

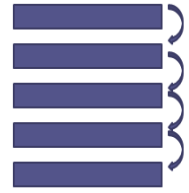
# Algoritma dan Struktur Data I

Mia Fitriawati, S.Kom, M.Kom

www.fgpt.info

## Runtunan

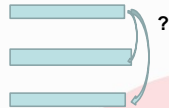
- Runtunan terdiri dari satu atau lebih pernyataan
- Tiap pernyataan dikerjakan secara berurutan sesuai urutannya
- Urutan instruksi menentukan keadaan akhir algoritma
- Bila urutan diubah, maka hasil akhirnya juga berubah



www.fgpt.info

## Pemilihan

- Instruksi dikerjakan jika memenuhi kondisi tertentu
- Contoh kasus *traffic light*
- Jika lampu berwarna merah maka kendaraan harus berhenti



www.fgpt.info

## Pemilihan (2)

- Pernyataan:

jika lampu traffic light berwarna merah, maka berhenti

- Pernyataan pemilihan:

if kondisi then  
aksi

www.fgpt.info

## Pemilihan(3)

if kondisi then  
aksi 1

else  
aksi 2

\*else artinya kalau tidak.

www.fgpt.info

## Pengulangan

- Salah satu kelebihan komputer adalah kemampuannya untuk mengerjakan pekerjaan yang sama berulang kali tanpa lelah.
- Ada 3 notasi pengulangan di algoritma:
  1. for
  2. repeat
  3. while

www.fgpt.info

## Tipe Data

Tipe data terdiri dari tipe:

- Tipe dasar
  - Tipe yang dapat langsung dipakai (disediakan oleh bahasa pemrograman)
  - Contoh: boolean, integer, real, char, **string**
- Tipe bentukan
  - Tipe yang didefinisikan sendiri oleh pemrogram
  - Tipe yang dibentuk dari tipe dasar atau dari tipe bentukan lain yang sudah didefinisikan
  - Contoh: tipe dasar yang diberi nama tipe baru, record

www.fppt.info

## Tipe Dasar(2)

NAMA	DOMAIN	KONSTANTA	OPERATOR
boolean	true, false	true, false	and, or, xor, not
integer	tak terbatas	ditulis tanpa titik desimal	Aritmatika : +, -, *, div, mod Perbandingan : =, <, <=, >, >=
real	tak terbatas	ditulis dengan titik desimal	Aritmatika : +, -, *, / Perbandingan : =, <, <=, >, >=
char	huruf, angka, tanda baca, operator aritmatika, karakter khusus	harus diapit tanda petik tunggal ('...')	Perbandingan : =, <, <=, >, >=
string	Deretan karakter	harus diapit tanda petik tunggal ('...')	Penyambungan : + Perbandingan : =, <, <=, >, >=

www.fppt.info

## Tipe Dasar(3) Boolean

a	not a
true	false
false	true

a	b	a and b	a or b	a xor b
true	true	true	true	false
true	false	false	true	true
false	true	false	true	true
false	false	false	false	false

www.fppt.info

## Tipe Bentukan

- Tipe dasar yang diberi nama tipe baru
  - Nama baru untuk tipe dasar menggunakan kata kunci **type**
  - Domain nilai, cara menulis konstanta, dan operasi-operasi yang dapat dijalankan pada tipe baru tersebut tidak berubah, sama seperti tipe dasarnya.
  - Contoh: **type** BilanganBulat : **integer**
- Rekaman (record)
  - Rekaman disusun atas satu atau lebih field
  - Tipe field menyimpan data dan tipe dasar tertentu atau dari tipe bentukan lain yang sudah didefinisikan sebelumnya
  - Nama rekaman ditentukan oleh pemrogram
  - Rekaman disebut juga tipe terstruktur

www.fppt.info

## Contoh Record

### DEKLARASI

```
type MataKuliah : record <KodeMK : string, {kode matakuliah}
  NamaMK : string, {nama matakuliah}
  Nilai : char {indeks nilai}
>
```

```
type Mahasiswa : record <NIM : integer, {nomor mhs}
  NamaMhs : string, {nama mhs}
  MK : array[1..4] of MataKuliah
>
```

```
LarikMhs : array[1..100] of Mahasiswa
```

www.fppt.info

## Nama

Dalam algoritma nama diberikan pada:

- **Variabel**
  - Tempat penyimpanan data/informasi di memori yang nilainya dapat diubah selama pelaksanaan program
- **Konstanta**
  - Tempat penyimpanan di memori yang nilainya tidak dapat diubah selama pelaksanaan program
- **Tipe bentukan**
  - Tipe data baru yang didefinisikan oleh program dari tipe data yang sudah ada
- **Prosedur**
  - Modul program (sederetan instruksi) yang ditulis terpisah dari badan program utama dan dapat dipanggil berulang dari program utama
- **Fungsi**
  - Prosedur yang mengembalikan suatu nilai dengan tipe data sederhana

www.fppt.info

## Aturan Penulisan Nama

- Harus dimulai dengan huruf alfabet, tidak boleh dimulai dengan angka, spasi, atau karakter khusus lainnya.
- Huruf besar atau huruf kecil tidak dibedakan.
- Karakter penyusun nama hanya boleh: huruf alfabet, angka dan “\_” (underscore)
- Tidak boleh dipisahkan dengan spasi
- Panjang nama tidak terbatas
- Semua nama yang dipakai harus dideklarasikan dulu pada bagian deklarasi

www.fppt.info

## Contoh Penamaan

- **SALAH**
  - 6titik {dimulai dg angka}
  - nilai ujian {dipisahkan spasi}
  - PT-1 {mengandung operator kurang}
  - hari! {mengandung karakter khusus}
- **BENAR**
  - titik6 **atau** titik\_6
  - nilai\_ujian **atau** nilaiUjian
  - PT\_1 **atau** PT1
  - hari

www.fppt.info

## Nilai

- Merupakan besaran dari tipe data yang sudah didefinisikan (tipe dasar maupun tipe bentukan)
- Nilai dapat berupa:
  - Isi variabel atau konstanta
  - Nilai dari hasil perhitungan
  - Nilai yang dihasilkan oleh fungsi
- Nilai yang disimpan di variabel dimanipulasi dengan cara:
  - Mengisikan ke variabel lain yang bertipe sama
  - Dipakai untuk perhitungan
  - Dituliskan ke piranti keluaran

www.fppt.info

## Nilai(2)

Pengisian nilai ke variabel:

- Pengisian nilai secara langsung (assignment)
  - Memasukkan sebuah nilai ke dalam nama variabel langsung di dalam teks algoritma
  - Syaratnya nilai yang di isikan harus bertipe sama dengan tipe peubah
  - Notasi:  $\leftarrow$
  - Contoh:

variabel $\leftarrow$ konstanta	NoMhs $\leftarrow$ 1234
variabel1 $\leftarrow$ variabel2	Nil_prev $\leftarrow$ Nil_cur
variabel $\leftarrow$ ekspresi	Luas $\leftarrow$ $0.5 * p * l$

www.fppt.info

## Nilai(3)

- Pembacaan nilai dari piranti masukan
  - Nilai untuk nama variabel dapat diisi dari piranti masukan, misalnya dari keyboard.
  - Dinamakan dengan operasi pembacaan data
  - Notasi dalam teks algoritma: read
  - Contoh:
    - read(nama1, nama2, ..., namaN)

www.fppt.info

## Ekspresi

- Ekspresi terdiri atas: **operand** dan **operator**
- **Operand** adalah nilai yang dioperasikan dengan **operator** tertentu
- Operand dapat berupa konstanta, nama variabel, nama konstanta, atau hasil suatu fungsi
- Hasil evaluasi dari sebuah ekspresi adalah nilai di dalam domain yang sesuai dengan tipe operand yang dipakai, ada tiga macam: ekspresi aritmetik, ekspresi relasional, ekspresi string.

www.fppt.info

## Ekspresi(2)

- Ekspresi Aritmetika
  - Ekspresi yang baik operand dan hasilnya berupa numerik
  - (ingat: tingkat prioritas operator)
    - i. /, div, mod
    - ii. \*
    - iii. +, -
- Ekspresi relasional
  - Ekspresi dengan operator <, <=, >, >=, =, ≠, not, and, or, dan xor
  - Hasil evaluasi adalah nilai bertipe boolean
  - Ekspresi string

www.fpppt.info

## Ekspresi (3)

- Ekspresi string
  - Ekspresi dengan operator penyambungan/concatenation "+".

www.fpppt.info

## Menuliskan Nilai ke Piranti Keluaran (monitor/printer)

- Dilakukan dengan notasi write
- Contoh:
  - write(nama1, nama2, ..., namaN)

www.fpppt.info

## Contoh

Algoritma Hello\_World  
 {mencetak string Hello World diikuti nama orang.  
 Nama orang diinputkan dari piranti masukan}

### DEKLARASI

const ucapan = 'Hello World'  
 namaUser : string

### DESKRIPSI

read(namaUser)  
write(ucapan + ' ' + namaUser)

www.fpppt.info

## Notasi Algoritmik

- Notasi-notasi deskriptif yang berisi langkah-langkah pencapaian solusi.
- Notasi algoritmik bukan bahasa pemograman.
- Notasi algoritmik yang baik adalah notasi yang mudah dibaca dan mudah **pula** ditranslasikan ke dalam notasi bahasa pemograman.

www.fpppt.info

## Notasi Algoritmik (1)

Notasi 1: menyatakan langkah-langkah algoritma dengan untaian kalimat deskriptif.

### Contoh:

- Bagaimana mencari luas segitiga?  
 Untuk menghitung Luas Segi tiga :
- Masukan Nilai Alas
  - Masukan Nilai Tinggi
  - Hitung Luas =( Alas \* Tinggi ) / 2
  - Cetak Luas

www.fpppt.info

## Notasi Algoritmik (2)

Notasi 2 : menggunakan *pseudo-code*.

- Notasi algoritmik dalam bentuk *pseudo-code* mempunyai korespondensi dengan notasi bahasa pemrograman.
- Proses penterjemahan ke kode program menjadi lebih mudah

www.fppt.info

## Notasi Algoritmik (3)

Algoritmik

write(X, Y)

Pascal

writeln(X, Y);

C

printf("%d %d", X, Y);

Java

System.out.println(X, Y)

PHP

echo \$X. " ". \$Y;

www.fppt.info

## Notasi Algoritmik (4)

- Ada 3 bagian dari notasi *pseudo-code*
  1. Bagian judul (*header*)
  2. Bagian deklarasi (*declaration*)
  3. Bagian algoritma

www.fppt.info

## Notasi Algoritmik (5)

- Contoh notasi algoritmik dengan menggunakan *pseudo-code*.

Program Luas\_Segitiga

Deklarasi:

a, t : integer {alas, tinggi}

l : integer {luas}

Algoritma:

read (a)

read (t)

$l \leftarrow (a * t) / 2$

write(l)

www.fppt.info

## Latihan (1)

1. Diketahui  $P=0$ ,  $Q=5$  dan  $R=15$ . Diberikan Algoritma  $P=Q$ ,  $Q=R$ , maka nilai P, Q, R sekarang?
2. Diketahui algoritma  $P=10$ ,  $P=P-1$  dan  $Q=P$ , berapakah nilai P dan Q?
3. Diketahui 3 variabel peubah P, Q dan R. Agar isi Q ditaruh di P, isi R ditaruh di Q dan isi P ditaruh di R, maka algoritma yang dapat ditulis adalah...
4. Di ketahui 2 peubah  $K=10$  dan  $L=20$ . buatlah algoritma untuk mempertukarkan isi K dan L.

www.fppt.info

## Latihan (2)

1. Buatlah notasi algoritmik dengan menggunakan *pseudo-code* untuk menghitung luas persegi panjang.
2. Diketahui terdapat sebuah variabel peubah X dan Y. Buatlah notasi algoritmik (psedo-code) untuk menentukan apakah kedua variabel tersebut dapat diidentifikasi sebagai segi empat atau persegi panjang.

www.fppt.info

