

**eSinaukoe**  
learn with mas anto

## Tipe, Variabel dan Operator Algoritma Pemrograman


**Agus Sumaryanto, S.Kom**  
**mas.anto72@gmail.com**

<http://learning.mas-anto.com>


1

**eSinaukoe**  
learn with mas anto


## Materi



**PENULISAN ALGORITMA**



**TIPE DATA, VARIABEL OPERATOR**




**NAMA DAN EKSPRESI**

<http://learning.mas-anto.com>

2

**eSinaukoe**  
learn with mas anto



## PENULISAN ALGORITMA

<http://learning.mas-anto.com>

3

**eSinaukoe**  
learn with mas anto

## PENULISAN ALGORITMA

- Algoritma ditulis dalam bentuk terstruktur
- Masing-masing bagian ditandai dengan identitas tertentu dan aturan tertentu.
- Format Penulisan Algoritma :
  - Judul
  - Deklarasi
  - Algoritma

<http://learning.mas-anto.com>

4

**eSinaukoe**  
learn with mas anto

## PENULISAN ALGORITMA

**JUDUL ALGORITMA**

Penjelasan tentang Algoritma, yang berisi uraian singkat mengenai apa yang dilakukan oleh Algoritma.

**DEKLARASI**

Semua nama tetapan (konstanta), tipe, peubah (variabel), fungsi dan prosedur yang ada dalam Algoritma didefinisikan disini.

**ALGORITMA**

Semua langkah/aksi/proses algoritma di tulis pada bagian ini

<http://learning.mas-anto.com> 5

**eSinaukoe**  
learn with mas anto

## Contoh Penulisan

**Menghitung\_Luas\_Lingkaran**

Algoritma menghitung Luas Lingkaran berdasarkan nilai Jari2 yang dimasukkan lewat perangkat masukan

**DEKLARASI**


```
Const pi=3.14
Jari2 : Integer
Luas : Real
```

**ALGORITMA**

```
Read(Jari2)
Luas ← pi * Jari2 * Jari2
Write(Luas)
```

<http://learning.mas-anto.com> 6

**eSinaukoe**  
learn with mas anto



## TIPE DATA, VARIABEL OPERATOR

<http://learning.mas-anto.com> 7

**eSinaukoe**  
learn with mas anto

## Tipe Data

- Tipe data menyatakan pola penyajian data dalam komputer
- Mendefinisikan tipe data berarti :
  - Menentukan nama tipe data
  - Menentukan nilai yang dikandung oleh nama
  - Penulisan dalam bentuk tetapan (konstanta)
  - Operator dapat dioperasikan pada obyek bertipe tersebut

<http://learning.mas-anto.com> 8

## eSinaukoe<sup>TM</sup> learn with mas anto

# Tipe Data (lanj.)

Kelompok Tipe Data :

1. Tipe Dasar (bilangan logik, bilangan bulat, real, karakter)
2. Tipe Bentuk (gabungan tipe dasar dan tipe bentuk lain)

<http://learning.mas-anto.com> 9

## eSinaukoe<sup>TM</sup> learn with mas anto

# Tipe Data Dasar

TIPE DATA	NILAI	OPERATOR
Logika (boolean)	True, false	AND, OR, NOT, XOR
Bil.Bulat (integer)	-100, 0, 290	+, -, *, MOD, DIV >, <, >=, <=, <>
Bil.Nyata (real)	-67.7, 0, 2.8	+, -, *, / >, <, >=, <=, <>
Karakter (char)	'1', 'a', 'A'	>, <, >=, <=, <>

<http://learning.mas-anto.com> 10

## eSinaukoe<sup>TM</sup> learn with mas anto

# Tipe Bentuk

TIPE DATA	NILAI	OPERATOR
String (string)	'Jakarta', '1000', 'B1010'	Penyambungan >, <, >=, <=, <>
Rekaman (record)	Type MHS =record < Nim:string, nama:string, nilai:real> dll	

<http://learning.mas-anto.com> 11

## eSinaukoe<sup>TM</sup> learn with mas anto

# Operator

**Operator** : memungkinkan dua buah nilai dapat melakukan operasi tertentu

P	Q	Not P	Not Q	P and Q	P or Q
True	True	False	False	True	True
True	False	False	True	False	True
False	True	True	False	False	True
False	False	True	True	False	False

**Operasi Matematika** :  
 $A + B$ ,  $28 \bmod 3$ ,  $35 \text{ div } 3$ ,  $A * B$ ,  $A - B$ ,  $A/B$

**Operasi Relasional (true/false)**:  
 $10 > 9$ ,  $A >= 10$ ,  $6 < 8$ ,  $8 <= 7$ ,  $3 <> 4$

**Operasi Penyambungan** :  
 'abc' + 'def' = 'abcdef'

<http://learning.mas-anto.com> 12

## Assignment (penugasan)

- Bentuk pemberian nilai ke dalam sebuah peubah/variabel
- Menggunakan notasi " $\leftarrow$ "
  - Contoh :  
A  $\leftarrow$  10 cara membacanya A diberikan nilai 10
- Pemberian nilai harus sesuai dengan tipe data yang didefinisikan untuk variabel tersebut.

<http://learning.mas-anto.com>

13

## Contoh Assignment

```

01| Deklarasi:
02| a,b : integer
03| c,d : real
04| nama1, nama2 : string
05| huruf : char
06| ketemu : boolean
07|
08| Deskripsi:
09| a ← 3 (boleh)
10| b ← a + 4 (boleh)
11| c ← 3 (boleh)
12| a ← 0.5 (tidak boleh karena a tidak dapat menampung real)
13| a ← b + c (tidak boleh karena b+c bertipe real)
14| huruf ← '?' (boleh)
15| nama1 ← 'Ani' ← (boleh)
16| nama2 ← nama1 + huruf (boleh)
17| nama1 ← 'a' (boleh)
18| huruf ← nama1 (tidak boleh karena nama1 bertipe string)
19| ketemu ← true (boleh)
20| ketemu ← 'true' (tidak boleh karena 'true' bertipe string)

```

<http://learning.mas-anto.com>

14

## Input Statement

- Input atau masukan disimbolkan dengan pernyataan "**Read**"
- Pernyataan meminta user untuk memberikan sebuah nilai ke dalam variabel sesuai dengan tipe datanya.
- Semua tipe data dapat diberikan nilai masukan melalui alat masukan kecuali tipe data Boolean (bilangan logik)

<http://learning.mas-anto.com>

15

## Contoh Input Statement

```

01| Algoritma ContohPengeinputan
02| (contoh penggunaan read untuk menginput sejumlah data dari user)
03| Deklarasi:
04| a,b,c : integer
05| d : real
06| e : char
07| f : string
08| g : boolean
09| Deskripsi:
10| (menginput data bil bulat, kemudian dimasukkan ke variabel a)
11| Read (a)
12| (menginput 2 data integer, dimasukkan ke variabel b dan c)
13| read (b,c)
14| (menginput data pecahan, kemudian dimasukkan ke variabel d)
15| read (d)
16| (menginput data bulat dan pecahan)
17| read (a,d)
18| (menginput data bulat, pecahan, dan 1 karakter)
19| read (a,d,e)
20| (menginput data string)
21| read (f)
22| (berikut ini adalah suatu kesalahan)
23| read (g) (data boolean tidak dapat diinputkan)

```

<http://learning.mas-anto.com>

16

## Output Statement

- Output, mencetak informasi dari algoritma.
- Pernyataan output menggunakan **“Write”**

```

01) Algoritma ContohOutput
02) (tujuan contoh-contoh penggunaan write untuk mencetak data)
03) Deklarasi:
04) const k = 0.5
05) a : Integer
06) b : Real
07) c : Char
08) d : String
09) e : Boolean
10) Deskripsi:
11) read(a) (menginput sebuah data Integer)
12) write(a) (mencetak data hasil input tadi)
13) write(b) (tidak boleh, karena variabel b belum ada isinya)
14) c ← 'y'
15) d ← 'strik'
16) write(d:0) (mencetak isi variabel d dan 0)
17) write('Buat kulia di ',d) (mencetak Buit kulia di strik)
18) write(5 + 6) (mencetak bilangan 11, yaitu hasil dari 5+6)
19) e ← 2 = 1 + 3
20) write(e) (mencetak FALSE karena = bukanlah false)
21) write(k) (mencetak 0.5 karena k bertipe 0.5)
  
```

<http://learning.mas-anto.com>

17



## Nama & Ekspresi

<http://learning.mas-anto.com>

18

## Nama

- Segala sesuatu harus diberi nama untuk membedakan satu dengan yang lain
- Yang harus diberi nama :
  - Peubah (variabel) – untuk nilai yang dinamis
  - Tetapan (konstanta) – untuk nilai tetap
  - Tipe
  - Nama Prosedur
  - Nama Fungsi

<http://learning.mas-anto.com>

19

## Aturan Pemberian Nama

- Harus dimulai dengan alfabet, tidak boleh menggunakan angka atau karakter khusus
- Tidak dibedakan huruf kapital atau huruf kecil
- Penyusun nama hanya boleh alfabet, angka dan **“\_”** (underscore).
- Tidak dipisahkan dengan spasi
- Panjang nama tidak dibatasi

<http://learning.mas-anto.com>

20

**eSinaukoe**  
learn with mas anto

## Contoh Pemberian Nama

**Penulisan nama yang benar :**

Nama  
Tgl1  
nm\_siswa  
namamahasiswa

**Penulisan nama yang salah :**

1nama  
nama siswa  
gaji@bersih

<http://learning.mas-anto.com> 21

**eSinaukoe**  
learn with mas anto

## Ekspresi

- Ekspresi terdiri dari operand dan operator
- Operand adalah harga yang dioperasikan dengan operator tertentu.
- Operand dapat berupa nama peubah, nama tetapan, atau tetapan.
- Terdapat dua macam ekspresi , ekspresi numerik dan ekspresi boolean
- Prioritas operator tanpa tanda kurung  
**pangkat ->kali/bagi->tambah/kurang**

<http://learning.mas-anto.com> 22

**eSinaukoe**  
learn with mas anto

## Ekspresi Numerik

- Ekspresi harus memungkinkan sebuah nilai dapat dioperasikan.
- Contoh :  
Deklarasi  
a, b, c : real  
d : integer  
i, j, k : integer

$k \leftarrow (i + j) \text{ div } 2$   
(hasil ekspresi  $(i + j) \text{ div } 2$  disimpan dalam peubah k)

<http://learning.mas-anto.com> 23

**eSinaukoe**  
learn with mas anto

## Notasi Ekspresi Numerik

Dikenal 3 jenis notasi ekspresi

- 1. Infix**  
format : operand1 **operator** operand2  
contoh :  $a + b/c*d - e*f$
- 2. Prefix**  
format : **operator** operand1 operand2  
contoh :  $*+a/bc-d*e*f$
- 3. Postfix**  
format : operand1 operand2 **operator**  
contoh :  $abc/+def*-*$

<http://learning.mas-anto.com> 24

## Notasi Relasional

- Membandingkan 2 buah nilai dan asumsi yang didapat adalah **True** atau **False**
- Menggunakan simbol  $>$ ,  $<$ ,  $>=$ ,  $<=$ ,  $=$ ,  $<>$

Contoh :

**$10 > 8$ ,  $8=8$ ,  $9<>7$  dll**