



*Analisis leksikal*  
( *SCANNER* )

**Teknik Kompilasi**  
**Dosen: Utami Dewi W**



# Tugas Scanner

- ❑ Melakukan pembacaan kode sumber dengan menurut karakter demi karakter
- ❑ Mengenali besaran leksik
- ❑ Mentransformasi menjadi sebuah token dan menentukan jenis tokennya
- ❑ Mengirimkan token
- ❑ Membuang / mengabaikan blank dan komentar dalam program
- ❑ Menangani kesalahan
- ❑ Menangani tabel simbol



# Alat Bantu Scanner

Scanner bekerja berdasarkan mesin FSA yang ada pada bahasa regular. Untuk membantu mengkonstruksi scanner dapat menggunakan **diagram keadaan**.



# Besaran Leksik/Token

Besaran pembangun bahasa / leksis meliputi:

## 1. Identifier (huruf dan angka)

Bisa berupa keyword atau nama.

- ✓ keywords adalah kata kunci yang sudah didefinisikan oleh suatu bahasa seperti BEGIN, END, IF, ELSE dalam Pascal.
- ✓ Nama dideklarasikan sendiri oleh pemakai, seperti nama sebuah variabel.



## Contoh :

VAR

Nomor: INTEGER;

Suhu: REAL;

Maka Nomor dan Suhu adalah Token berupa **nama variabel**, sedangkan VAR, INTEGER, dan REAL adalah **keywords** pada program tersebut.



## 2. Nilai Konstanta

Yaitu konstanta yang terdapat pada program. Berupa konstanta integer, real, boolean, character, string.

Contoh:

$N := R + 5 * 10$

$Kata := kata1 + 'makan'$

$A := 0.333$

$Selesai := TRUE$

Maka 5, 10, 'makan', 0.333, TRUE, termasuk **Token Nilai Konstanta** pada program.



### 3. **Operator dan Delimiter**

Termasuk Operator adalah operator aritmetika , operator logika (< , = , >).

Delimiter berguna sebagai pemisah/pembatas, contohnya: ( ), ; , : , white-space yaitu spasi (yg diabaikan di program), enter, end of file.



# Implementasi Scanner

Scanner biasanya diimplementasikan sebagai sebuah prosedur yang dipanggil oleh Parser.

Prosedur Scan sederhana:

Didefinisikan dulu *Procedure GetChar* untuk mengambil sebuah karakter dari file input.

**Procedure GetChar;**

**begin**

**Read (FileInput, Kar);**

**end;**





## *Implementasi scan sederhana*

Dimana

**FileInput: text, Kar: character**

Pada scanner kita akan maju satu-satu per karakter untuk mendapatkan token. Selanjutnya kita akan buat *Procedure Scan*.

Procedure Scan

Begin


While Kar=' ' do

    GetChar {selama ketemu spasi maju terus}

repeat

## Case kar OF

```
'<': begin
    GetChar;
    Case Kar OF
        '=': begin token:=t_LE; exit end;
        '>': begin token:=t_NE; exit end;
    Else
        Begin
            token:=t_L;
            exit;
            end;
'=:begin token:=t_E exit;end;
'>':begin
    GetChar;
    If Kar='=' then
        begin token:=t_GE;exit;end;
    Else
        begin token:=t_G;exit;end;
End;
```



```
{: begin
  Repeat
  getChar;
{ maju sampai ketemu penutup komentar }
  until Kar= ' }'
  GetChar;
{ lanjutkan maju, tanpa memperoleh token }
  end;
EOF: exit; { akhir file }
Until false { sampai ketemu sebuah token atau akhir file }
End;
```



## **DARI PROCEDURE SCAN DIDAPATKAN HIMPUNAN TOKEN:**

Token={ <, >, =, <=, >=, < }

Atau dibaca sebagai token-token:

**t\_L (less), t\_G(greater), t\_E(equal), t\_LE, t\_GE, t\_NE(not equal)**



## Latihan:

Tentukan Token Leksikal yang terbentuk dari potongan program berikut:

BEGIN

.....

.....

LuasSegitiga := alas\*tinggi\*0.5;

Readln;

END.