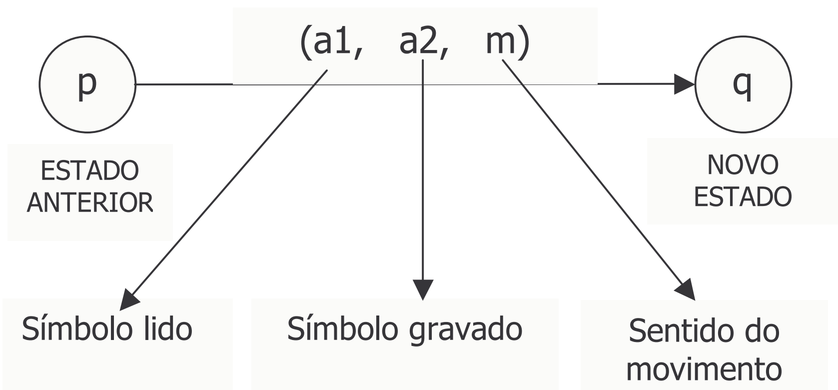
Linguagens Formais, Autômatos e Computabilidade

Resumo para o ENADE 2005

Linguagens Regulares			
Linguagens Formais na Classificação de Chomski	*Linguagens*: 0. LER 1. LSC 2. LLC 3. LR	*Formalismos*: 0. GI/MT/A2P 1. GSC/MT/A2P 2. GLC/A1P/AFN/AFε 3. GRs/A0P/AFD/ER	*Conceitos Básicos*: Alfabeto, Palavra, Palavra Vazia, Subpalavra, Prefixo, Sufixo, Concatenação, Concatenação Sucessiva, ...
Tipos de Formalismos	*Axiomáticos ou Geradores*: Gramáticas	*Reconhecedores ou Operacionais*: Autômatos Finitos	*Denotacionais*: Expressões Regulares
Autômato Finito Determinístico (AFD)	*Definição*: $M=(\Sigma,Q,\delta,q_0,F)$	*Função de Transição*: δ: $Q x \Sigma \to Q$, parcial	Leva de um par estado-símbolo em um *novo estado*.
Autômato Finito Não-Determinístico (AFN)	*Definição*: $M=(\Sigma,Q,\delta,q_0,F)$	*Função de Transição*: δ: $Q x \Sigma \to 2^Q$, parcial	Leva de um par estado-símbolo em *um conjunto de estados possíveis*.
Autômato Finito com Movimento Vazio (AFε)	*Definição*: $M=(\Sigma,Q,\delta,q_0,F)$	*Função de Transição*: δ: $Qx(\Sigma \cup \{\varepsilon\}) \to 2^Q$, parcial	Como no AFN e ainda permite a execução de *movimentos vazios*.
Expressões Regulares (ER)	*Operações*: 1. união (+) 2. concatenação 3. concatenação sucessiva (*)	*Exemplos de ERs*: aa ba* (a+b)* (a+b)*aa(a+b)* a*ba*ba* (a+b)*(aa+bb) (a+ε)(b+ba)*	*Teorema*: Construção de AFε a partir de ERs.
Gramáticas Regulares (GR)	*Definição*: G=(V,T,P,S)	*G. Lineares*: 1. GLD 2. GLE 3. GLUD 4. GLUE	*Produções*: $A \to wB$ ou $A \to w$ $A \to Bw$ ou $A \to w$ GLD com $\|w\| \leq 1$ GLE com $\|w\| \leq 1$
Máquina de Mealy	*Definição*: $M=(\Sigma,Q,\delta,q_0,F,\Delta)$	*Função de Transição*: δ: $Q x \Sigma \to Q x \Delta^*$	É um AFD com saídas associadas às *transições*.
Máquina de Moore	*Definição*: $M=(\Sigma,Q,\delta,q_0,F,\Delta,\delta_S)$	*Função de Transição*: δ: $Q x \Sigma \to Q$ *Função de Saída*: δ_S: $Q \to \Delta^*$	É um AFD com saídas associadas aos *estados.*

Linguagens Livres de Contexto			
Gramáticas Livres de Contexto (GLC)	*Definição*: Uma GLC G é uma gramática G = (V, T, P, S), onde qualquer regra de produção em P é da forma A → α, onde A é uma variável de V e α é palavra de (V $\cup$T)*. Única restrição: uma e somente uma variável no lado esquerdo da produção.	*Exemplo*: G=({E},{+,*,[,],x},P,E), onde: P={ E→E+E , E→E*E , E→[E] , E→x } Gera expressões do tipo: x + [x * x] [x * [x + x]] Uma linguagem é dita "*livre de contexto*" se e somente se for gerada por uma GLC.	A expressão "livre de contexto" significa que para tais linguagens, cuja produção é da forma A → α, em uma derivação a variável A deriva α sem depender (livre) de qualquer análise dos símbolos que antecedem ou seguem A (contexto).
Árvore de Derivação	a) A raiz é o símbolo inicial da GLC; Os vértices interiores obrigatoriamente são variáveis. Se A é um vértice interior e X1, X2, ..., Xn são os filhos de A, então A → X1X2...Xn é uma produção da gramática e os vértices X1, X2, ..., Xn estão ordenados da esquerda para a direita. Um vértice folha é um símbolo terminal ou um símbolo vazio. Neste caso o vazio é o único filho de seu pai (A → ε)	*Exemplo*: *Árvore de derivação* da expressão x + [x*x] na gramática exemplificada acima. As folhas, da esquerda para a direita, geram a expressão: E / | \\ E + E | / | \\ x [E] / | \\ E * E | | x x *Cadeia de derivação*: E➔E+E➔E+[E]➔ E+[E*E]➔x+[E*E]➔ x+[x*E]➔x+[x*x].	*Notas*: Uma GLC é dita uma gramática *ambígua* se existe uma palavra que possui duas ou mais àrvores de derivação. (Ex: x+x*[x+x]). *Forma Normal de Chomsky*: Todas as produções são do tipo A→BC ou A→a. *Forma Normal de Greibach*: Todas as produções são da forma A→aα, onde α é uma palavra de variáveis.
Autômato com 1 Pilha (A1P)	*Definição*: M=(Σ,Q,δ,q_0,F,V)	*Função de Transição*: δ: Qx($\Sigma\cup${ε,?})x(V$\cup${ε,?}) → 2^{QxV*} (O A1P é um AFN).	*Nota*: O A1P reconhece qualquer LLC, com um único estado (ou 3 estados, conforme a definição). Isto é, a pilha é suficiente como memória auxiliar, não sendo necessário usar estados para memorizar as entradas.

Máquina de Turing			
Máquina de Turing (MT)	*Definição*: $M=(\Sigma,Q,\delta,q_0,F,V,\beta)$	*Função de Transição*: $\delta:Qx(\Sigma\cup V\cup\{\beta\})\rightarrow Qx(\Sigma\cup V\cup\{\beta\})x\{E,D\}$, parcial	*Reconhece*: LSC e LER *Gramáticas*: GSC e GI
Componentes da MT	*a) Fita*: É limitada a esquerda e usada simultaneamente como dispositivo de entrada, saída e memória de trabalho.	*b) Unidade de Controle*: Reflete o estado corrente da máquina. Possui uma unidade de leitura e gravação (cabeça da fita) que acessa uma célula da fita de cada vez, movendo-se para a esquerda e para a direita.	*c) Programa ou Função de Transição*: Comanda as leituras e gravações, o sentido do movimento da cabeça e define o estado da máquina.
Hipótese de Church	"A Máquina de Turing é um modelo canônico para a computação universal"	"Se x é uma função computável, x pode ser computada pela Máquina de Turing".	Alonzo Church, 1936
Transição na MT **$\delta(p,a1)=(q,a2,m)$**			
Observações Finais	Tanto a MT quanto o AnP podem entrar em LOOP.	A MT tem poder computacional equivalente a um A2P (Autômato com DUAS pilhas).	A MT é em geral considerada um AFD. Sabe-se que o não-determinismo não aumenta o seu poder computacional.