

## Tugas Pendahuluan Modul 07

### Array

#### Instruksi Pengerjaan

1. Tugas ini bersifat Individu.
2. Jawaban tugas dikirim dalam bentuk kode dalam bahasa Go.
3. Dilarang melakukan tindakan plagiasi ataupun menggunakan bantuan tool Generative AI.
4. Tugas dikumpulkan pada LMS dengan deadline Selasa, 8 April 2025 pukul 06.00 WIB
5. Tugas ini merupakan latihan untuk bisa mengikuti praktikum dengan baik.
6. Contact Person Praktikum Algoritma Pemrograman:
  - 085847990203 (Gede Bagus)
  - 085212528394 (Arief Rahman)
  - 085105303555 (Kayyisa Zulfa)

#### Soal Tugas Pendahuluan

1. [Translasi] Translasikan pseudocode di bawah ini ke dalam bahasa Go.

```
program BacaCetak
kamus (lokal)
    constant NMAX : integer = 10
    A : array[0..NMAX-1] of string
    i, n : integer
algoritma
    input(n)
    for i <- 0 to n-1 do
        input(A[i])
    endfor
    for i <- 0 to n-1 do
        output(A[i])
```

```
endfor
endprogram
```

Jalankan dan periksa apakah sudah sesuai dengan masukan dan keluaran di bawah ini.

No	Masukan	Keluaran
1	5 Telkom University sinergi bangun negeri	Telkom University sinergi bangun negeri
2	1 telyutizen	telyutizen

2. **[Baca dan Cetak Array 1]** Buatlah program dalam bahasa Go untuk membaca data berupa bilangan bulat dan memasukkannya ke dalam array yang memiliki kapasitas maksimum 10 dengan mendeklarasikan sebuah *tipe bentukan alias* **tabInt** secara global. Setelah itu cetaklah semua elemen array itu sesuai template pencetakan. Pembacaan dan pencetakan bilangan dilakukan dengan prosedur **bacaData()** dan **cetakData()**.

```
program baca_dan_cetak_array_1
kamus (global)
    constant NMAX : integer = 10
    type tabInt : array[0..NMAX-1] of integer
kamus (lokal)
    data : tabInt
    banyakData : integer
algoritma
    bacaData(data, banyakData)
```

```
cetakData(data, banyakData)

endprogram
```

```
procedure bacaData(in/out A : tabInt, in/out n : integer)
{ IS:      Array bilangan bulat (A) dan banyak elemen (n) terdefinisi
  sembarang
  Proses: a) Membaca banyak data (n). Jika n > NMAX, maka n bernilai
           NMAX.
          b) Membaca bilangan bulat sebanyak n dan memasukkannya ke
              dalam array bilangan bulat (A).
  FS:      Array bilangan bulat (A) berisi nilai. Banyak elemen (n)
           berisi nilai NMAX, jika n > NMAX }
```

```
procedure cetakData(in A : tabInt, in n : integer)
{ IS: Array bilangan bulat (A) dan banyak elemen (n) terdefinisi
  FS: Mencetak sebanyak n elemen array bilangan bulat (A) dengan
      format:
      <e0> <e1> <e2> ... <en-1>. Jika array kosong, maka cetak: "Array
      kosong" }
```

Masukan dan keluaran program adalah sebagai berikut:

**Masukan:** Pada baris pertama sebuah bilangan bulat n yang menyatakan banyaknya elemen array. Pada baris kedua bilangan bulat sebanyak n buah, jika n > 0.

**Keluaran:** Elemen array sebanyak n jika ada, jika elemennya tidak (array kosong) cetak string "Array kosong".

#### Contoh masukan dan keluaran

No	Masukan	Keluaran
1	5 11 23 34 45 67	11 23 34 45 67
2	11	45 76 34 28 12 45 76 34 28 12

No	Masukan	Keluaran
	45 76 34 28 12 45 76 34 28 12 33	
3	0	Array kosong

Pada testcase nomor 2,  $n > NMAX$ , sehingga  $n = NMAX$ . Oleh karena itu walaupun bilangan bulat yang diinput sebanyak 11 yang masuk ke dalam array hanya 10 bilangan bulat.

3. **[Baca dan Cetak Array 2]** Buatlah program dalam bahasa Go untuk membaca data berupa bilangan bulat dan memasukkannya ke dalam array yang memiliki kapasitas maksimum 5 dengan mendeklarasikan sebuah *tipe bentukan struktur* **tabInt** secara global. Setelah itu cetaklah semua elemen array itu. Pembacaan dan pencetakan bilangan dilakukan dengan prosedur **bacaData()** dan **cetakData()**. Pada program utama akan dilakukan 6x pembacaan data dan 1x pencetakan data.

```

program baca_dan_cetak_array_2

kamus (global)
    constant NMAX : integer = 5
    type tabInt <
        info : array[0..NMAX-1] of integer
        n : integer
    >

kamus (lokal)
    data : tabInt

algoritma
    bacaData(data)
    bacaData(data)
    bacaData(data)
    bacaData(data)
    bacaData(data)

```

```

    bacaData(data)

    cetakData(data)

endprogram

```

**procedure** bacaData(**in/out** A : tabInt)

{ IS: A.info[i] adalah field yang menampung data array, sedangkan A.n adalah field untuk menampung banyaknya elemen data. Kedua field itu terdefinisi sembarang yang berarti bisa kosong atau berisi nilai.

FS: Data array A.info bertambah satu elemen dan A.n bertambah satu nilai. keduanya bertambah, selama belum melebihi kapasitas maksimum array (NMAX) }

**procedure** cetakData(**in** A : tabInt)

{ IS: Array A terdefinisi sembarang, yang berarti bisa berisi nilai atau kosong

FS: Mencetak sebanyak A.n elemen array dengan format:

<e<sub>0</sub>> <e<sub>1</sub>> <e<sub>2</sub>> ... <e<sub>n-1</sub>>". }

#### Contoh masukan dan keluaran

No	Masukan	Keluaran
1	11 22 33 44 55 66	11 22 33 44 55
2	4 0 2 5 7 1 4	4 0 2 5 7 1

4. **[Penjumlahan dan Rata-rata]** Buatlah program dalam bahasa Go untuk membaca data berupa serangkaian bilangan bulat dan memasukkannya ke dalam array yang memiliki kapasitas maksimum 10 dengan mendeklarasikan sebuah tipe bentukan struktur **tabInt** secara global. Setelah itu cetaklah semua elemen array itu. Pembacaan dan pencetakan karakter dilakukan dengan prosedur baca dan cetak. Buatlah pula fungsi untuk menghitung jumlah dan rata-rata nilai yang terdapat pada array.

```

program JumlahdanRataRata

kamus (global)

    constant NMAX : integer = 10

    type tabInt : array[0..NMAX-1] of integer

kamus (lokal)

    data : tabInt

    nData : integer

algoritma

    baca(data)

    cetak(data)

    output(penjumlahan(data, nData))

    output(rataRata(data, nData))

endprogram

```

```

procedure baca(in/out A : tabInt, in/out n : integer)
{ IS:      Array bilangan bulat (A) dan banyak elemen (n) terdefinisi
          sembarang.

  Proses: Setiap bilangan yang terbaca ditampung dalam sebuah
          variabel. Jika bilangan tersebut negatif, maka ubah menjadi
          bilangan positif dan masukkan ke dalam array. Sedangkan
          setiap bilangan positif selalu dimasukkan ke dalam array.
          Pembacaan berakhir bila terbaca bilangan 0 atau melebihi
          kapasitas maksimum array.

  FS:      Array bilangan bulat (A) berisi bilangan bulat positif.
          Banyak elemen (n) berisi nilai NMAX, jika n > NMAX }

```

```

procedure cetak(in A : tabInt, in n : integer)
{ IS: Array bilangan bulat (A) dan banyak elemen (n) terdefinisi

```

FS: Mencetak sebanyak n elemen array bilangan bulat (A) dengan format:  $\langle e_0 \rangle \langle e_1 \rangle \langle e_2 \rangle \dots \langle e_{n-1} \rangle$ .

```
function jumlah(A : tabInt, n : integer) -> integer
{ Mengembalikan jumlah dari nilai n elemen array A }
function rataRata(A : tabInt, n : integer) -> integer
{ Mengembalikan rata-rata dari nilai n elemen array A. Panggil fungsi
jumlah untuk melakukan perhitungan jumlah nilai n elemen. }
```

Masukan dan keluaran pada program adalah sebagai berikut:

**Masukan** berupa serangkaian bilangan bulat (positif dan negatif) yang diakhiri nilai 0 sebagai sentinel. Jika terbaca bilangan negatif, ubah ke dalam bilangan positif dan masukkan ke dalam array. Sedangkan bilangan bulat langsung dimasukkan ke dalam array. Jika bilangan bulat yang disediakan di piranti masukan melebihi kapasitas maksimum array, maka bilangan itu tidak dibaca.

**Keluaran** berupa seluruh elemen array, jumlah, dan rata-ratanya.

#### Contoh masukan dan keluaran

No	Masukan	Keluaran
1	10 2 23 -34 12 0	10 2 23 34 12 81 16.2
2	1 2 3 -4 5 6 7 -8 9 10 -11 0	1 2 3 4 5 6 7 8 9 10 55 5.5

Pada testcase ke-1, bilangan -34 diubah ke 23 lalu dimasukkan ke dalam array data. Pada testcase ke-2, bilangan -4 dan -8 diubah menjadi 4 dan 8 lalu dimasukkan ke dalam array data. Namun bilangan -11 tidak dibaca karena sudah melebihi kapasitas maksimum array, yaitu 10 elemen.