

GLC:  $(V, \Sigma, R, S)$  definem LLC

V Conjunto finito de Variáveis

$\Sigma$  Conjunto finito de terminais

R Conjunto finito de regras

S Variável inicial

AFD  $\rightarrow$  GLC  $\delta(q_i, a) = q_j \begin{cases} R_i \rightarrow a R_j \\ R_i \rightarrow \epsilon, q_i \in F \end{cases}$

$\rightarrow \textcircled{A} \xrightarrow{a} \textcircled{B} \xrightarrow{b} \textcircled{C} \quad \begin{matrix} A \rightarrow aB \\ B \rightarrow bC \\ C \rightarrow \epsilon \end{matrix}$

União de gramáticas  $\{0^n 1^n \mid n > 0\} \cup \{1^n 0^n \mid n > 0\}$

$S \rightarrow S_1 | S_2$

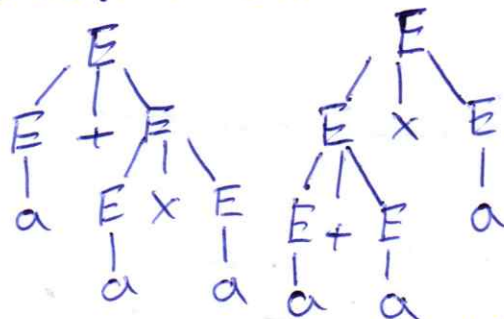
$S_1 \rightarrow 0 S_1 1$

$S_2 \rightarrow 1 S_2 0$

GLC ambigua possui mais de uma árvore para mesma palavra

$E \rightarrow E + E \mid E \times E \mid a$

$w = a + a \times a$



**Teorema** Qualquer LLC é gerada por GLC na forma normal de Chomsky

Prova: Converter gramática por FNC

1. Criar nova variável inicial com regra para a antiga
2. Remover toda regra  $A \rightarrow \epsilon$ , A não é variável inicial, para cada ocorrência de A no lado direito criar uma regra trocando A por  $\epsilon$
3. Remover toda regra  $A \rightarrow B$ , trocando B pelas regras dele
4. substituir toda regra com mais de 2 elementos por uma com apenas dois que leva ao resto da antiga regra

$S \rightarrow ASA | aB$   
 $A \rightarrow B | S$   
 $B \rightarrow b | \epsilon$

1.  $S_0 \rightarrow S$   
 $S \rightarrow ASA | aB$   
 $A \rightarrow B | S$   
 $B \rightarrow b | \epsilon$

2.  $S_0 \rightarrow S$   
 $S \rightarrow ASA | aB | a$   
 $A \rightarrow B | S | \epsilon$   
 $B \rightarrow b | \epsilon$