Compiladores

Aula 04
Unidade Jundiaí

Kroton Educacional
Anhanguera
Aqui o seu esforço ganha força.

Plano de Ensino

- Revisão de Conjuntos e Funções
- Linguagens, Expressões Regulares e Gramáticas
- **Autômatos**
- Conceitos básicos sobre compiladores e interpretadores
- Visão geral do processo de compilação
- Tipos de compiladores
- Análise léxica
- Análise sintática
- Análise semântica
- Geração de Código

Livro-Texto

- Bibliografia Básica:
  - » AHO, A.; ULLMANN, J.; REVI, S.. Compiladores : princípios, técnicas e ferramentas. 3ª ed. Rio de Janeiro: LTC, 2006.
- Bibliografia Complementar:
  - » TOSCANI, Simão Sirineo; PRICE, Ana M. A.. Implementação de Linguagens de Programação. 1ª ed. Porto Alegre: Bookman Companhia Ed., 2008.
  - » DELAMARO, Marcio Eduardo. Como Construir um Compilador : Utilizando Ferramentas Java. 1ª ed.: Novatec, 2004.

4. Autômatos – AFD

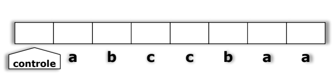
- Um Autômato Finito Determinístico (AFD) ou simplesmente Autômato Finito é sistema com um número finito de estados:
  - » Elevador
  - » Circuitos booleanos
  - » Jogos de Tabuleiros
  - » Computadores, etc.

4. Autômatos – AFD

- Um AFD é composto de três partes:
  - » Fita → dispositivo de entrada que contém a informação a ser processada;
  - » Unidade de Controle → reflete o estado corrente da máquina; possui uma unidade de leitura a qual acessa uma célula da fita de cada vez e movimenta-se para a direita.
  - » Programa ou função de transição → função que comanda as leituras e define o estado da máquina.

4. Autômatos – AFD

- A fita é finita (à esquerda e à direita), sendo dividida em células, onde cada uma armazena um símbolo. Os símbolos pertencem a um alfabeto de entrada.
- A unidade de controle possui um número finito e predefinido de estados. A unidade lê o símbolo de uma célula de cada vez. Após a leitura, a cabeça da fita move-se uma célula para a direita.
- O programa é uma função parcial que, dependendo do estado corrente e do símbolo lido, determina o novo estado do autômato.

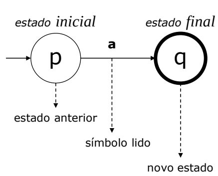


4. Autômatos – AFD

- Definição: um AFD é uma 5-upla:
  $M = (\Sigma, Q, \delta, q_0, F)$ onde:
  - Σ → alfabeto de símbolos de entrada.
  - Q → conjunto de estados possíveis do autômato o qual é finito.
  - δ → função programa ou função transição: δ: Q$\times\Sigma$→Q (*função parcial*)
  - q_0 → estado inicial, tal que $q_0 \in Q$.
  - F → conjunto de estados finais tal que $F \subseteq Q$.

4. Autômatos – AFD

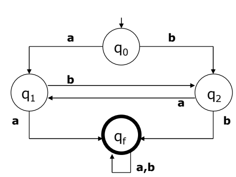
- A função programa δ pode ser interpretada como um grafo finito direto ou uma tabela de transição de estados, conforme mostrado abaixo:



δ	a
→p	q
*q	-

4. Autômatos – AFD

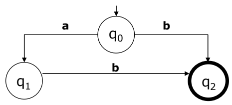
- Exemplo: considere a linguagem L_1 e o AFD M_1, conforme abaixo:
  - L_1 = {w | w possui aa ou bb como subpalavra}
  - M_1= ({a,b}, {q_0, q_1, q_2, q_f}, δ_1, q_0, {q_f}) onde δ_1 é como abaixo representado, reconhece L_1.



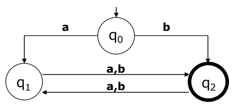
δ_1	a	b
→ q_0	q_1	q_2
q_1	q_f	q_2
q_2	q_1	q_f
* q_f	q_f	q_f

4. Autômatos – Propriedades do AFD

- Determinístico:
  - A transição dos estados ocorre apenas com uma possibilidade com cada símbolo de entrada.



- Função Total
  - A transição de cada estado processa todos os símbolos de entrada de forma determinística.



4. Autômatos – Minimização de AFD

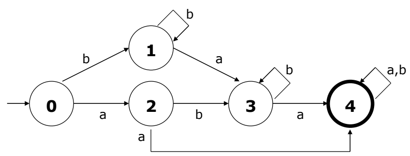
- O objetivo da minimização é gerar um Autômato Finito equivalente com o menor número de estados possíveis.
- Para que isto ocorra devem existir alguns pré-requisitos:
  - » Deve ser determinístico.
  - » Não deve ter estados inacessíveis (não-atingíveis a partir do estado inicial).
  - » A função programa deve ser total (a partir de qualquer estado são previstas transições para todos os símbolos do alfabeto).

4. Autômatos – Minimização de AFD

- Os passos para minimização são os seguintes:
  - » Construir uma tabela relacionando os estados distintos, onde cada par ocorre uma única vez.
  - » Marcação com um **x** de todos os estados do tipo *{estado final, estado não-final}*.
  - » Análise de cada par $\{q_u, q_v\}$ não-marcado, marcando-se com um ⊗ os pares deste etapa.
  - » Unificação dos pares não-marcados na tabela ao final.

4. Autômatos – Minimização de AFD

- Exemplo 1: Dado o AFD M=({a,b}, {0,1,2,3,4}, δ, {0}, {4}), onde δ é dado pelo grafo abaixo, escreva seu AFD M' equivalente minimizado.



4. Autômatos – Minimização de AFD

1-) Construção da Tabela com os estados do AFD.

1				
2				
3				
4				
	0	**1**	**2**	**3**

2-) Marcação com **x** dos estados *{final, não-final}*.

1				
2				
3				
4	x	x	x	x
	0	**1**	**2**	**3**

4. Autômatos – Minimização de AFD

3-) Análise dos pares não-marcados:

1º Par (0,1):
- δ(0,a)=2 e δ (1,a)=3 ➔ (2,3)
- δ(0,b)=1 e δ (1,b)=1 ➔ (1,1)

2º Par (0,2):
- δ(0,a)=2 e δ (2,a)=4 ➔ (2,4)
- δ(0,b)=1 e δ (2,b)=3 ➔ (1,3)

3º Par (0,3):
- δ(0,a)=2 e δ (3,a)=4 ➔ (2,4)
- δ(0,b)=1 e δ (3,b)=3 ➔ (1,3)

4º Par (1,2):
- δ(1,a)=3 e δ (2,a)=4 ➔ (3,4)
- δ(1,b)=1 e δ (2,b)=3 ➔ (1,3)

5º Par (1,3):
- δ(1,a)=3 e δ (3,a)=4 ➔ (3,4)
- δ(1,b)=1 e δ (3,b)=3 ➔ (1,3)

6º Par (2,3):
- δ(2,a)=4 e δ (3,a)=4 ➔ (4,4)
- δ(2,b)=3 e δ (3,b)=3 ➔ (3,3)

4. Autômatos – Minimização de AFD

» Analisando o 1º par (0,1), concluímos que tanto o par (2,3) não está marcado e (1,1) não é representado, então devemos colocar o par (0,1) na lista encabeçada por (2,3).

» A análise anterior serve para o par (2,3).

» Analisando o 2º par (0,2), concluímos que o par (2,4) está marcado, portanto devemos marcar com um ⊗ também o par (0,2).

» A análise anterior serve para os pares: (0,3), (1,2) e (1,3).

4. Autômatos – Minimização de AFD

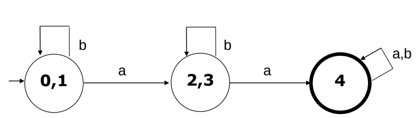
4-) Lançamento dos símbolos ⊗.

1				
2	⊗	⊗		
3	⊗	⊗		
4	x	x	x	x
	0	**1**	**2**	**3**

5-) Unificação dos pares restantes: (0,1) e (2,3).

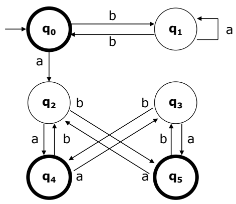
4. Autômatos – Minimização de AFD

- Portanto o AFD M=({a,b}, {0,1,2,3,4}, δ, {0}, {4}), possui um autômato equivalente M' mostrado abaixo.



4. Autômatos – Minimização de AFD

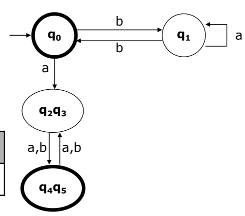
- Exemplo 2: dado o AFD M=({a,b}, {q_0,q_1,q_2,q_3,q_4,q_5}, δ, {q_0}, {q_0,q_4,q_5}), onde δ é dado pelo grafo abaixo, escreva seu AFD M' equivalente minimizado.



4. Autômatos – Minimização de AFD

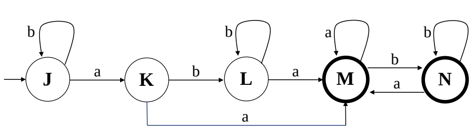
- Portanto o AFD M possui um equivalente M' mostrado abaixo.

q_1	X				
q_2	X	⊗			
q_3	X	⊗			
q_4	⊗	X	X	X	
q_5	⊗	X	X	X	
	q_0	q_1	q_2	q_3	q_4



4. Autômatos – Minimização de AFD

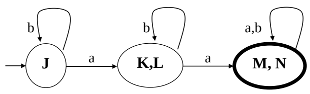
- Exemplo 3: dado o AFD M=({a,b}, {J, K, L, M, N}, δ, {J}, {M, N}), onde δ é dado pelo grafo abaixo, escreva seu AFD M' equivalente minimizado.



4. Autômatos – Minimização de AFD

- AFD M' equivalente e minimizado

K	⊗			
L	⊗			
M	x	x	x	
N	x	x	x	
	J	**K**	**L**	**M**

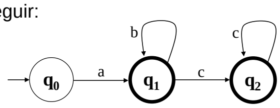


4. Autômatos – Conversão AFD - Gramática

- É possível escrever uma gramática regular para todo AFD. Para tal basta seguir o algoritmo a seguir:
  - » a cada estado é associado um não-terminal da gramática, sendo o estado inicial q_0 associado ao símbolo inicial (S)
  - » para cada transição de estado representada no grafo cria-se uma regra de produção na gramática, tal que o estado de origem torna-se o não-terminal à esquerda da regra e o estado destino torna-se um não-terminal do lado direita da regra após o terminal lido na transição.
  - » cria-se uma regra para cada não-terminal associado a um estado final onde o lado direito da regra é formado apenas pela palavra vazia (ε).

4. Autômatos – Conversão AFD - Gramática

- Exemplo: dado o AFD M = (Σ, Q, δ, q0, F) onde Σ = {a, b, c}, Q = {q_0, q_1, q_2}, δ está representada pelo grafo a seguir:



- A Gramática equivalente é dado por:
  - » Considerando-se q_0 = S, q_1 = A e q_2 = B, temos:

S → aA

A → bA | cB | ε

B → cB | ε

Compiladores
Aula 04

Engenharia da Computação
clayton.valdo@anhanguera.com