

Modul O2 Terbimbing

Fungsi

1. **[Vocal Uppercase]** Buatlah program untuk menghitung banyaknya vokal uppercase (kapital) dalam suatu rangkaian karakter ['a'..'z'] dan ['A'..'Z']. Program yang dibuat memanfaatkan fungsi yang mengecek apakah suatu karakter merupakan vokal uppercase atau bukan.

```
function vowel_uppercase(kar : char) -> boolean
{ Mengembalikan true apabila kar adalah vokal uppercase, dan false apabila sebaliknya }
```

Masukan dan keluaran program adalah sebagai berikut:

Masukan terdiri dari sekumpulan karakter yang diakhiri oleh karakter titik '.

Keluaran berupa teks yang menyatakan banyaknya vokal uppercase.

Petunjuk: Gunakan tipe byte untuk deklarasi char.

Contoh masukan dan keluaran

No	Masukan	Keluaran
1	PertamA.	Jumlah vokal uppercase adalah 1
2	EkoNOmi.	Jumlah vokal uppercase adalah 2
3	INforMATika.	Jumlah vokal uppercase adalah 3

2. **[Diskon dan cashback Trost]** Sebuah toko perlengkapan penjaga dinding di Kota Trost mengadakan promo besar-besaran untuk menyambut pembukaan cabang baru di kota tersebut. Setiap pembelian perlengkapan penjaga dinding di toko tersebut akan diberikan poin yang dapat ditukar sebagai diskon dan cashback. Buatlah program untuk menghitung persentase diskon dan cashback yang diperoleh pembeli dengan masukan dan keluaran sebagai berikut:

Masukan terdiri dari status keanggotaan Survey Corps (boolean true atau false) dan lima bilangan bulat a, b, c, d, dan e yang menyatakan poin yang diperoleh setiap anggota.

Keluaran terdiri dari dua bilangan real yang menyatakan persentase diskon dan cashback yang diperoleh.

Program akan memanfaatkan empat buah fungsi. Aturan perhitungan diskon dan cashback terdapat pada spesifikasi fungsi yang diberikan.

```
function ganjil(a, b, c, d, e : integer) -> boolean  
{Mengembalikan true apabila masing-masing a, b, c, d, dan e adalah ganjil}
```

```
function genap(a, b, c, d, e : integer) -> boolean  
{Mengembalikan true apabila masing-masing a, b, c, d, dan e adalah genap}
```

```
function diskon(member : char, a, c, e: integer) -> real  
{ Mengembalikan nilai diskon sebesar 1.5x dari total a, c, dan e. Tambahkan  
25% dari nilai diskon awal apabila member, nilai diskon maksimal adalah  
50%}
```

```
function cashback(member : char, b, d: integer) -> real  
{ Mengembalikan nilai cashback sebesar 2.5x dari total b dan d. Tambahkan  
15% dari nilai cashback awal apabila member, nilai cashback maksimal  
adalah 40% }
```

program Promo

```
{ Program untuk menampilkan besaran diskon dan cashback yang diperoleh, apabila  
diberikan status membership m dan lima poin yang diperoleh a, b, c, d, dan e. Aturan  
pemberian diskon dan cashback:
```

- Diskon diberikan apabila lima poin bernilai ganjil semua, dan dihitung dari tiga poin a, c, dan e.
- Cashback diberikan apabila lima poin bernilai genap semua, dan dihitung dari dua poin b dan d.
- Diskon dan cashback diberikan apabila lima poin ada yang bernilai ganjil dan genap.}

Besaran diskon dan cashback dicetak dengan pembulatan 2 digit di belakang koma

Contoh masukan dan keluaran

No	Masukan	Keluaran
1	true 1 2 3 4 5	16.88 17.25
2	false 1 2 3 4 5	13.50 15.00
3	true 3 3 3 3 3	16.88 0.00
4	true 6 4 2 4 2	0.00 23.00

Modul O2 Mandiri

Fungsi

1. **[Persamaan matematika]** Apabila didefinisikan $f(x, y, z) = ((2x)/(x + y + z))$ dan $g(x, y) = ((1/(5x)) + y)$, maka buatlah subprogram fungsi berikut berdasarkan spesifikasi yang diberikan. Catatan: Perhatikan urutan parameter seperti terlihat pada contoh.

```
function fungsi_f(x, y, z : real) -> real
```

```
{ Fungsi menerima masukan x, y, dan z serta mengembalikan nilai real berdasarkan persamaan  $f(x, y, z) = ((2x)/(x + y + z))$  }
```

```
function fungsi_g(x, y : real) -> real
```

```
{ Fungsi menerima masukan x, dan y, serta mengembalikan nilai real berdasarkan persamaan  $g(x, y) = ((1/(5x)) + y)$  }
```

Selanjutnya kedua fungsi dipanggil dalam program dengan masukan dan keluaran sebagai berikut:

Masukan berupa tiga bilangan real b1, b2, dan b3. Ketiganya akan menjadi parameter aktual terhadap parameter formal x, y, dan z.

Keluaran berupa dua bilangan real hasil perhitungan fungsi f dan fungsi g.

Contoh masukan dan keluaran

No	Masukan	Keluaran
1	1.5 2.4 3.1	0.42857142857142855 2.4
2	100 45 50	1.0256410256410255 45

2. **[Kunai Kakashi]** Kakashi Hatake sedang berlatih melemparkan kunai. Kakashi melemparkan kunai dengan sudut elevasi **T** derajat dan kecepatan awal **V** m/s. Buatlah program dalam bahasa Go untuk menghitung:

- Waktu yang diperlukan kunai untuk mencapai titik tertinggi. Gunakan rumus $t = V \sin(T) / g$, dengan t adalah waktu, V adalah kecepatan, T adalah sudut elevasi, dan g adalah konstanta gravitasi.
- Jarak horizontal yang ditempuh kunai sebelum mencapai tanah. Gunakan rumus $x = V^2 \sin(2T) / g$, dengan x adalah jarak, V adalah kecepatan, T adalah sudut elevasi, dan g adalah konstanta gravitasi.
- Ketinggian maksimum yang dicapai oleh kunai. Gunakan rumus $y = V^2 \sin^2(T) / 2g$, dengan y adalah ketinggian, V adalah kecepatan, T adalah sudut elevasi, dan g adalah konstanta gravitasi.

Catatan: Konstanta gravitasi g adalah 9.8 m/s^2 .

Program dibuat dengan memanfaatkan fungsi-fungsi berikut:

function konversiDerajatkeRadian($T : \text{real}$) -> **real**

{ Mengembalikan nilai konversi sudut dari T derajat ke radian. Sudut satu derajat setara dengan satu $\pi/180$ radian. Gunakan nilai π dari `math.Pi` }

function waktu($V, T : \text{real}$) -> **real**

{ Mengembalikan nilai waktu yang diperlukan untuk mencapai posisi tertinggi, jika kunai dilempar dengan sudut T derajat dan kecepatan awal $V \text{ m/s}$. Gunakan fungsi **konversiDerajatkeRadian** untuk mengubah sudut dari derajat ke radian. }

function jarak($V, T : \text{real}$) -> **real**

{ Mengembalikan nilai jarak horizontal yang ditempuh kunai sebelum mencapai tanah, jika kunai dilempar dengan sudut T derajat dan kecepatan awal $V \text{ m/s}$. Gunakan fungsi **konversiDerajatkeRadian** untuk mengubah sudut dari derajat ke radian. }

function ketinggian(V, T : real) -> real

{ Mengembalikan nilai ketinggian maksimum yang dicapai kunai, jika kunai dilempar dengan sudut T derajat dan kecepatan awal V m/s. Gunakan fungsi **konversiDerajatkeRadian** untuk mengubah sudut dari derajat ke radian. }

Masukan dan keluaran program adalah sebagai berikut:

Masukan berupa dua bilangan real V dan T yang menyatakan kecepatan awal lemparan dan sudut elevasi dalam derajat.

Keluaran berupa tiga bilangan real yang menyatakan waktu tempuh, jarak horizontal, dan ketinggian maksimum. Masing-masing dibulatkan dalam dua digit di belakang koma.

Contoh masukan dan keluaran

No	Masukan	Keluaran
1	60 30	3.06 318.13 45.92
2	100 45	7.22 1020.41 255.10
3	40 40	2.62 160.78 33.73