

FIRST, FOLLOW e FIRST+

- Como o analisador LL(1) sabe qual regra aplicar, dado o lookahead? " ϵ " = ϵ
- Examinando os *conjuntos de lookahead* (FIRST+) de cada regra *alternativa*

$$\text{FIRST+}(A \rightarrow w) = \begin{cases} \text{FIRST}(w) \cup \text{FOLLOW}(A) - \{ '\epsilon' \}, & \text{se } '\epsilon' \text{ em FIRST}(w) \\ \text{FIRST}(w), & \text{caso contrário} \end{cases}$$

- E quem são os conjuntos FIRST e FOLLOW? Revisão de linguagens formais!

$$\begin{aligned} \text{FIRST}(w) &= \{ x \text{ é terminal} \mid w \xrightarrow{*} xv, \text{ para alguma string } v \} \cup \\ &\quad \{ '\epsilon' \mid w \xrightarrow{*} '\epsilon' \} \\ \text{FOLLOW}(A) &= \{ x \text{ é terminal} \mid S \xrightarrow{*} wAxv \text{ para algum } w \text{ e } v \} \cup \\ &\quad \{ \text{EOF} \mid S \xrightarrow{*} wA \text{ para algum } w \} \end{aligned}$$

$$\begin{aligned} \text{FIRST}(xw) &\subseteq \{x\} \quad \text{se } x \in T \\ \text{FIRST+}(A \rightarrow xw) &\subseteq \{x\} \quad \text{se } x \in T \end{aligned}$$

$$X \rightarrow w_i$$

FIRST

$$\text{FIRST}(xw) = \{x\} \quad x \in T, w \in (T \cup V)^*$$

$$\text{FIRST}(\epsilon) = \{\epsilon\}$$

$$\text{FIRST}(X) = \text{FIRST}(w_1) \cup \dots \cup \text{FIRST}(w_n), \quad x \in V$$

$$\text{FIRST}(Xw) = \begin{cases} \text{FIRST}(X) & \text{if } \epsilon \notin \text{FIRST}(X) \\ (\text{FIRST}(X) - \{\epsilon\}) \cup \text{FIRST}(w) & \text{if } \epsilon \in \text{FIRST}(X) \end{cases} \quad (X, w_n) \in P$$

$$\underline{Xw} \xRightarrow{*} w \xRightarrow{*} xw'$$

FOLLOW

$$\text{FOLLOW}(A) = \bigcup_{(B, w_1 \triangleright w_2) \in P} \begin{cases} \text{FIRST}(w_2), & \text{if } \epsilon \notin \text{FIRST}(w_2) \\ (\text{FIRST}(w_2) - \{\epsilon\}) \cup \text{FOLLOW}(B), & \text{if } \epsilon \in \text{FIRST}(w_2) \end{cases}$$

A condição LL(1)

- Uma gramática é LL(1) se os conjuntos FIRST+ das regras de cada não-terminal são *disjuntos*
- Por que isso faz a análise LL(1) funcionar? Vejamos a consequência de uma escolha LL(1):

...At... \rightarrow ...t... ou
...A... \rightarrow ...tw...

- No primeiro caso isso quer dizer que t está no FOLLOW(A)
- No segundo caso, t está no FIRST do lado direito da regra de A que foi usada
- A derivação é mais à esquerda, então o primeiro ... é um prefixo de terminais, logo t é o lookahead!