CS 444 - Compiler Construction **Winter 2020**

Lecture 7: January 27nd, 2020

Lecturer: Ondřej Lhoták *Notes By: Harsh Mistry*

7.1 Analysis Continued

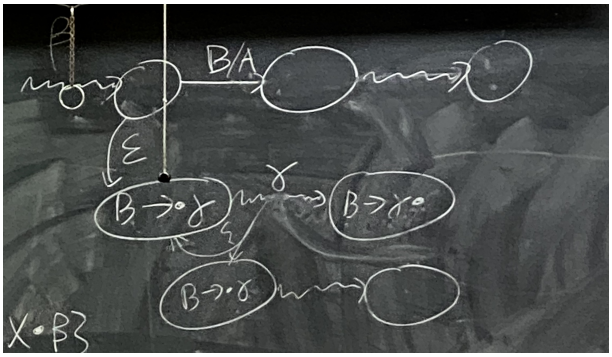
7.1.1 Parsing Continued

7.1.1.1 LR(0) Parsing Algorithm

- for each a in $\ell\$$ (ℓ is input)
  - While ($Reduce(\text{stack}) = \{A \to \gamma\}$)
    - Pop $|\gamma|$ times
    - Push A
  - if ($Reject(stack + a)$) then throw ERROR
  - push a
- $Reject(\alpha) = \alpha$ is not a viable prefix
- $Reduce(\alpha) = \{A \to \gamma | \exists P, \alpha = \beta\gamma$ and βA is a viable prefix $\}$

7.1.1.2 LR(0) NFA

- $\sigma = T \cup N$
- $Q = \{A \to \alpha \cdot \beta | A \to \alpha\beta \in R\}$
- $q_0 = S^1 \to \cdot S\$$
- $A = Q$
- $\delta(A \to \alpha \cdot X\beta, X) = \{A \to \alpha X \cdot B\}$
- $\delta(A \to \alpha \cdot \beta, \epsilon) = \{B \to \cdot\gamma | B \to \gamma \in R\}$



7.1.1.3 Convert NFA to DFA

- If a DFA state contains $A \to \gamma\cdot$ and $B \to \alpha \cdot XB$, shift-reduce conflict
- If a DFA state contains $A \to \gamma\cdot$ and $B \to \delta\cdot$, reduce-reduce conflict

Definition 7.1 *A grammar is LR(0) if LR(0) DFA has no conflicts*