Linguagens livres de contexto

Teoria da computação

Prof. Allan Rodrigo Leite

Linguagens livres de contexto

- Abordagem mais ampla que as linguagens regulares
  - Análise sintática
  - Balanceamento de parênteses
  - Blocos de programas

- Relação entre linguagens regulares e livres de contexto
  - Pela hierarquia de Chomsky, a classe das gramáticas livres de contextos contém a classe das linguagens regulares
  - Uma linguagem regular pode ser representada por uma gramática regular, mas o contrário não é válido

Linguagens livres de contexto

- Exemplo
  - Como construir uma gramática para expressar cadeias que são palíndromos, considerando o alfabeto $\Sigma = \{0,1\}$?
  - Palíndromos
    - 0110
    - 11011
    - λ
  - Não palíndromos
    - 011
    - 0101
    - 10

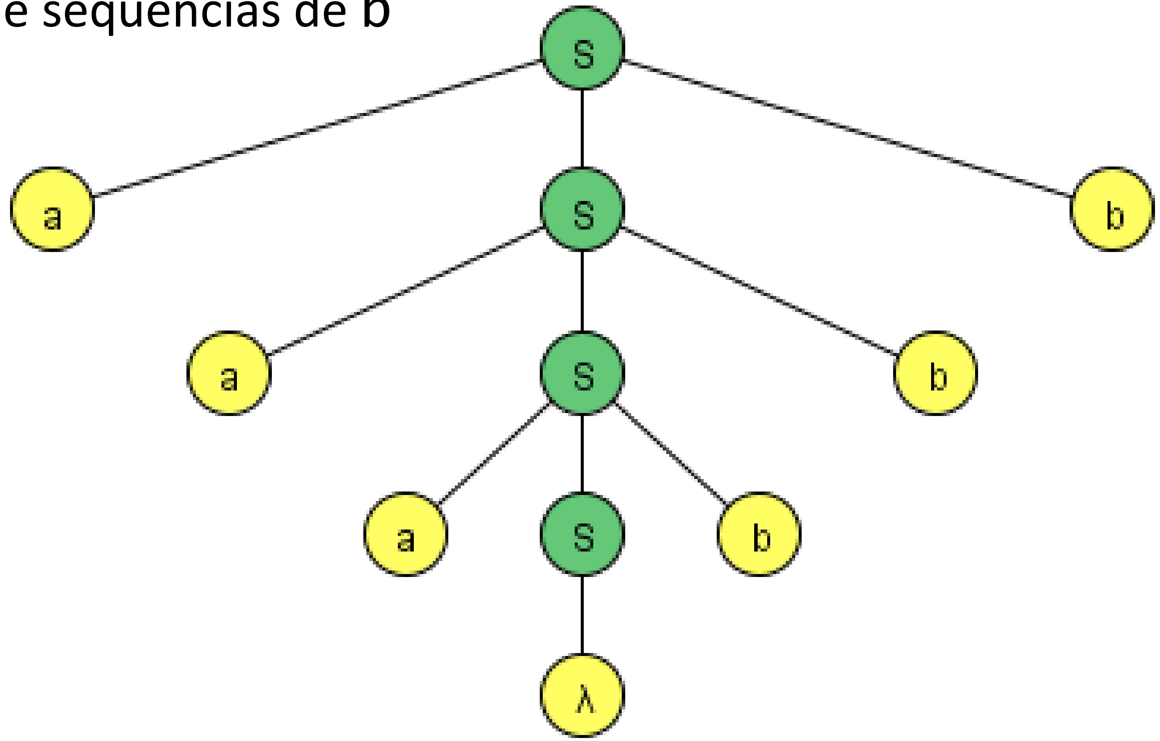
Gramática livre de contexto

- Formalmente $G = < V, T, S, P >$
  - V: conjunto finito de símbolos variáveis ou não-terminais
  - T: conjunto finito de símbolos terminais disjunto de V
  - P: relação finita de produções conforme $(V \cup T)^+ \to (V \cup T)^*$
  - S: elemento distinguido de V que representa o símbolo ou variável inicial

- Representação de uma regra de produção
  - $A \to \alpha$, onde $A \in V$ e $\alpha \in (V \cup T)^*$
  - Isto é, não apresenta as restrições de uma gramática linear (regular)

Gramática livre de contexto

- Exemplo
  - $L = \{ a^n b^n \mid n \geq 0 \}$ é uma linguagem com G para expressar:
    - Cadeias que iniciem com um ou mais a e terminem com um ou mais b
    - Sequencias de a com o mesmo tamanho de sequencias de b
  - $G = < \{S\}, \{a,b\}, S, P >$
    - G é uma gramática livre de contexto
    - $P = \{ S \rightarrow aSb, S \rightarrow \lambda \}$



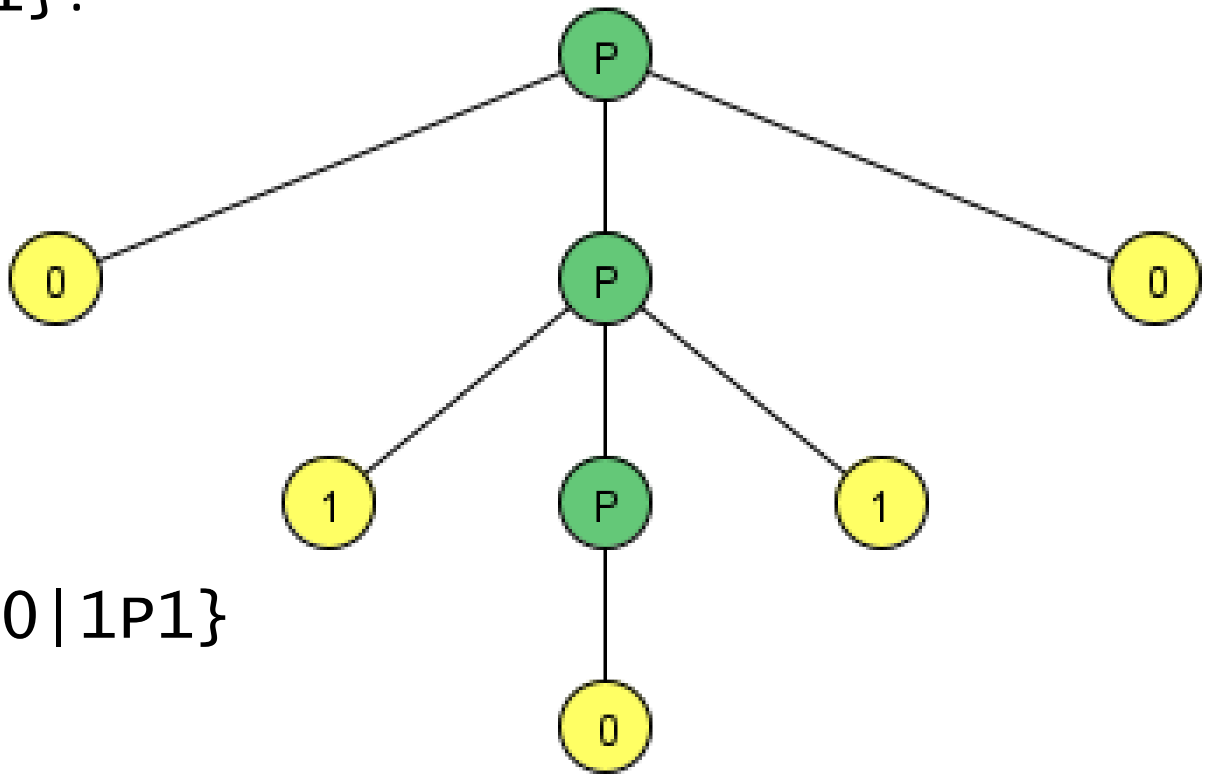
Gramática livre de contexto

- Exemplo
  - Como construir uma gramática para expressar cadeias que são palíndromos, considerando o alfabeto $\Sigma = \{0,1\}$?

- $G = < V,T,P,S >$
  - $V: \{S\}$
  - $T: \{0,1\}$
  - $P: \{P \rightarrow \lambda, P \rightarrow 0|1, P \rightarrow 0P0|1P1\}$

Gramática livre de contexto

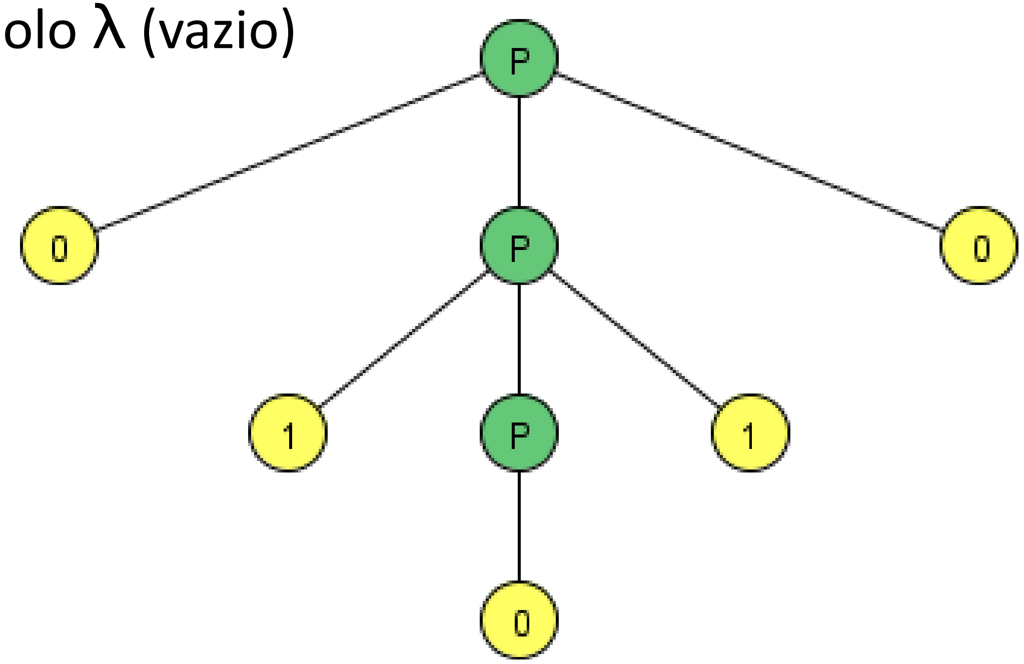
- Exemplo
  - Como construir uma gramática para expressar cadeias que são palíndromos, considerando o alfabeto $\Sigma = \{0,1\}$?
  - Exemplo: 01010



- $G = < V,T,P,S >$
  - V: $\{S\}$
  - T: $\{0,1\}$
  - P: $\{P \rightarrow \lambda, P \rightarrow 0|1, P \rightarrow 0P0|1P1\}$

Árvore de derivação

- Representação do produto da aplicação sucessiva de regras de produção
  - A raiz da árvore é o símbolo inicial da gramática (S)
  - Os vértices interiores são variáveis
  - Se A é um vértice interior e $X_1, \ldots, X_N$ são filhos de A, então $A \rightarrow X_1 | \ldots | X_N$ é uma produção da gramática
  - Os vértices folha são símbolos terminais ou símbolo λ (vazio)



- Tipos de derivação
  - Derivação a esquerda: em cada passo, a variável $A \in V$ mais a esquerda é substituída
  - Derivação a direita: em cada passo, a variável $A \in V$ mais a direita é substituída

Árvore de derivação

- Dada a linguagem $L = \{a^n b^m \mid n \neq m\}$
  - G = < {S,T,A,B},{a,b},S,P >
  - P = { P → AT|TB, T → aTb|λ, A → aA|a,B → bB|b }

- Exemplo: aaab
  - Derivação a esquerda
  - S → **A**T → a**A**T → aa**T** → aaa**T**b → aaab

  - Derivação a direita
  - S → A**T** → Aa**T**b → **A**ab → a**A**ab → aaab

Árvore de derivação

- Gramática ambígua
  - Resulta em diferentes derivações para uma mesma cadeia

- Exemplo: $a + a$
  - $G_1 = < \{S\},\{a,+\},S,P >$
  - $P = \{S \rightarrow a | S + S\}$
  - G_1 é ambígua pois:
    - $S \rightarrow S + S \rightarrow S + a \rightarrow a + a$ (derivação à esquerda)
    - $S \rightarrow S + S \rightarrow a + S \rightarrow a + a$ (derivação à direita)

Árvore de derivação

- Gramática ambígua
  - Resulta em diferentes derivações para uma mesma cadeia

- Exemplo: `a + a`
  - G_2 `= < {S},{a,+},S,P >`
  - `P = {S → a|a + a|S + S}`
  - G_2 é ambígua pois:
    - `S → a + a`
    - `S → S + S → S + a → a + a` (derivação à esquerda)
    - `S → S + S → a + S → a + a` (derivação à direita)

Árvore de derivação

- Gramática ambígua
  - Resulta em diferentes derivações para uma mesma cadeia
- Exemplo: `a + a`
  - G_3 `= < {S},{a,+},S,P >`
  - `P = {S → a|a + S}`
  - G_3 não é ambígua pois:
    - `S → S + S → a + S → a + a` (derivação à direita)

Árvore de derivação

- Gramática ambígua
  - Resulta em diferentes derivações para uma mesma cadeia

- Exemplo: `a + a`
  - No entanto, a linguagem $L = L(G_1) = L(G_2) = L(G_3)$ não é ambígua, já que existe uma gramática G_3 que não é ambígua

Gramática livre de contexto

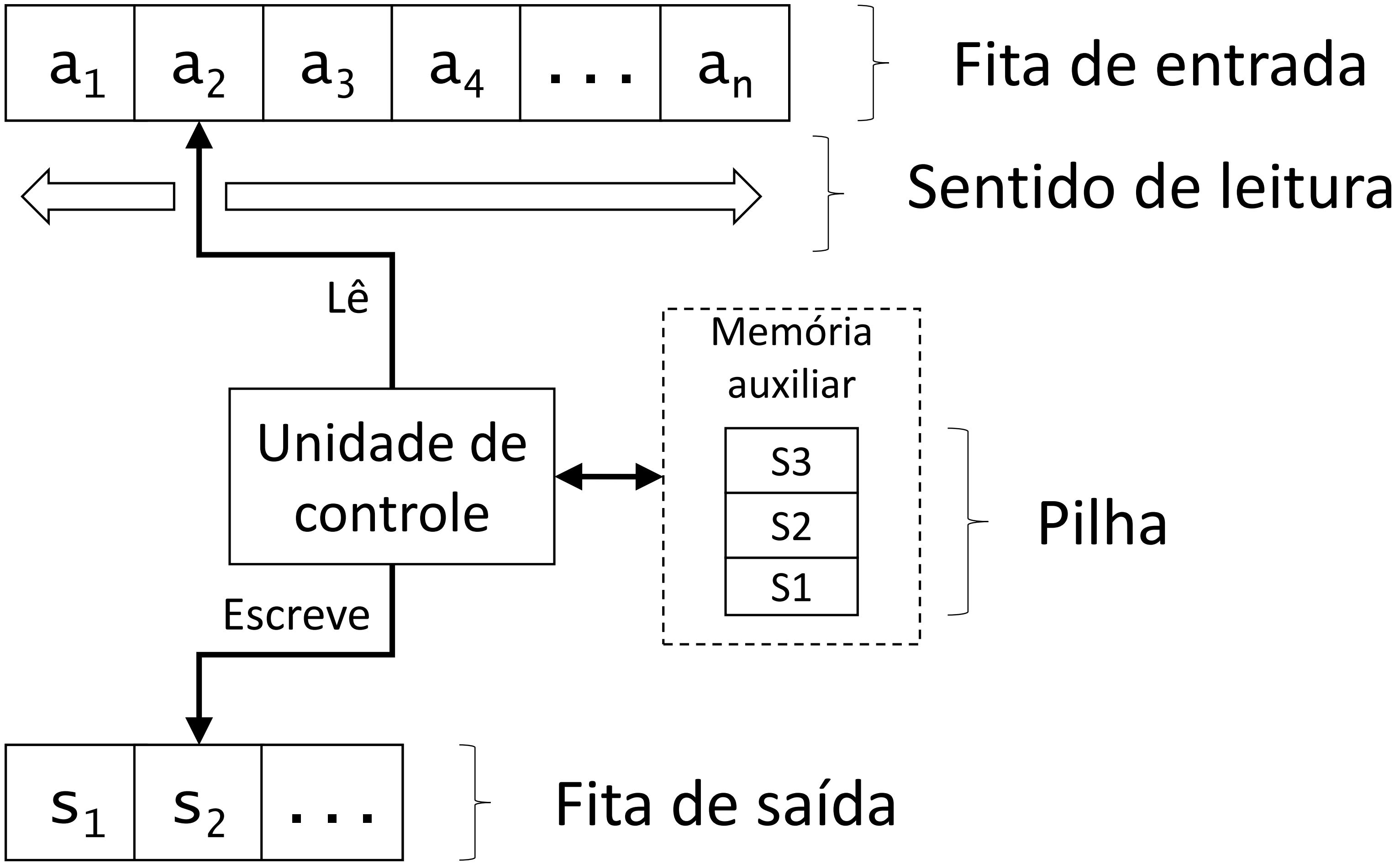
- Regras para simplificação de gramáticas livres de contexto
  - Retirada de produções vazias $A \rightarrow \lambda$
  - Retirada de produções da forma $A \rightarrow B$ que simplesmente substitui uma variável por outra
  - Retirada de variáveis ou símbolos terminais não usados

- Formas normais
  - Introduzem restrições sobre as gramáticas
  - As mais comuns são Chomsky e Greibach

Autômatos com pilha

- Autômatos com pilha são reconhecedores ou aceitadores de linguagens livre de contexto
  - Os mais usuais são os autômatos com pilha não determinísticos
  - Possuem memória auxiliar para o processamento da entrada
  - Normalmente a memória tem uma capacidade teórica infinita
  - Por ser uma pilha, o último símbolo gravado na pilha será o primeiro a ser lido

Autômatos com pilha



Autômatos com pilha

- Formalmente M = $< Q, \Sigma, \Gamma, \delta, q_0, z, F >$
  - Q: conjunto de estados finitos
  - Σ: alfabeto de entrada (conjunto finito de símbolos)
  - Γ : alfabeto da pilha (conjunto finito de símbolos)
  - δ: função de transição definida por $\delta: Q \times \Sigma \rightarrow Q$
  - q_0: estado inicial ($q_0 \in Q$)
  - z: símbolo inicial da pilha ($z \in \Gamma$)
  - F: conjunto de estados finais ($F \in Q$)

Autômato com pilha

- Funções de transição
  - Indicam qual transição será realizada quando um dado símbolo for lido
  - Porém, em autômatos com pilhas a transição também depende do símbolo que está no topo da pilha

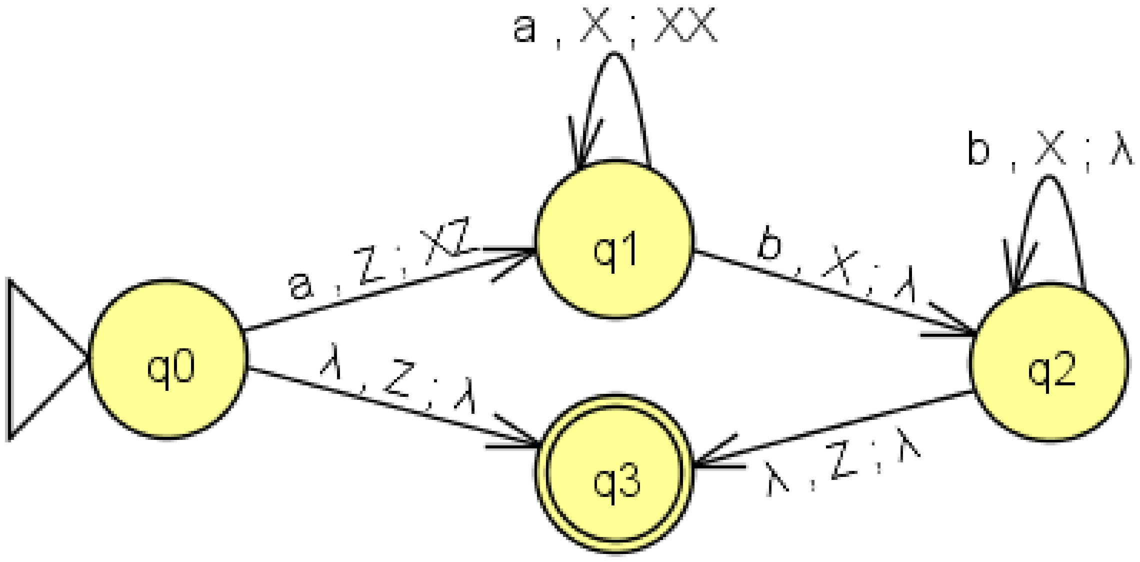
- Exemplo 1
  - $\delta(q_1, a, b) = \{(q_2, cd), (q_3, \lambda)\}$
  - Se em algum momento o autômato estiver no estado q_1 e o símbolo lido for a e o símbolo desempilhado for b, então:
    - Altera para o estado q_2 e cd é empilhado, ou
    - Altera para o estado q_3 e nada será empilhado

Autômato com pilha

- Exemplo 2
  - Dada a linguagem $L = \{ a^n b^n \mid n \geq 0 \}$,
  - M é um autômato com pilhas que aceita L
  - $M = < \{q_0,q_1,q_2,q_3\},\{a,b\},\{Z,X\},\delta,q_0,Z,\{q_3\} >$
    - $\delta(q_0,a,Z) = \{(q_1,XZ)\}$
    - $\delta(q_0,\lambda,Z) = \{(q_3,\lambda)\}$
    - $\delta(q_1,a,X) = \{(q_1,XX)\}$
    - $\delta(q_1,b,X) = \{(q_2,\lambda)\}$
    - $\delta(q_2,b,X) = \{(q_2,\lambda)\}$
    - $\delta(q_2,\lambda,Z) = \{(q_3,\lambda)\}$

Autômato com pilha

- Exemplo 2
  - Dada a linguagem $L = \{ a^n b^n \mid n \geq 0 \}$
  - M é um autômato com pilhas que aceita L
  - $M = < \{q_0, q_1, q_2, q_3\}, \{a, b\}, \{Z, X\}, \delta, q_0, Z, \{q_3\} >$
    - $\delta(q_0, a, Z) = \{(q_1, XZ)\}$
    - $\delta(q_0, \lambda, Z) = \{(q_3, \lambda)\}$
    - $\delta(q_1, a, X) = \{(q_1, XX)\}$
    - $\delta(q_1, b, X) = \{(q_2, \lambda)\}$
    - $\delta(q_2, b, X) = \{(q_2, \lambda)\}$
    - $\delta(q_2, \lambda, Z) = \{(q_3, \lambda)\}$



Autômato com pilha

- Exemplo 2 (cont.)
  - $M = < \{q_0, q_1, q_2, q_3\}, \{a, b\}, \{Z, X\}, \delta, q_0, Z, \{q_3\} >$
    - $\delta(q_0, a, Z) = \{(q_1, XZ)\}$
    - Estando no estado q_0, lido o símbolo a e desempilhado o símbolo Z, então pode ir para o estado q_1 e empilhar XZ

    - $\delta(q_0, \lambda, Z) = \{(q_3, \lambda)\}$
    - Estando no estado q_0, lido o símbolo λ e desempilhado o símbolo Z, então pode ir para o estado q_3 e não se empilha nada

    - $\delta(q_1, a, X) = \{(q_1, XX)\}$
    - Estando no estado q_1, lido o símbolo a e desempilhado o símbolo X, então pode ir para o estado q_1 e empilha-se XX

Autômato com pilha

- Exemplo 2 (cont.)
  - $M = < \{q_0,q_1,q_2,q_3\},\{a,b\},\{Z,X\},\delta,q_0,Z,\{q_3\} >$
    - $\delta(q_1,b,X) = \{(q_2,\lambda)\}$
    - Estando no estado q_1, lido o símbolo b e desempilhado o símbolo X, então pode ir para o estado q_2 e não se empilha nada

    - $\delta(q_2,b,X) = \{(q_2,\lambda)\}$
    - Estando no estado q_2, lido o símbolo b e desempilhado o símbolo X, então pode ir para o estado q_2 e não se empilha nada

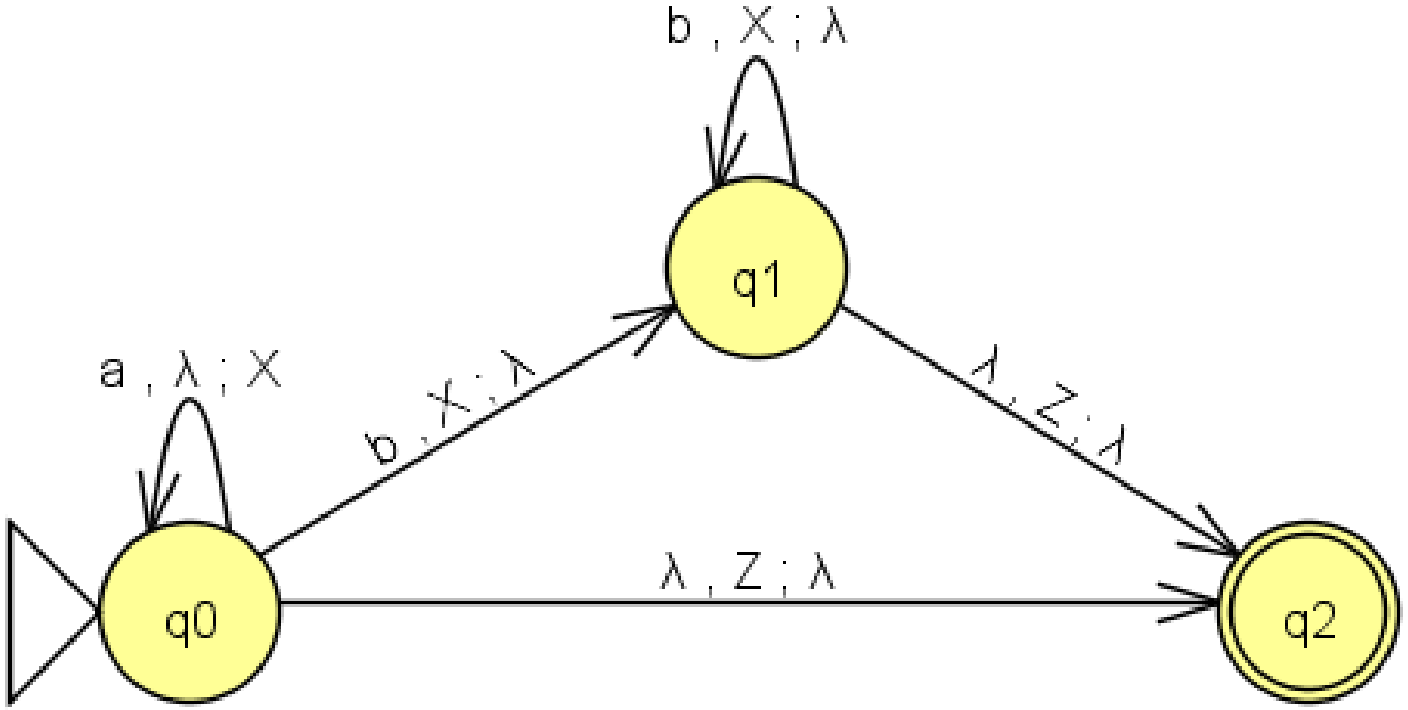
    - $\delta(q_2,\lambda,Z) = \{(q_3,\lambda)\}$
    - Estando no estado q_2, lido o símbolo λ e desempilhado o símbolo Z, então pode ir para o estado q_3 e não se empilha nada

Autômato com pilha

- Exemplo 3
  - Dada a linguagem $L = \{ a^n b^n \mid n \geq 0 \}$
  - M é um autômato com pilhas que aceita L
  - $M = < \{q_0, q_1, q_2\}, \{a, b\}, \{Z, X\}, \delta, q_0, Z, \{q_2\} >$
    - δ: ???

Autômato com pilha

- Exemplo 3
  - M = < $\{q_0,q_1,q_2\},\{a,b\},\{x,z\},\delta,q_0,Z,\{q_2\}$ >
    - $\delta(q_0,a,\lambda) = (q_0,X)$
    - $\delta(q_0,\lambda,Z) = (q_3,\lambda)$
    - $\delta(q_0,b,X) = (q_1,\lambda)$
    - $\delta(q_1,b,X) = (q_1,\lambda)$
    - $\delta(q_1,\lambda,Z) = (q_2,\lambda)$

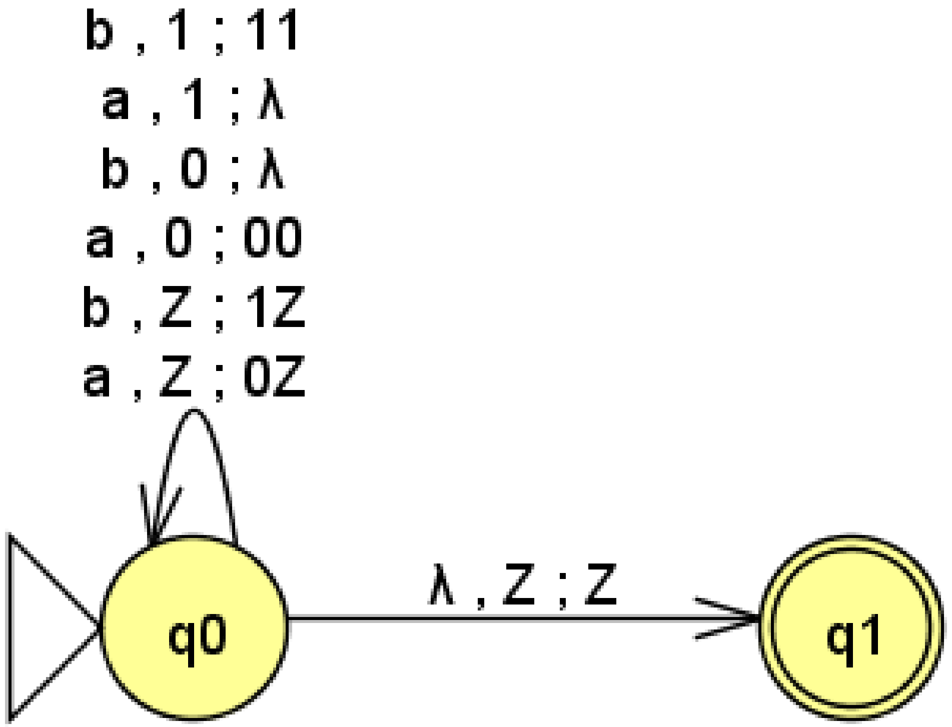


Autômato com pilha

- $L = \{w \in \{a,b\}^* \mid w$ possui o mesmo número de a e de $b\}$
- $L = \{ a^n b^m \mid n = m \}$
- Construa o autômato $M = < Q, \Sigma, \Gamma, \delta, q_0, z, F >$ onde:
  - Q: ???
  - Σ: $\{a,b\}$
  - Γ: ???
  - δ: ???

Autômato com pilha

- $L = \{w \in \{a,b\}^* \mid w$ possui o mesmo número de a e de $b\}$
- Construa o autômato $M = < Q,\Sigma,\Gamma,\delta,q_0,z,F >$ onde:
  - Q: $\{q_0,q_1\}$
  - Σ: $\{a,b\}$
  - Γ: $\{0,1,Z\}$
  - δ: funções de transição
    - $\delta(q_0,\lambda,Z) = (q_1,Z)$
    - $\delta(q_0,a,Z) = (q_0,0Z)$
    - $\delta(q_0,b,Z) = (q_0,1Z)$
    - $\delta(q_0,a,0) = (q_0,00)$
    - $\delta(q_0,b,0) = (q_0,\lambda)$
    - $\delta(q_0,a,1) = (q_0,\lambda)$
    - $\delta(q_0,b,1) = (q_0,11)$
  - F: q_1



Linguagens livres de contexto

- Gramática livre de contexto
  - Se L é uma linguagem livre de contexto, então existe uma gramática G livre de contexto tal que $L = L(G)$

- Autômato com pilhas
  - Se L é uma linguagem livre de contexto, então existe um autômato com pilha não determinístico M tal que $L = L(M)$