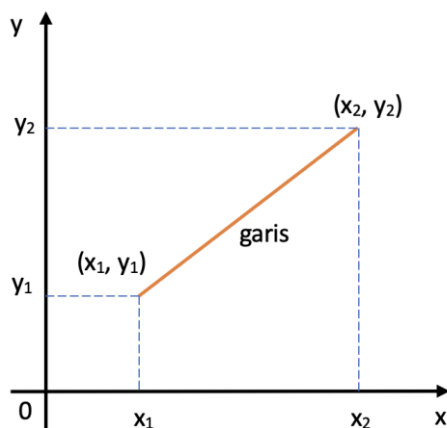
        "Lingkaran berpusat di titik (<x, y>) dan berradius <radius> memiliki
        luas <luas> dan keliling sebesar <keliling>"
        Luas dan keliling dibulatkan 1 digit di belakang koma
    */
    fmt.Printf("")
    fmt.Printf("")
}

```

### Contoh Masukan dan Keluaran

No	Masukan	Keluaran
1	1 1 5 2 2 10	Lingkaran berpusat di titik (1, 1) dan berradius 5 memiliki luas 78.5 dan keliling sebesar 31.4 Lingkaran berpusat di titik (2, 2) dan berradius 10 memiliki luas 314.0 dan keliling sebesar 62.8
2	1.5 1.5 10.4 2.1 2.4 5.5	Lingkaran berpusat di titik (1.5, 1.5) dan berradius 10.4 memiliki luas 339.6 dan keliling sebesar 65.3 Lingkaran berpusat di titik (2.1, 2.4) dan berradius 5.5 memiliki luas 95.0 dan keliling sebesar 34.5

2. **[Titik dan Garis]** Sebuah titik pada koordinat kartesius 2 dimensi dibentuk oleh komponen absis (x) dan ordinat (y). Sementara itu, sebuah garis lurus cukup dibentuk oleh dua buah titik. Untuk menghitung panjang garis lurus dapat digunakan rumus jarak antara dua titik.



$$\text{panjang} = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

Buatlah program dalam bahasa Go untuk menghitung panjang garis lurus yang dibentuk oleh dua buah titik pada koordinat kartesius 2 dimensi. Untuk menyelesaikan problem ini implementasikan penggunaan tipe bentukan struct dan pemrograman modular. Tipe bentukan struct **titik** memiliki field x dan y bertipe real,

sedangkan tipe bentukan struct **garis** memiliki field t1 dan t2 bertipe titik. Dalam problem ini ditambahkan pula field panjang bertipe real pada tipe bentukan garis.

```
package main

import (
    "fmt"
    "math"
)

// Tipe bentukan struct titik dengan field x dan y bertipe float64
type titik struct {
    ---
}

// Tipe bentukan struct garis dengan field t1 dan t2 bertipe titik
// serta panjang bertipe float64
type garis struct {
    ---
}

func main() {
    var p1, p2, p3, p4 titik // p = point
    var l1, l2 garis         // l = line
    var x, y float64

    baca_x_y(&x, &y)
    // buat titik p1 dengan memanggil fungsi titik_baru
    p1 = ---

    baca_x_y(&x, &y)
    // buat titik p2 dengan memanggil fungsi titik_baru
    p2 = ---

    baca_x_y(&x, &y)
    // buat titik p3 dengan memanggil fungsi titik_baru
    p3 = ---

    baca_x_y(&x, &y)
    // buat titik p4 dengan memanggil fungsi titik_baru
    p4 = ---

    // buat garis l1 dengan memanggil fungsi garis_baru dengan titik p1 dan p2
    l1 = ---

    // hitung panjang garis l1
    ---

    // hitung panjang garis l2
    ---

    cetak_data(l1)
    cetak_data(l2)
}

func baca_x_y(a, b *float64) {
    /* IS: a dan b terdefinisi sembarang (data a dan b tersedia pada piranti masukan)
       FS: a dan b berisi nilai
    */
    ---
}
```



```
func titik_baru(a, b float64) titik {
    /* Mengembalikan titik dengan field a dan b */
    var p titik
    ---
    return p
}

func garis_baru(a, b titik) garis {
    /* Mengembalikan garis dengan field titik a dan b */
    var g garis
    ---
    return g
}

func hitung_panjang_garis(g *garis) {
    /* Mengembalikan panjang g.panjang yang dihitung dengan memanggil fungsi panjang */
    g.panjang = ---
}

func panjang(a, b titik) float64 {
    /* Mengembalikan panjang atau jarak dari titik a ke b dengan menggunakan fungsi
    math.Sqrt (akar) dan math.Pow (pangkat). Gunakan rumus jarak antara 2 titik */
    return ---
}

func cetak_data(g garis) {
    /* IS: Garis g terdefinisi
    FS: Tercetak data garis g dengan format:
    "Garis dibentuk oleh titik <x, y> dan <x, y> memiliki panjang sebesar <panjang>."
    */
    fmt.Printf("", --- )
    fmt.Printf("", --- )
}
```

Program memiliki masukan dan keluaran sebagai berikut:

**Masukan** terdiri dari 4 baris bilangan real yang menyatakan 4 buah pasangan x dan y.

**Keluaran** berupa teks berisi data 2 garis, yaitu masing-masing 2 titik pembentuk garis dan panjang garisnya. Perhatikan format pencetakannya pada contoh.

#### Contoh Masukan dan Keluaran

No	Masukan	Keluaran
1	1 1 4 5 -1 -1 -4 -5	Garis dibentuk oleh titik (1, 1) dan (4, 5) memiliki panjang sebesar 5. Garis dibentuk oleh titik (-1, -1) dan (-4, -5) memiliki panjang sebesar 5.
2	0 0 -2 -3 0 0 -8 10	Garis dibentuk oleh titik (0, 0) dan (-2, -3) memiliki panjang sebesar 3.605551275463989. Garis dibentuk oleh titik (0, 0) dan (-8, 10) memiliki panjang sebesar 12.806248474865697.