


### 3 Análise Léxica #léxica

consiste na conversão de sequência de caracteres de entrada numa sequência de elementos léxicos (tokens), a serem processados pelo Parser

Faz ainda outras operações como excluir espaços e comentários do parser e a correlação entre erros e o código fonte (linha)

implementado com recurso a gramáticas do tipo-3, ou seja, regulares e a autômatos finitos

gramáticas regulares  
se existe um autômato finito que a reconhece  
autômato finito máquina de estados com estado inicial e 1 ou + de aceitação  
transições feitas tendo em conta estado atual e entrada  


linguagem regular gerada por gramáticas regulares

sobre um alfabeto A, define-se indutivamente da seguinte forma

1.  $\emptyset$  é uma LR
2.  $a$   $a \in A$ ,  $\{a\}$  é uma LR
3.  $L_1 \cup L_2$  é uma LR união
4.  $L_1 \cdot L_2$  é uma LR concatenação
5.  $(L_1)^*$  é uma LR fecho de Kleene
6. Nada mais é LR

como operações sobre LR geram uma LR, diz-se que é fechada sobre as suas operações

exemplos qualquer linguagem finita é uma linguagem regular

qualquer linguagem reconhecida num autômato finito é LR

gramática quadruplo  $G = (T, N, S, P)$  que define linguagens

- T alfabeto terminal (conjunto finito não vazio)
- N conjunto finito não vazio de símbolos não terminais
- S símbolo inicial ( $S \in N$ )
- P conjunto finito de regras  $\alpha \rightarrow \beta$  (a tem pelo menos um simb. terminal)



gramática se para qualquer produção  $(\alpha \rightarrow \beta) \in P$   
regular

•  $\alpha \in N$

•  $\beta \in T^* \underset{\text{à direita}}{U T^* N} \mid T^* \underset{\text{à esquerda}}{U N T^*}$

operações sobre GR dizem-se **fechadas** pois geram GR

consideremos  $G_1 = (T_1, N_1, S_1, P_1)$  e  $G_2 = (T_2, N_2, S_2, P_2)$  com  $N_1 \cap N_2 = \emptyset$

união  $G_1 \cup G_2$  gera  $G = (T, N, S, P)$

$$T = T_1 \cup T_2$$

$$N = N_1 \cup N_2 \cup \{S\}, S \notin (N_1 \cup N_2)$$

$$S = S$$

$$P = \{S \rightarrow S_1, S \rightarrow S_2\} \cup P_1 \cup P_2$$