

Minggu 08 Pertemuan 1

Tipe Bentuk dan Array

Algoritma Pemrograman – CII1F4

Fakultas Informatika

2021

Outline

Tipe Bentuk dan Operasinya

Array dan Operasinya

Kombinasi Array dan Tipe Bentuk

Latihan Soal

TIPE BENTUKAN

1. Tipe data dasar >> telah terdefinisi (Contoh: integer, real, string, boolean dan char)
2. **Tipe Bentuk** >> suatu **TYPE** yang **dibuat** (oleh pemrogram) dari beberapa komponen/field bertipe tertentu yang mempunyai makna semantic.
3. Contoh tipe bentuk :
 - Tipe Waktu (jam, menit, dan detik)
 - Tipe Orang (nama, gendre, alamat, agama dan biodata lainnya)

4. Deklarasi dari Tipe Bentuk

Karena yang deklarasi adalah tipe data maka, perlu ada **variable dengan tipe bentuk** tersebut

```
type <type_name> <  
    <field_1> : <data type>,  
    <field_2> : <data type>,  
    ...  
>  
<variable_name> : <type_name>
```

Contoh:

```
type date<  
    day: integer,  
    month : string,  
    year  : integer >
```

```
tgl : date
```

Assignment dan Operasi Tipe Data

Kamus

```
type date<  
    day    : integer,  
    month  : string,  
    year   : integer >
```

```
tgl : date
```

algoritma

```
input(tgl.day)  
tgl.month <- "Agustus"  
tgl.year  <- 1900 + 45  
print("Indonesia merdeka pada tanggal",  
tgl.day,"bulan", tgl.month,  
"dan tahun",tgl.year)
```

```
endprogram
```

1. Akses dilakukan melalui field yang didefinisikan
2. Operasi dilakukan sesuai tipe data dari fieldnya.
3. Pahami hirarki berikut:

Keterangan
date adalah tipe data, bukan variable
tgl bertipe date
tgl.day bertipe integer
tgl.month bertipe String
tgl.year bertipe integer



Contoh Soal

Sebuah program digunakan untuk menyimpan data suatu balok beserta volume dan luas selimutnya dan menampilkannya. Gunakan tipe bentukan balok yang digunakan untuk menyimpan data-data tersebut.

Jawaban: data balok >> volume, luas selimut + panjang, lebar, tinggi

```
program contohSoal
kamus
    type balok<
        p,l,t    : integer,
        luas,volume : integer >
    cube : balok
algoritma
    input(cube.p, cube.l, cube.t)
    cube.volume <- cube.p * cube.l * cube.t
    cube.luas <- 2 * (cube.p*cube.l + cube.p * cube.t + cube.l * cube.t)
    print(cube.p, cube.l, cube.t, cube.luas, cube.volume)
endprogram
```



Implementasi dalam GoLang

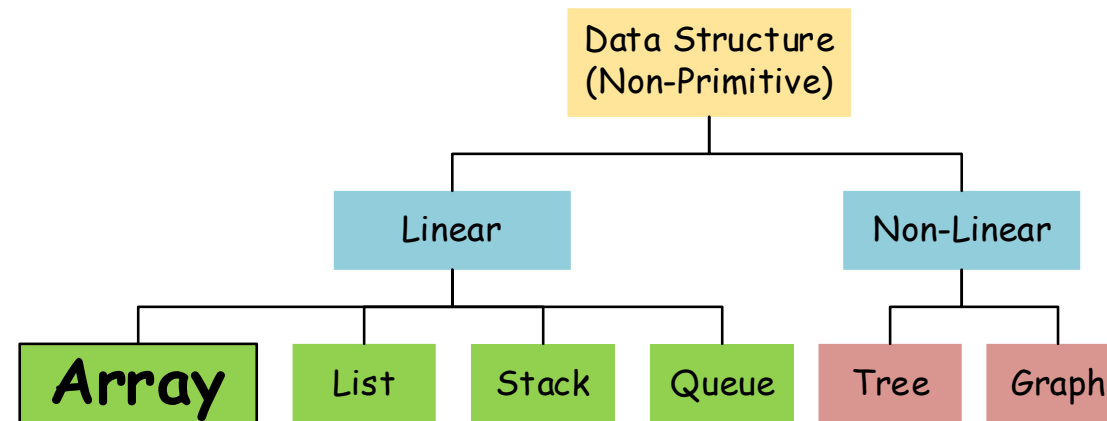
- Tidak ada perbedaan yang signifikan dalam bahasa Go
- Deklarasi tipe bentukan ditulis di luar blok dari subprogram

```
package main
import "fmt"

type balok struct{
    p,l,t int
    luas,volume int
}

func main(){
    var cube balok
    fmt.Scanln(&cube.p, &cube.l, &cube.t)
    cube.volume = cube.p * cube.l * cube.t
    cube.luas = 2 * (cube.p * cube.l + cube.p * cube.t + cube.l * cube.t)
    fmt.Print(cube.p, cube.l, cube.t, cube.luas, cube.volume)
}
```

ARRAY



Digunakan untuk menyimpan **kumpulan data**.

1. Array dapat menyimpan data/elemen dengan **tipe homogen/sejenis**.
2. Element **terurut berdasarkan indeks**
3. **Banyaknya elemen adalah tetap**.
4. Index adalah nomor untuk mengakses elemen dari array.
5. **Tipe indeks adalah integer atau character abjad**.

Contoh:

- ✓ T1 dan T2 adalah Array
- ✓ Jumlah elemen T1 = 5, Indeks = 1, 2, 3, 4, 5
- ✓ Jumlah elemen T2 = 3, Indeks = 0, 1, 2
- ✓ Tipe T1 : Array of Integer, dan Tipe T2 : Array of String

(Gambar hanyalah ilustrasi)

T1

30	12		17	
1	2	3	4	5

T2

"Raden"	0
"King"	1
"Sultan"	2

Deklarasi dan Akses dari Array

Deklarasi array `<variable name> : array [start..end] of <data type>`

di mana indeks array adalah `start` hingga `end`. contohnya:

Kamus

T1 : array [1..5] of integer {cara 1. deklarasi langsung sebagai variabel}

type arrString : array [0..2] of string {cara 2. deklarasi sebagai alias dengan type}

T2 : arrString {arrString adalah tipe data}

Algoritma

<u>input</u> (T1)	{SALAH, harus spesifik elemen dari T1 yang akan di-assign}
<u>input</u> (T1[1])	{BENAR, elemen spesifik dan indeks valid}
T2[0] <- "Raden"	{BENAR, T2 adalah array of string, T2[0] adalah string}
T1[4] <- 2	{BENAR, T1[4] adalah integer}
<u>input</u> (T2[10])	{SALAH, indeks tidak valid}
T2[T1[4]] <- "Sultan"	{BENAR, T1[4] adalah integer dan bernilai 2, indeks valid}
T1[3] <- T[1] + T[4]	{BENAR, operasi integer}
print(T2)	{SALAH, harus spesifik elemen dari T2 yang akan dicetak}
print(T1[1], T2[2])	{BENAR, elemen spesifik dan indeks valid}

Endprogram

Pengisian dan mencetak Array

1. Indeks terurut berdasarkan enumerasinya
2. Bisa memanfaatkan perulangan.
3. Indeks array 0, 1, 2, ..., 2020

Contoh:

Kamus

```
constant NMAX : integer = 2021 {konstanta}
type tabInt : array [0..NMAX-1] of integer
```

```
Procedure mengisi(in/out T:tabInt, in N:integer)
{IS. terdefinisi bilangan bulat N
 FS. array T berisi N buah bilangan dari input}
```

kamus

```
k : integer
```

algoritma

```
k <- 0
```

```
while k <= N do
```

```
    input(T[k])
```

```
    k <- k + 1
```

```
endwhile
```

```
endprocedure
```

```
Procedure cetak(in T:tabInt, N:integer)
```

```
{IS. terdefinisi array T yang berisi N elemen
```

```
 FS. menampilkan isi dari array T}
```

kamus

```
k : integer
```

algoritma

```
k <- 0
```

```
while k <= N do
```

```
    print(T[k])
```

```
    k <- k + 1
```

```
endwhile
```

```
endprocedure
```

Catatan: Nilai N itu menunjukkan bahwa dari index 0 s.d 2020, hanya index 0 s.d. N yang berisi data, di mana $0 \leq N \leq NMAX$

Kombinasi Array dan Tipe Bentukan

```
constant NMAX : integer = 2021
```

```
type mahasiswa <
```

```
    nama, nim, jurusan : string,
```

```
    nilai : real >
```

{variasi 1. deklarasi sebagai alias}

```
type arrMhs : array [0..NMAX-1] of mahasiswa
```

```
nMhs : integer {jumlah mahasiswa}
```

```
Mhs1 : arrMhs
```

{variasi 2. deklarasi sebagai tipe bentukan}

```
type mhsType <
```

```
    info : : array [0..NMAX-1] of mahasiswa
```

```
    mMhs : integer {jumlah mahasiswa} >
```

```
Mhs2 : mhsType
```

Contoh dalam pembuatan **table** mahasiswa

- Variasi 1, jumlah elemen array dipisah, Sehingga dampaknya adalah jumlah elemen harus selalu disertakan Bersama arraynya
- Variasi 2, jumlah elemen array dimasukkan ke dalam tipe bentukan. Lebih rumit, tetapi jumlah elemen array selalu ada bersama arraynya

Keterangan Variasi 1

arrMhs adalah tipe array of mahasiswa

Mhs1 bertipe arrMhs

Mhs1[0] bertipe mahasiswa

Mhs1[NMAX-1].nama bertipe string

Keterangan Variasi 2

mhsType adalah tipe bentukan

Mhs2.info bertipe array of mahasiswa

Info adalah variable, **bukan tipe data**

Mhs2.info[0] bertipe mahasiswa

Mhs2 bertipe mhsType

Mhs2.info[0].nama bertipe string

**KEEP
CALM
AND
STUDY
HARD**

Sesi Latihan 😊



MOTIVATIONAL
PICTURE QUOTES.COM

Image <https://id.pinterest.com/pin/498703358729526087/>

Soal 1 : Jarak Titik

Sebuah program digunakan untuk menghitung jarak antara dua buah titik

Masukan berupa 4 nilai yang menyatakan sumbu x dan y dari masing-masing titik.

Keluaran sebuah bilangan real yang menyatakan jarak antara 2 titik tersebut

Buatlah dengan melengkapi dan menggunakan subprogram berikut

```
function jarak(P1, P2 : titik) -> real
```

```
{diterima nilai sumbu dari titik P1 dan P2, untuk mengembalikan jarak dari P1 dan P2}
```

```
function akar(x : real) -> real
```

```
{diterima nilai suatu x untuk mengembalikan akar kuadrat dari x, fungsi ini terdefinisi dan tinggal digunakan saja}
```

jarak antara titik (a,b) dan (c,d) adalah

$$jarak = \sqrt{(a - c)^2 + (b - d)^2}$$



Soal 2 : Array

1. Buatlah sebuah tipe array of integer yang berkapasitas 256!
2. Buatlah procedure untuk reverse isi dari array!
3. Buatlah function untuk mengecek apakah suatu array memiliki pola palindrom! Nilai elemen membentuk pola simetris.
Contoh A = [10, 20, 30, 20, 10], B = [15, 75, 75, 15] dan C = [100]
4. Buatlah procedure untuk mengisi array untuk menghasilkan array ganjil dan array genap.





Soal 3 : Pertandingan Bola

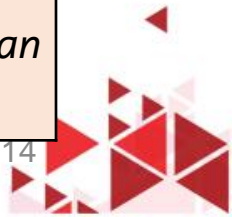
Sebuah program digunakan untuk menyimpan jumlah kemenangan pertandingan dalam setiap tahunnya.

Masukan terdiri dari 3 baris, di mana setiap barisnya adalah data kemenangan dari suatu tim tertentu. Data kemenangan berupa beberapa bilangan yang dipisahkan oleh spasi, yang menyatakan jumlah kemenangan suatu tim bola setiap tahunnya. Masukan pada setiap barisnya berakhir apabila bilangan adalah negatif.

Keluaran berupa 3 bilangan yang menyatakan rata-rata kemenangan dari 3 tim tersebut setiap tahunnya.

Buatlah dengan melengkapi dan menggunakan subprogram berikut

```
procedure inputData(in/out t:tabGol, n: integer)  
{I.S. data kemenangan suatu tim telah siap pada piranti masukan  
 F.S. t berisi n data kemenangan suatu tim.}  
  
function rata(t: tabGol, n : integer) → real  
{diberikan array t yang berisi n data kemenangan, untuk mengembalikan rata-rata kemenangan}  
  
program PertandinganBola  
{program untuk menyimpan data kemenangan 3 tim bola, dan menampilkan rata-rata kemenangan  
 setiap timnya}
```





TERIMA KASIH

