

Método sistemático

As observações que nós acabamos de fazer já são suficientes para obter um método que converte gramáticas livres de contexto em autômatos de pilha de maneira sistemática.

E o mais interessante é que esse autômato sempre tem 3 estados apenas.

Quer dizer, a ideia é fazer a divisão de tarefas de maneira ligeiramente diferente

- nós vamos ter um estado q_f que é responsável por ler símbolos na palavra de entrada (que estão indicados na pilha)
- e um outro estado q_R que trata os lembretes que forma deixados na pilha, aplicando as regras da gramática

O terceiro estado apenas dá início ao processo, colocando o símbolo da variável inicial S na pilha.

Vejamos um exemplo concreto.

Exemplo 5: a gramática de BAL e o seu autômato de pilha

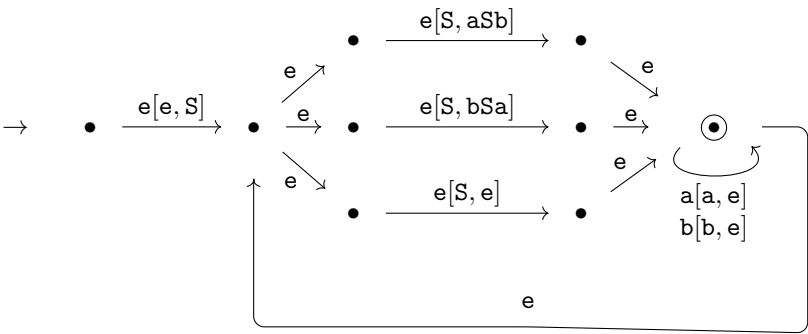
Abaixo nós temos a gramática de BAL

$S \rightarrow aSb \mid bSa \mid SS \mid e$

Como indicado acima, a ideia é ter

- um estado q_R com uma transição para cada regra da gramática
- um estado q_f que lê símbolos da palavra de entrada e os compara com os símbolos da pilha
- e o estado inicial do autômato

Abaixo nós temos o diagrama do autômato de pilha que reconhece a linguagem BAL



◇

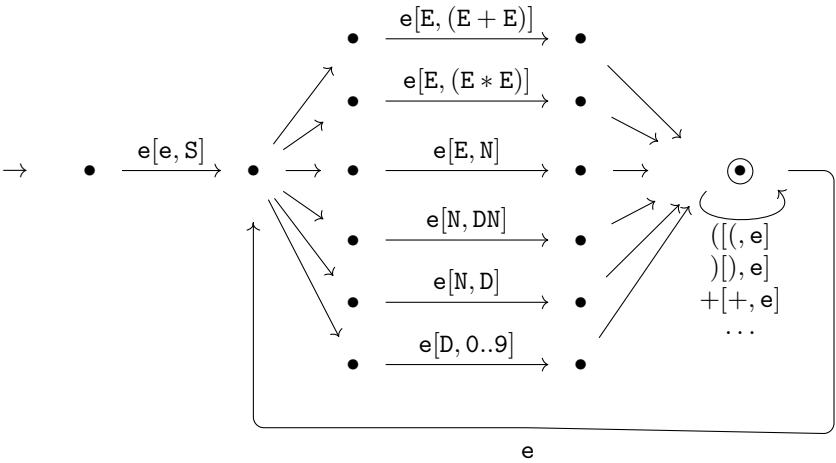
Abaixo nós temos mais um exemplo, que aplica o método sistemático sobre a gramática de EXPR.

Exemplo 6: a gramática de EXPR e o seu autômato de pilha

Abaixo nós temos a gramática de EXPR

$E \rightarrow (E + E) \mid (E * E) \mid N$
 $N \rightarrow D N \mid D$
 $D \rightarrow 0 \mid \dots \mid 9$

E abaixo nós temos o autômato de pilha que reconhece EXPR



◇