

Tugas Pendahuluan Modul O4

Prosedur

Instruksi Pengerjaan

1. Diperuntukkan bagi mahasiswa S1 Informatika.
2. Tugas ini bersifat Individu.
3. Jawaban tugas dikirim dalam bentuk kode dalam bahasa Go.
4. Dilarang melakukan tindakan plagiasi ataupun menggunakan bantuan tool Generative AI.
5. Tugas dikumpulkan pada LMS dengan deadline Senin, 10 Maret 2025 pukul 06.00 WIB
6. Tugas ini merupakan latihan untuk bisa mengikuti praktikum dengan baik.
7. Contact Person Praktikum Algoritma Pemrograman:
 - 085847990203 (Gede Bagus)
 - 085212528394 (Arief Rahman)
 - 085105303555 (Kayyisa Zulfa)

Soal Tugas Pendahuluan

1. **[Lingkaran dan Persegi]** Buatlah program dalam bahasa Go untuk menghitung total luas dan total keliling dari benda 2D lingkaran dan persegi. Program akan memanfaatkan prosedur-prosedur berikut:

```
procedure hitungLuasKelilingLingkaran(in r : real, in/out l, k : real)
{IS:      Radius lingkaran (r) terdefinisi. Luas (l) dan Keliling (l)
          lingkaran terdefinisi sembarang
  Proses: Luas lingkaran dihitung dengan rumus  $l = \pi \times r \times r$ . Keliling
          lingkaran dihitung dengan rumus  $k = 2 \times \pi \times r$ . Gunakan nilai  $\pi = 3.14$ 
  FS:      luas dan keliling lingkaran berisi nilai }
```

```
procedure hitungLuasKelilingPersegi(in s : real, in/out l, k : real)
{IS:      Sisi persegi (s) terdefinisi. Luas (l) dan Keliling (l)
          persegi terdefinisi sembarang
  Proses: Luas persegi dihitung dengan rumus  $l = s \times s$ . Keliling persegi
          dihitung dengan rumus  $k = 4 \times s$ .
  FS:      Luas dan keliling persegi berisi nilai }
```

```
procedure hitungTotal(in lL, lP, kL, kP: real, in/out totLuas, totKel: real)
{ IS:      Luas lingkaran (lL), luas persegi (lP), keliling lingkaran (kL) dan
          keliling persegi (kP) terdefinisi. Total luas (totLuas) dan total
          keliling (totKel) terdefinisi sembarang.
  Proses: Total luas dihitung dengan rumus  $totLuas = lL + lP$ . Keliling dihitung
          dengan rumus  $totKel = kL + kP$ 
  FS:      Total luas dan total keliling berisi nilai }
```

Masukan dan keluaran program adalah sebagai berikut:

Masukan berupa serangkaian pasangan bilangan real yang menyatakan radius lingkaran dan sisi persegi dengan diakhiri bilangan 0 0 sebagai sentinel.

Keluaran berupa 8 bilangan real, yaitu radius (R), sisi (S), luas lingkaran (LL), luas persegi (LP), keliling lingkaran (KL), keliling persegi (KP), total luas (TL), dan total

keliling (TP). Semua bilangan dibulatkan dua digit di belakang koma. Perhatikan format pencetakan pada tabel contoh masukan dan keluaran.

Contoh masukan dan keluaran

No	Masukan	Keluaran							
1	10.5 11.2	R	S	LL	LP	KL	KP	TL	TP
	22.4 25.1	10.50	11.20	346.19	125.44	65.94	44.80	471.62	110.74
	15.5 25.75	22.40	25.10	1575.53	630.01	140.67	100.40	2205.54	241.07
	12.33 13.21	15.50	25.75	754.38	663.06	97.34	103.00	1417.45	200.34
	16.45 21.35	12.33	13.21	477.37	174.50	77.43	52.84	651.87	130.27
	0 0	16.45	21.35	849.69	455.82	103.31	85.40	1305.51	188.71
2	10.00 10.00	R	S	LL	LP	KL	KP	TL	TP
	0 0	10.00	10.00	314.00	100.00	62.80	40.00	414.00	102.80
3	0 0								

2. **[Skor Bola]** Buatlah program dalam bahasa Go untuk merekap data-data pertandingan yang diperoleh sebuah klub sepakbola berdasarkan hasil pertandingan yang diperolehnya. Data-data tersebut berupa banyaknya pertandingan yang dimainkan, jumlah kemenangan, jumlah draw, jumlah kekalahan, jumlah gol, jumlah kegoalan, jumlah selisih gol, dan total poin yang diperoleh.

Masukan berupa banyaknya pertandingan (N) dan diikuti oleh hasil pertandingan sebanyak N baris. Pada setiap baris terdapat skor pertandingan berupa banyak gol dan banyak kegoalan.

Keluaran berupa data pertandingan, yaitu banyaknya pertandingan yang dimainkan, jumlah kemenangan, jumlah draw, jumlah kekalahan, jumlah gol, jumlah kegoalan, jumlah selisih gol, dan total poin yang diperoleh.

```

procedure hitungMenang(in g, k : integer, in/out jm : integer)
{IS: Gol (g), kegoalan (k), dan jumlah menang (jm) terdefinisi
FS: Jumlah menang (jm) berisi nilai baru. Menang adalah jika jumlah
    gol lebih besar dari jumlah kegoalan (g > k) }

```

procedure hitungDraw(**in** g, k : **integer**, **in/out** jd : **integer**)

{IS: Gol (g), kegoalan (k), dan jumlah menang (jd) terdefinisi

FS: Jumlah draw (jd) berisi nilai baru. Draw adalah jika jumlah gol sama dengan jumlah kegoalan ($g == k$) }

procedure hitungKalah(**in** g, k : **integer**, **in/out** jk : **integer**)

{IS: Gol (g), kegoalan (k), dan jumlah kalah (jk) terdefinisi

FS: Jumlah kalah (jk) berisi nilai baru. Kalah adalah jika jumlah gol lebih kecil dari jumlah kegoalan ($g < k$) }

procedure hitungJumGolKegolanSelisih(**in** g, k : **integer**, **in/out** jg, jk, jsg : **integer**)

{IS: Gol (g), kegoalan (k), jumlah gol (jg), jumlah kegoalan (jk), dan jumlah selisih gol (jsg) terdefinisi

FS: Jumlah gol (jg), jumlah kegoalan (jk), dan jumlah selisih gol (jsg) berisi nilai baru }

procedure hitungJumPoint(**in/out** jp : **integer**)

{IS: Jumlah poin (jp) terdefinisi sembarang

FS: Jumlah poin (jp) berisi nilai. Jumlah poin dihitung dari banyak kemenangan x 3 ditambah jumlah draw x 1 }

Contoh masukan dan keluaran

No	Masukan	Keluaran
1	4 1 0 3 0 2 2 0 3	4 2 1 1 6 5 1 7

2	26	26 13 2 11 35 38 -3 41
	1 3	
	1 2	
	2 1	
	2 1	
	3 0	
	4 3	
	2 2	
	1 2	
	3 2	
	0 2	
	0 0	
	0 3	
	2 1	
	0 1	
	3 0	
	1 0	
	1 0	
	0 3	
	2 1	
	0 1	
	1 0	
	1 3	
	1 3	
	3 2	
	0 2	
	1 0	

3. **[Kalkulator Sederhana]** Buatlah sebuah program kalkulator sederhana dalam bahasa Go dengan menu pilihan penjumlahan, perkalian, dan pembagian. Lengkapi prosedur-prosedurnya dengan spesifikasi yang diminta.

```

program kalkulator
kamus
    pilih : integer
algoritma
    repeat
        menu()
        output("Pilih (1/2/3/4)?")
        input(pilih)
        depend on (pilih)

```

```

        1 : hitungJumlah()
        2 : hitungKali()
        3 : hitungBagi()

    enddependon
until pilih == 4
endprogram

```

```

procedure menu()
{ IS: -
  FS: Tercetak di layar tampilan sebagai berikut:
    -----
          M E N U
    -----
    1. Hitung Penjumlahan
    2. Hitung Perkalian
    3. Hitung Pembagian
    4. Exit
    -----
}

```

```

procedure hitungJumlah()
{ IS: -
  Proses: Membaca dua bilangan bulat dan menjumlahkan kedua bilangan
          itu
  FS:     Mencetak hasil penjumlahan }

```

```

procedure hitungKali()
{ IS: -
  Proses: Membaca dua bilangan bulat dan mengalikan kedua bilangan itu
  FS:     Mencetak hasil perkalian }

```

```

procedure hitungBagi()
{ IS: -

```

Proses: Membaca dua bilangan real dan membagi bilangan pertama oleh bilangan kedua

FS: Mencetak hasil pembagian }

Contoh interaksi masukan dan keluaran

(Yang diberi garis bawah adalah masukan).

M E N U

1. Hitung Penjumlahan
2. Hitung Perkalian
3. Hitung Pembagian
4. Exit

Pilih (1/2/3/4)? 1

Masukkan dua bilangan yang akan dijumlahkan: 11 12

Hasil penjumlahan: 23

M E N U

1. Hitung Penjumlahan
2. Hitung Perkalian
3. Hitung Pembagian
4. Exit

Pilih (1/2/3/4)? 2

Masukkan dua bilangan yang akan dikalikan: 34 45

Hasil perkalian: 1530

M E N U

1. Hitung Penjumlahan
2. Hitung Perkalian
3. Hitung Pembagian
4. Exit

Pilih (1/2/3/4)? 3

Masukkan dua bilangan yang akan dibagikan: 1 2

Hasil pembagian: 0.5

M E N U

1. Hitung Penjumlahan
2. Hitung Perkalian
3. Hitung Pembagian
4. Exit

Pilih (1/2/3/4)? 4