ITC: Introdução à Teoria da Computação

Marcos Castilho

DInf/UFPR

19 de maio de 2021

Linguagens Regulares

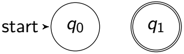
Objetivo:

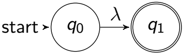
- Estabelecer relações entre ER's e AF's, Conjuntos Regulares e Linguagens Regulares

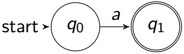
Autômatos Finitos e Conjuntos Regulares

Teorema: Todo conjunto regular é aceito por algum AFN-λ.
Prova

- Conjuntos regulares são construídos a partir de $\emptyset$, $\{\lambda\}$ e conjuntos unitários constituídos de elementos do alfabeto;
- Aqui estão os diagramas de máquinas que aceitam estes conjuntos:







Prova, continuação

Todas estas máquinas estão na forma:

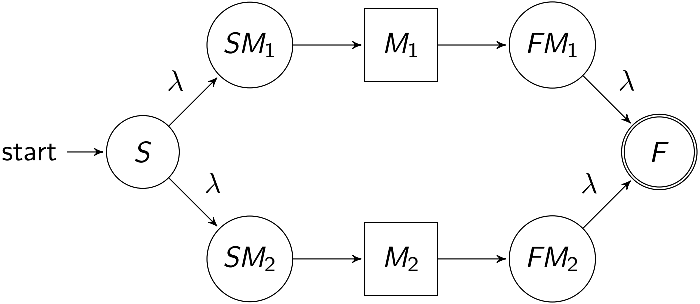
- contêm um único estado final;
- o grau de entrada de q_0 é zero.

Logo, podemos aplicar o lema da aula anterior.

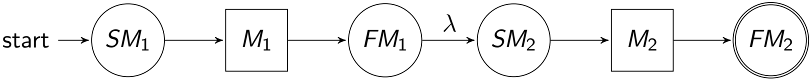
Prova, continuação

Sejam M_1 e M_2 dois AF's satisfazendo os critérios do lema. Assim, usando a mesma técnica do teorema, a partir da base, pode-se obter autômatos para qualquer conjunto regular.

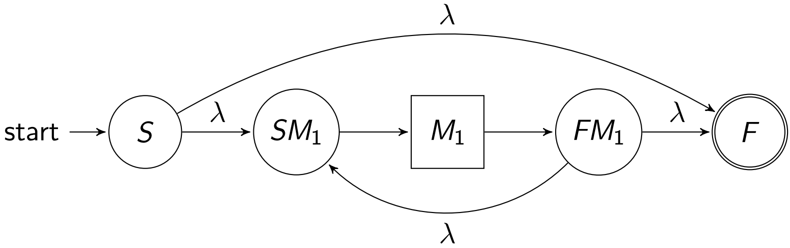
$L(M_1) \cup L(M_2)$



$L(M_1)L(M_2)$



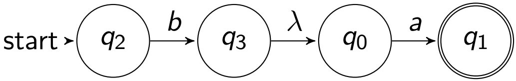
$L(M_1)^*$



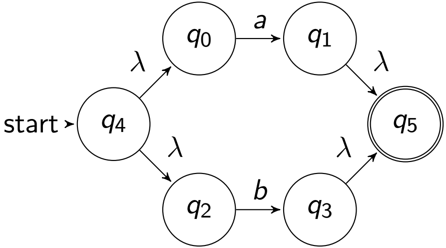
Exemplo

Obter um AFN-λ que aceita $(a+b)^*ba$:
Primeiramente, AF's para as linguagens $\{a\}$ e $\{b\}$:

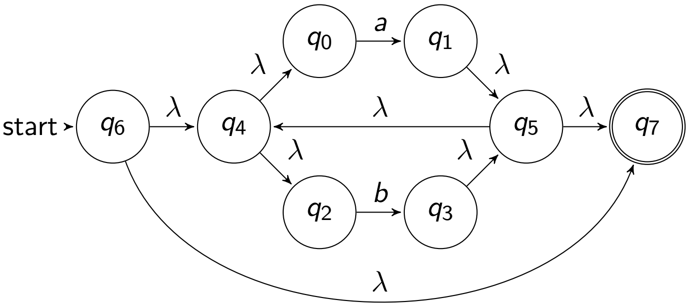
Aceita *ba*



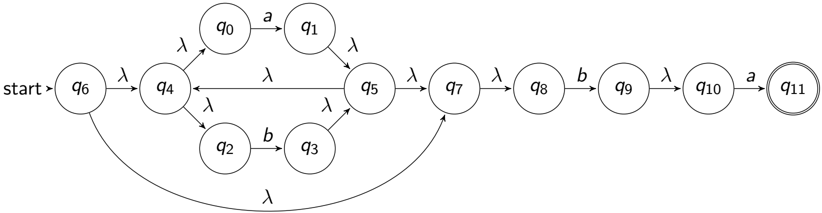
Aceita $(a + b)$



Aceita $(a + b)^*$


start
q_0
a
q_1
λ
λ
q_6
λ
q_4
λ
q_5
λ
q_7
λ
λ
q_2
b
q_3
λ

Aceita $(a+b)^*ba$



Grafos de expressão

Definição:
Um grafo de expressão é um grafo dirigido rotulado em que os arcos são rotulados por expressões regulares. Existe um nodo inicial e um conjunto de nodos finais.

Exemplo

Sejam u, v e w três expressões regulares. Então:

Aceitam respectivamente as expressões regulares u^* e u^*vw^*

Algoritmo

O algoritmo seguinte reduz qualquer grafo de AF's e produz uma expressão regular.

- O diagrama de estados pode ter qualquer número de estados finais;
- A linguagem da máquina é a união do conjunto de palavras aceitas em cada um desses estados finais;
- Assuma que os nodos são numerados e que os arcos de i até j sejam representado por $u_{i,j}$

Algoritmo AF para ER

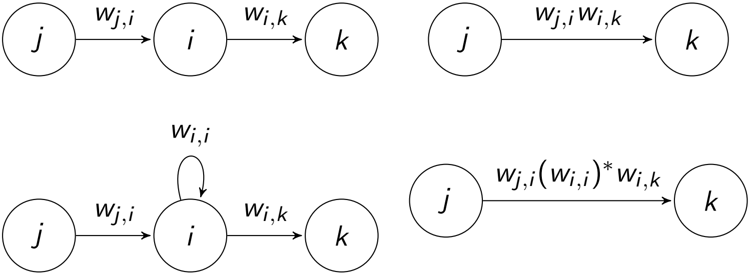
Entrada: Diagrama de estados G de um AF com nodos numerados $1, 2, \ldots, n$.

1. Seja m o número de estados finais de G.
Faça cópias de G, cada um tendo um único estado final.
Sejam $G_1, G_2, \ldots G_m$ estes grafos.
Os estados finais de G (cada um) é o estado final de algum G_i.

2. FOR cada G_T DO
   2.1 REPEAT
      2.1.1 Escolha um nodo i em G_T direrente de um estado inicial ou final
      2.1.2 Remova i de G_T assim:
         FOR todo $j, k \neq i$ (inclusive $j = k$) DO
         i) IF $w_{j,i} \neq \emptyset$ e $w_{i,i} = \emptyset$ THEN
         Ligue j a k rotulando com $w_{j,i}w_{i,k}$
         ii) IF $w_{j,i} \neq \emptyset$ e $w_{i,k} \neq \emptyset$ e $w_{i,i} \neq \emptyset$ THEN
         Ligue j a k rotulando com $w_{j,i}(w_{i,i})^* w_{i,k}$
         iii) IF os arcos que ligam j e k são rotulados $w_1, w_2, \ldots, w_s$
         THEN troque-os por um único rotulado $w_1 + w_2 + \ldots + w_s$
         iv) Remova o nodo i e os arcos incidentes a ele em G_T
   UNTIL os únicos nodos em G_T sejam os nodos inicial e final.
   Determine a expressão regular que é aceita por G_T.
END.

3. A expressão regular para G é a união das expressões regulares obtidas para cada G_T

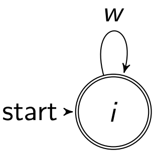
As figuras abaixo ilustram a parte principal do algoritmo:

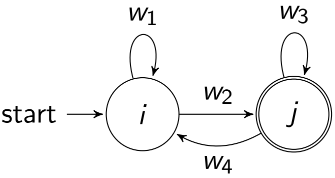


Após o algoritmo, temos dois casos:

Aceita $w_1^* w_2 (w_3 + w_4 (w_1)^* w_2)^*$

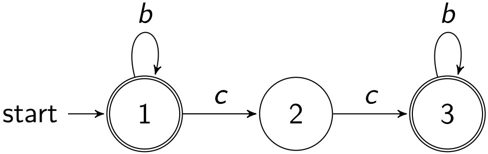
Aceita w^*



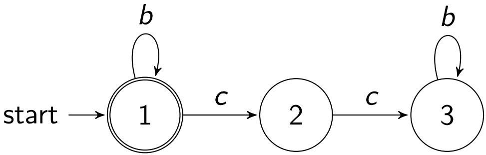


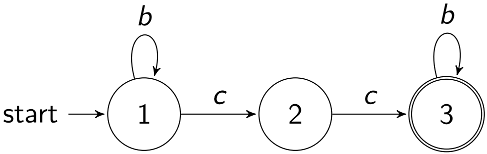
Exemplo

Seja o AFN:

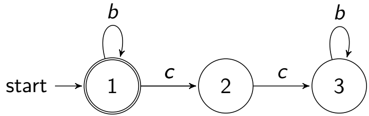


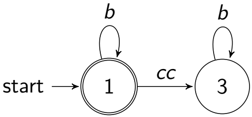
Ele deriva dois grafos

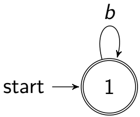




Trabalhando no primeiro:

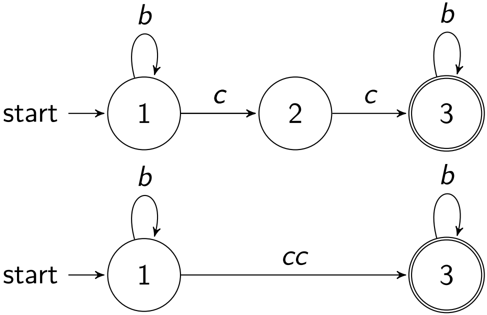






Que deriva a ER: b^*

Trabalhando no segundo:



Que deriva a ER: $b^* ccb^*$

Logo, temos a ER resultante: $b^* + b^* ccb^*$

Teorema (Kleene)

Uma linguagem L é aceita por um AFD com alfabeto Σ se, e somente se, L é um conjunto regular sobre Σ.

Licença

- Slides feitos em LaTeX usando beamer e tikz, editados com vim.
- Licença

Creative Commons Atribuição-Uso Não-Comercial-Vedada a Criação de Obras Derivadas 2.5 Brasil License.`http://creativecommons.org/licenses/by-nc-nd/2.5/br/`

Creative Commons Atribuição-Uso Não-Comercial-Vedada a Criação de Obras Derivadas 2.5 Brasil License.`http://creativecommons.org/licenses/by-nc-nd/2.5/br/`