

$B \rightarrow \text{true}$

$B.\text{code} = \text{gen}(\text{GOTO } B.\text{true})$

$B \rightarrow \text{false}$

$B.\text{code} = \text{gen}(\text{GOTO } B.\text{false})$

$B \rightarrow \text{not } B_1$

$B_1.\text{true} = B.\text{false}$

$B_1.\text{false} = B.\text{true}$

$B.\text{code} = B_1.\text{code}$

$B \rightarrow B_1 \parallel B_2$

$B_1.\text{true} = B.\text{true}$

$B_1.\text{false} = \text{newlabel}()$

$B_2.\text{true} = B.\text{true}$

$B_2.\text{false} = B.\text{false}$

$B.\text{code} = B_1.\text{code} \triangleright$

$\text{label}(B_1.\text{false}) \triangleright$

$B_2.\text{code}$

$B_1.\text{false}:$ $\begin{cases} B_1.\text{code} \\ B_2.\text{code} \end{cases}$

$B \rightarrow B_1 \&\& B_2$

$B_1.\text{true} = \text{newlabel}()$

$B_1.\text{false} = B.\text{false}$

$B_2.\text{true} = B.\text{true}$

$B_2.\text{false} = B.\text{false}$

$B.\text{code} = B_1.\text{code} \triangleright$

$\text{label}(B_1.\text{true}) \triangleright$

$B_2.\text{code}$

$B_1.\text{true}:$ $\begin{cases} B_1.\text{code} \\ B_2.\text{code} \end{cases}$

poli, zio delle grammatiche non libero

- si trovano gli elementi nullabili
- si aggiungono alle regole le loro versioni senza elementi nullabili, in ogni combinazione
- si eliminano le transizioni in ϵ e i non-terminali inutili

Pumping lemma linguaggi liberi

si concentra nella ricerca nell'albero di derivazione di sottoalberi annullati ripetuti

$s \rightarrow a s b \Rightarrow$

uv^ixy



