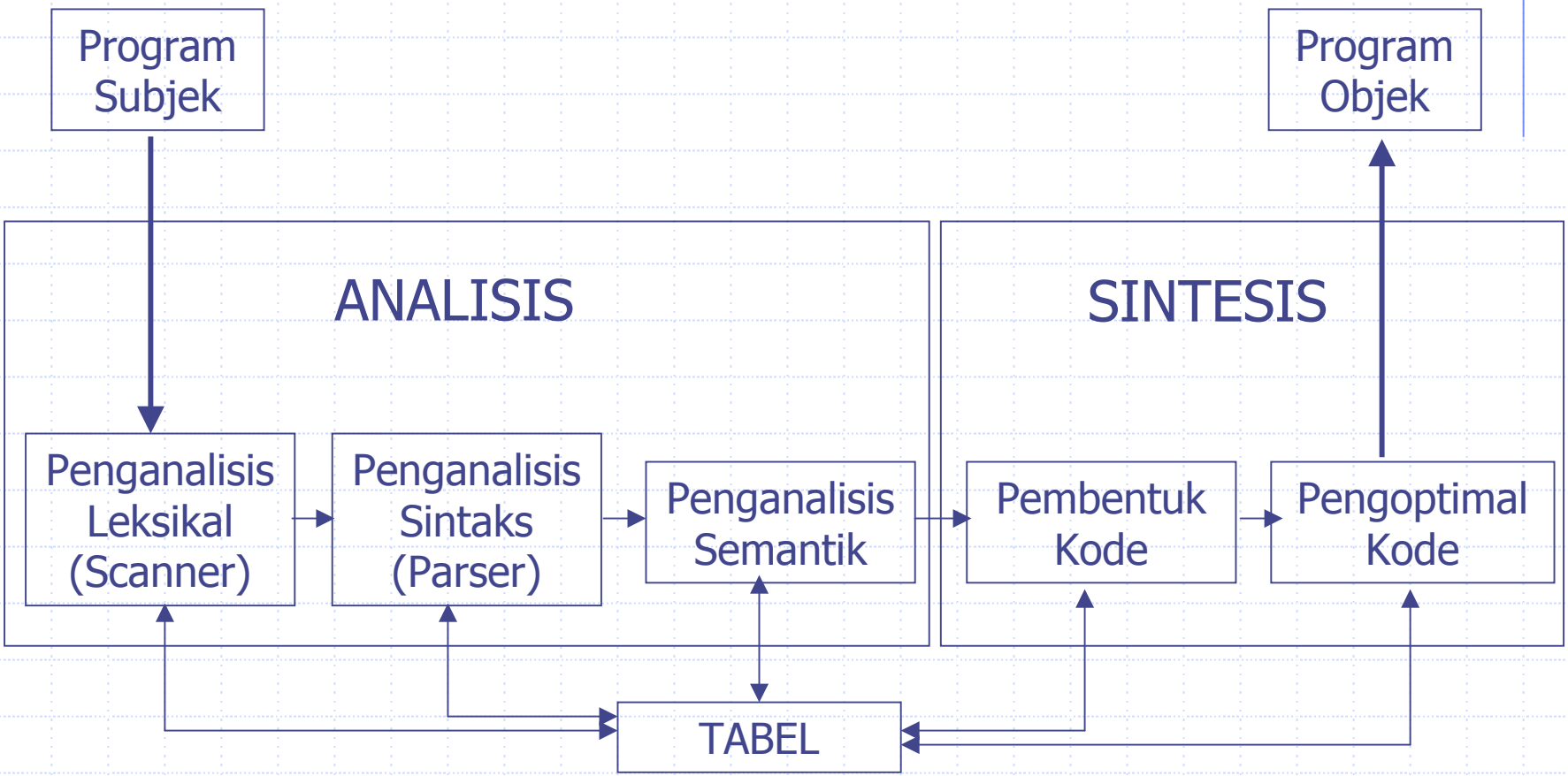
Komponen sebuah Kompilator



Scanning

- Scanning termasuk ke dalam **analisis lexical**, yaitu proses untuk mengidentifikasi satuan terkecil dari Bahasa, yang disebut **Token / Terminal / Daun** (identifier, keyword,label, operator aritmetika dan assignment, operator relasional, tanda baca, dsb)

- Aspek dalam Scanner :
  - Bagaimana bentuk dan penyajian Token. Metode yang digunakan **Grammar Regular** dan **Ekspresi Regular**
  - Pengenalan Token. Metode yang digunakan **Automata Hingga**, dengan penyajian menggunakan Diagram Transisi

Scanning (cont.)

- Scanner berinteraksi dengan Parser, dengan cara :
  - Scanner mengolah Program Source secara terpisah sebagai satu fase, dimana token disimpan dalam sebuah tabel sebelum Parser bekerja
  - Scanner berinteraksi denga Parser, dimana scanner dipanggil oleh parser bila token dalam program source diperlukan

Automata Hingga (AH)

- Automata Hinga (AH) / Finite state Automaton (FA) adalah suatu struktur abstrak yang didefinisikan, terdiri dari :
  1. Himpunan Hingga A berisi simbol Input
  2. Himpunan Hingga S berisi State (internal state)
  3. Himpunan Hingga Z berisi simbol output
  4. Sebuah fungsi f: S x Z $\rightarrow$ S, disebut fungsi next state
  5. Sebuah fungsi g: S x A $\rightarrow$Z, disebut fungsi output
- AH berhubungan dengan Regular Grammar
- Jenis AH :
  1. AH Determinstik (AHD)
  2. AH Non Deterministik (AHN)
  3. AHN dengan transisi untai hampa

Automata Hingga Deterministik (AHD)

- Automata Hinga Deterministik (AHD) didefinisikan dengan 5 tupel
  1. Himpunan Hingga internal state (S)
  2. Himpunan Hingga simbol input (V)
  3. Sebuah fungsi f: S x V $\rightarrow$ S ; merupakan fungsi next state
  4. State awal ($q_0 \in$ S)
  5. Himpunan hingga state penerima $\subset$ S

- AHD sering digambarkan dengan cara :
  - Table Transisi State
  - Transisi Digraph

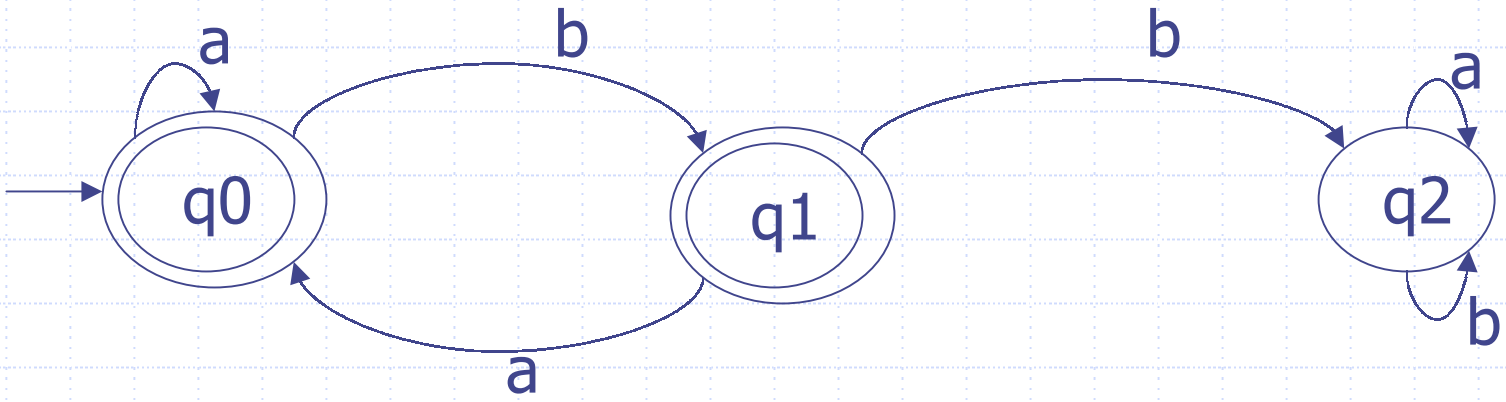
Automata Hingga Deterministik (AHD) (cont.)

Contoh : Diketahui AHD dengan 2 simbol input dan 3 state,

1. V = { a,b}
2. S = { q0, q1, q2)
3. T = {q0, q1} state penerima
4. q0 sebagai state awal
5. Fungsi next state didefinisikan f:(S,V)

f \ input	a	b
q0	q0	q1
q1	q0	q2
q2	q2	q2

Automata Hingga Deterministik (AHD) (cont.)



Periksalah string berikut :

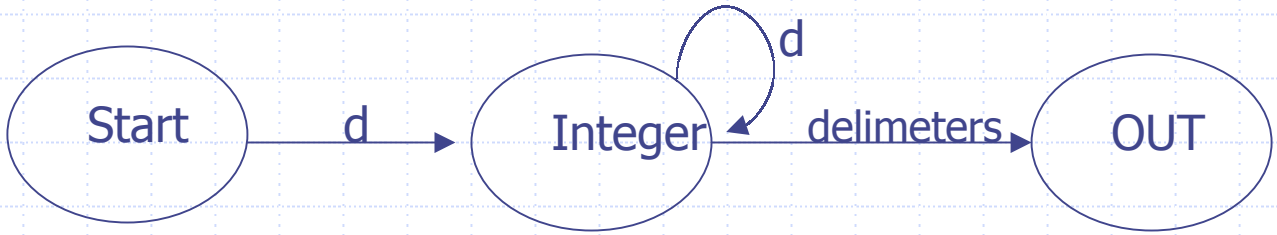
1. aba
   $q0 \rightarrow q0 \rightarrow q1 \rightarrow q0$ (diterima)

2. aabbaa
   $q0 \rightarrow q0 \rightarrow q0 \rightarrow q1 \rightarrow q2 \rightarrow q2 \rightarrow q2$ (ditolak, state penerima yang ditentukan adalah q0 dan q1, bukan q2)

Automata Hingga Deterministik (AHD) (cont.)

Contoh : Identifikasi Integer String

Grammar : <integer> ::= <digit> | <integer><digit>



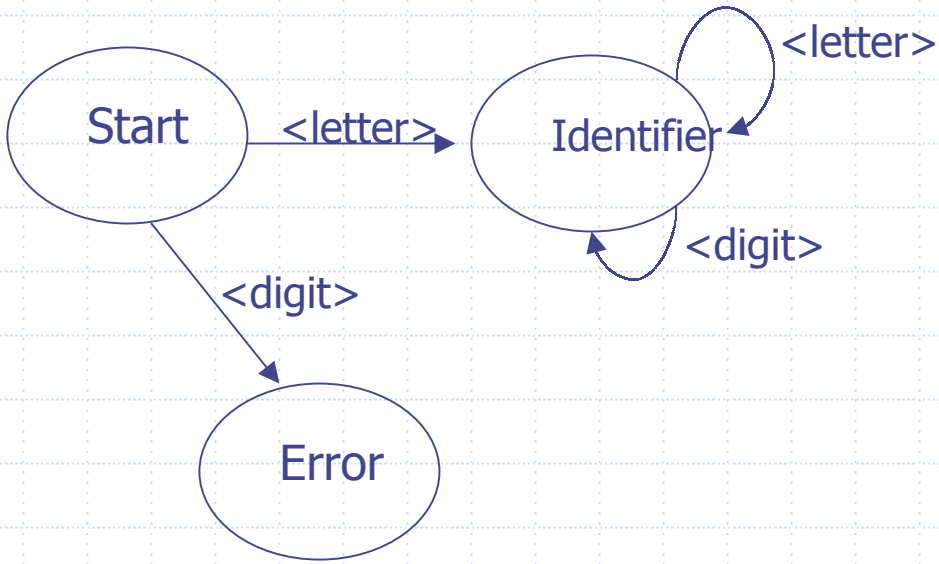
input / state	d	≠ d
START	INTEGER	-
INTEGER	INTEGER	OUT
OUT		

(Accept)

Automata Hingga Deterministik (AHD) (cont.)

Contoh : Identifikasi Identifier String

<identifier> ::= <letter>|<identifier><letter>|<identifier><digit>



Parsing

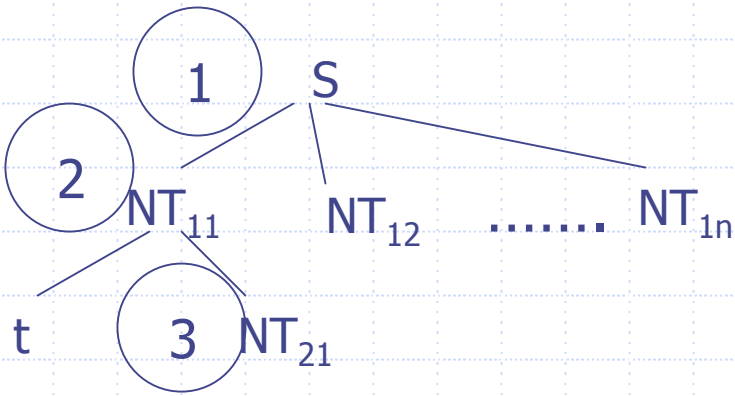
- Parsing adalah konsturksi atau pembentukan **Pohon Sintaks** untuk suatu kalimat (ekspresi)
- Bila terdapat lebih dari satu pohon sintaks untuk sebuah grammar maka dikatakan grammar tersebut **Ambiguous.**
- Dua cara melakukan validitas sintaks dengan parsing :
  - **TOP DOWN Parsing** : melakukan derivasi string dari NT
  - **BOTTOM UP Parsing** : melakukan reduksi simbol ke NT

Parsing Top Down

- Jika α adalah input string, maka derivasi dari Top Down Parse dapat ditunjukkan sebagai berikut :

  $S \Rightarrow \ldots \Rightarrow \ldots \Rightarrow \ldots \Rightarrow \alpha$

- Parse Tree untuk Top Down Parsing selalu dimulai dari sebelah kiri

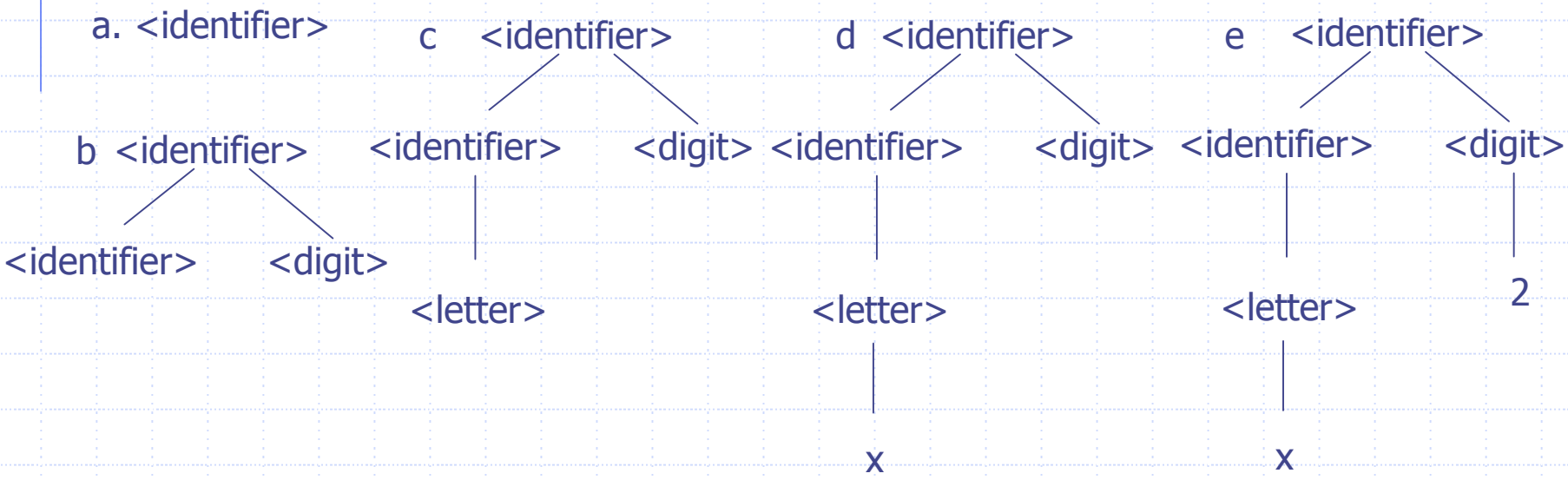


Parsing Top Down (cont.)

Contoh : Parsing Top Down untuk identifier x2

Derivasinya :

<identifier> ⇒ <identifier><digit> ⇒<letter><digit> ⇒ x <digit> ⇒ x2



Parsing Top Down (cont.)

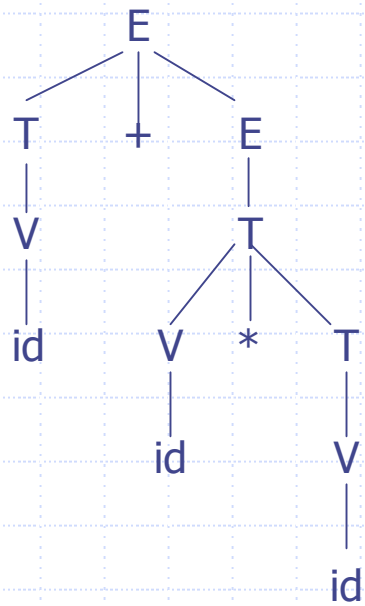
Contoh : ekspresi a + b * c

grammar : E ::= T + E | T

T ::= V * E | V

V ::= <id>

Prediction	Prediction Sentential Form
E⇒T+E	T+E
T⇒V	V+E
V⇒<id>	<id>+E
E⇒T	<id>+T
T⇒V*T	<id>+V*T
V ⇒<id>	<id>+<id>*T
T ⇒V	<id>+<id>*V
V ⇒<id>	<id>+<id>*<id>



Parsing Bottom Up

- Parsing Bottom Up membangun pohon sintaks melalui urutan simbol yang direduksi, atau dimulai dengan sebuah string hingga mencapai simbol start Grammar
- Contoh : diketahui identifier x2, dengan parsing bottom up menjadi :