Disciplinas MCTA015-13 / CCM-104 Linguagens Formais e Autômata / Teoria da Computação

Profa. Carla Negri Lintzmayer

carla.negri@ufabc.edu.br

www.professor.ufabc.edu.br/∼carla.negri

Centro de Matemática, Computação e Cognição – Universidade Federal do ABC

UFABC

Aviso legal 👍

Estes slides não contêm um conteúdo completo sobre ERs: são apenas um apoio gráfico a uma aula específica dada em sala.

Propriedades de expressões regulares

Propriedades das expressões regulares

Sejam A, B e C expressões regulares. Valem as seguintes propriedades:

1. $A \cup B = B \cup A$ (comut.)
2. $A \cup (B \cup C) = (A \cup B) \cup C$ (assoc.)
3. $A(BC) = (AB)C$ (assoc.)
4. $A(B \cup C) = AB \cup AC$ (distr.)
5. $(A \cup B)C = AC \cup BC$ (distr.)
6. $\emptyset A = A\emptyset = \emptyset$ (aniq.)
7. $\{\varepsilon\}A = A\{\varepsilon\} = A$ (ident.)
8. $A \cup \emptyset = \emptyset \cup A = A$ (ident.)
9. $A \cup A = A$ (idemp.)
10. $(A^*)^* = A^*$ (fecham.)
11. $\emptyset^* = \{\varepsilon\}$ (fecham.)
12. $\{\varepsilon\}^* = \{\varepsilon\}$ (fecham.)

Linguagens e expressões regulares

Linguagens e expressões regulares

Lema

A linguagem descrita por uma expressão regular é uma linguagem regular.

Linguagens e expressões regulares

Lema

A linguagem descrita por uma expressão regular é uma linguagem regular.

Demonstração.

Seja R uma expressão regular sobre um alfabeto Σ.
Queremos mostrar que $L(R)$ é uma linguagem regular.
Vamos provar isso por indução na quantidade de operações regulares de R.

Lema

A linguagem descrita por uma expressão regular é uma linguagem regular.

Lema

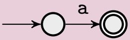
A linguagem descrita por uma expressão regular é uma linguagem regular.

Demonstração (cont.).

Caso base: R não tem operações regulares.

Então temos três casos:

1. $R = \mathbf{a}$, com a $\in \Sigma$. Nesse caso, $L(R) = \{\texttt{a}\}$. O AFN

   reconhece $L(R)$.

Linguagens e expressões regulares

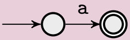
Lema

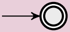
A linguagem descrita por uma expressão regular é uma linguagem regular.

Demonstração (cont.).

Caso base: R não tem operações regulares.

Então temos três casos:

1. $R = \mathbf{a}$, com $\mathtt{a} \in \Sigma$. Nesse caso, $L(R) = \{\mathtt{a}\}$. O AFN 

reconhece $L(R)$.
2. $R = \boldsymbol{\varepsilon}$. Nesse caso, $L(R) = \{\varepsilon\}$. O AFN  reconhece $L(R)$.

Linguagens e expressões regulares

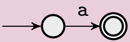
Lema

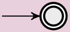
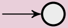
A linguagem descrita por uma expressão regular é uma linguagem regular.

Demonstração (cont.).

Caso base: R não tem operações regulares.

Então temos três casos:

1. $R = \mathbf{a}$, com $\mathtt{a} \in \Sigma$. Nesse caso, $L(R) = \{\mathtt{a}\}$. O AFN 

reconhece $L(R)$.
2. $R = \boldsymbol{\varepsilon}$. Nesse caso, $L(R) = \{\varepsilon\}$. O AFN  reconhece $L(R)$.
3. $R = \boldsymbol{\emptyset}$. Nesse caso, $L(R) = \emptyset$. O AFN  reconhece $L(R)$.

Linguagens e expressões regulares

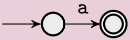
Lema

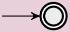
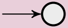
A linguagem descrita por uma expressão regular é uma linguagem regular.

Demonstração (cont.).

Caso base: R não tem operações regulares.

Então temos três casos:

1. $R = \mathbf{a}$, com $\mathtt{a} \in \Sigma$. Nesse caso, $L(R) = \{\mathtt{a}\}$. O AFN 

reconhece $L(R)$.
2. $R = \boldsymbol{\varepsilon}$. Nesse caso, $L(R) = \{\varepsilon\}$. O AFN  reconhece $L(R)$.
3. $R = \boldsymbol{\emptyset}$. Nesse caso, $L(R) = \emptyset$. O AFN  reconhece $L(R)$.

Então no caso base $L(R)$ é regular.

Expressões regulares e linguagens

Lema

A linguagem descrita por uma expressão regular é uma linguagem regular.

Demonstração (cont.).

Passo: R tem $n \geq 1$ operações regulares.

Lema

A linguagem descrita por uma expressão regular é uma linguagem regular.

Demonstração (cont.).

Passo: R tem $n \geq 1$ operações regulares.

Hipótese: Suponha que para qualquer expressão regular R' com k operações regulares, onde $0 \leq k < n$ vale que $L(R')$ é uma linguagem regular.

Expressões regulares e linguagens

Lema

A linguagem descrita por uma expressão regular é uma linguagem regular.

Demonstração (cont.).

Passo: R tem $n \geq 1$ operações regulares.

Hipótese: Suponha que para qualquer expressão regular R' com k operações regulares, onde $0 \leq k < n$ vale que $L(R')$ é uma linguagem regular.

Como R contém ao menos uma operação, temos três casos:

1. $R = R_1 \cup R_2$. Como ambas R_1 e R_2 contêm menos de n operações, vale por hipótese que $L(R_1)$ e $L(R_2)$ são regulares. Como $L(R) = L(R_1) \cup L(R_2)$ e linguagens regulares são fechadas sob união, então $L(R)$ é regular.

Lema

A linguagem descrita por uma expressão regular é uma linguagem regular.

Demonstração (cont.).

2. $R = R_1 R_2$. Por hipótese, $L(R_1)$ e $L(R_2)$ são regulares. Como $L(R) = L(R_1)L(R_2)$ e linguagens regulares são fechadas sob concatenação, então $L(R)$ é regular.

Lema

A linguagem descrita por uma expressão regular é uma linguagem regular.

Demonstração (cont.).

2. $R = R_1 R_2$. Por hipótese, $L(R_1)$ e $L(R_2)$ são regulares. Como $L(R) = L(R_1)L(R_2)$ e linguagens regulares são fechadas sob concatenação, então $L(R)$ é regular.
3. $R = R_1^*$. Por hipótese, $L(R_1)$ é regular. Como $L(R) = L(R_1)^*$ e linguagens regulares são fechadas sob concatenação, então $L(R)$ é regular.

Então $L(R)$ é uma linguagem regular. $\square$

Entendendo o resultado

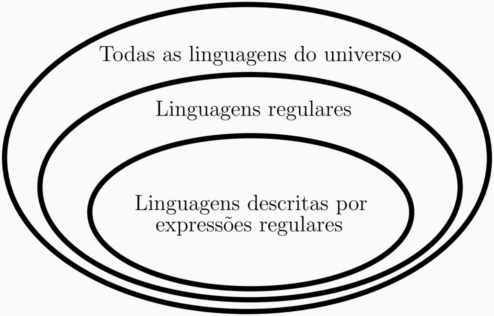
Lema

A linguagem descrita por uma expressão regular é uma linguagem regular.

Entendendo o resultado

Lema

A linguagem descrita por uma expressão regular é uma linguagem regular.



Autômatos e expressões regulares

Autômatos finitos não determinísticos generalizados

Um autômato finito não determinístico generalizado (AFNG) é uma 5-upla $(Q, \Sigma, \delta, q_i, q_f)$ em que

- Q é um conjunto finito de estados;
- Σ é um alfabeto;
- $\delta \colon (Q \setminus \{q_f\}) \times (Q \setminus \{q_i\}) \to \mathcal{R}$ é uma função de transição, em que $\mathcal{R}$ é o conjunto de todas as expressões regulares sobre Σ;
- $q_i \in Q$ é o estado inicial;
- $q_f \in Q$ é o estado final.

Lema

Toda linguagem regular pode ser descrita por uma expressão regular.

Seja A uma linguagem regular qualquer. Então existe um AFN N que reconhece A.

A ideia é transformar N em um AFNG G que seja equivalente (isto é, $L(N) = L(G)$), e então modificar o AFNG para que ele tenha apenas dois estados, o que significa que o rótulo da transição restante é a expressão regular equivalente.

Autômatos e expressões regulares

Lema

Toda linguagem regular pode ser descrita por uma expressão regular.

Demonstração.

Seja A uma linguagem regular e seja $N = (Q, \Sigma, \delta, q_0, F)$ um AFN que a reconhece.

Lema

Toda linguagem regular pode ser descrita por uma expressão regular.

Demonstração.

Seja A uma linguagem regular e seja $N = (Q, \Sigma, \delta, q_0, F)$ um AFN que a reconhece.
Criamos um novo estado inicial q_i com transição rotulada com ε para q_0.
Criamos um novo estado final q_f com transições rotuladas com ε chegando de todos os estados de F.
Se uma transição tem múltiplos rótulos, substituímos cada uma por uma única transição com rótulo igual à união dos rótulos anteriores.
Por fim, coloca-se transições rotuladas com $\emptyset$ entre estados que não tenham transições.

Autômatos e expressões regulares

Lema

Toda linguagem regular pode ser descrita por uma expressão regular.

Demonstração.

Seja A uma linguagem regular e seja $N = (Q, \Sigma, \delta, q_0, F)$ um AFN que a reconhece.
Criamos um novo estado inicial q_i com transição rotulada com ε para q_0.
Criamos um novo estado final q_f com transições rotuladas com ε chegando de todos os estados de F.
Se uma transição tem múltiplos rótulos, substituímos cada uma por uma única transição com rótulo igual à união dos rótulos anteriores.
Por fim, coloca-se transições rotuladas com $\emptyset$ entre estados que não tenham transições.
Chame tal AFNG de G e veja que $L(N) = L(G)$. (!!)

Lema

Toda linguagem regular pode ser descrita por uma expressão regular.

Demonstração (cont.).

Agora devemos reduzir o número de estados de G, usando um procedimento que mantém a nova máquina equivalente.

Autômatos e expressões regulares

Lema

Toda linguagem regular pode ser descrita por uma expressão regular.

Demonstração (cont.).

Agora devemos reduzir o número de estados de G, usando um procedimento que mantém a nova máquina equivalente.

Ideia: se um estado C será removido, qualquer outro par de estados A e B será afetado:

Lema

Toda linguagem regular pode ser descrita por uma expressão regular.

Demonstração (cont.).

A remoção de estados de G é feita pelo seguinte algoritmo.

Função $\textsc{Converte\_AFNG\_em\_ER}(G = (Q, \Sigma, \delta, q_i, q_f))$

Se $|Q| == 2$ **então Devolve** $\delta(q_i, q_f)$

Lema

Toda linguagem regular pode ser descrita por uma expressão regular.

Demonstração (cont.).

A remoção de estados de G é feita pelo seguinte algoritmo.

Função $\textsc{Converte\_AFNG\_em\_ER}(G = (Q, \Sigma, \delta, q_i, q_f))$

  Se $|Q| == 2$ **então Devolve** $\delta(q_i, q_f)$

  seja $C \in Q \setminus \{q_i, q_f\}$ o estado a ser removido

  Para cada $A \in Q \setminus \{C, q_f\}$ **faça**

    Para cada $B \in Q \setminus \{C, q_i\}$ **faça**

Lema

Toda linguagem regular pode ser descrita por uma expressão regular.

Demonstração (cont.).

A remoção de estados de G é feita pelo seguinte algoritmo.

Função $\textsc{Converte\_AFNG\_em\_ER}(G = (Q, \Sigma, \delta, q_i, q_f))$
    Se $|Q| == 2$ **então Devolve** $\delta(q_i, q_f)$
    seja $C \in Q \setminus \{q_i, q_f\}$ o estado a ser removido
    Para cada $A \in Q \setminus \{C, q_f\}$ **faça**
        Para cada $B \in Q \setminus \{C, q_i\}$ **faça**
            seja $R_4 = \delta(A, B)$
            seja $R_1 = \delta(A, C)$
            seja $R_2 = \delta(C, C)$
            seja $R_3 = \delta(C, B)$
            faça $\varphi(A, B) = R_1 R_2^* R_3 \cup R_4$

Autômatos e expressões regulares

Lema

Toda linguagem regular pode ser descrita por uma expressão regular.

Demonstração (cont.).

A remoção de estados de G é feita pelo seguinte algoritmo.

Função $\textsc{Converte\_AFNG\_em\_ER}(G = (Q, \Sigma, \delta, q_i, q_f))$

  Se $|Q| == 2$ **então Devolve** $\delta(q_i, q_f)$

  seja $C \in Q \setminus \{q_i, q_f\}$ o estado a ser removido

  Para cada $A \in Q \setminus \{C, q_f\}$ **faça**

    Para cada $B \in Q \setminus \{C, q_i\}$ **faça**

      seja $R_4 = \delta(A, B)$

      seja $R_1 = \delta(A, C)$

      seja $R_2 = \delta(C, C)$

      seja $R_3 = \delta(C, B)$

      faça $\varphi(A, B) = R_1 R_2^* R_3 \cup R_4$

  construa $G' = (Q \setminus \{C\}, \Sigma, \varphi, q_i, q_f)$

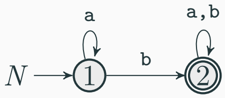
  Devolve $\textsc{Converte\_AFNG\_em\_ER}(G')$

Resta provar que a expressão devolvida por $\textsc{Converte\_AFNG\_em\_ER}(G)$ é equivalente a G... □

Exemplo de execução

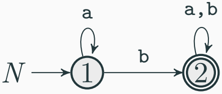
Um exemplo de conversão

Vamos encontrar uma expressão regular equivalente ao seguinte AFN N:

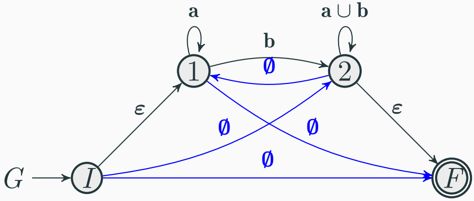


Um exemplo de conversão

Vamos encontrar uma expressão regular equivalente ao seguinte AFN N:



Primeiro, transformamos N no AFNG G equivalente:

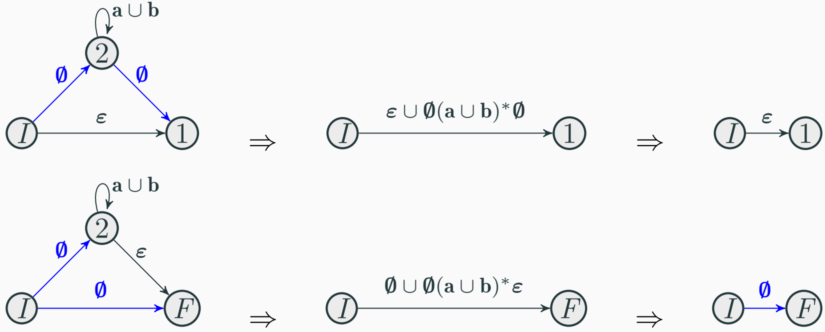


E agora executamos $\textsc{Converte\_AFNG\_em\_ER}(G)$.

Autômatos e expressões regulares

Como $|Q| = 4 > 2$, vamos remover o estado $C = 2$. Nesse caso, A pode assumir valores em $\{I, 1\}$ e B pode assumir valores em $\{1, F\}$.

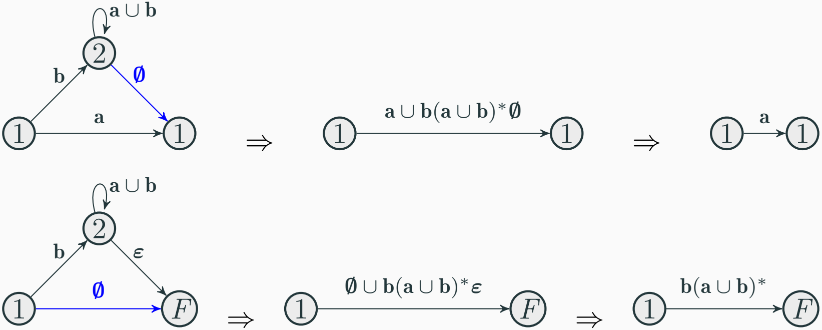
Assim, as novas transições serão



Autômatos e expressões regulares

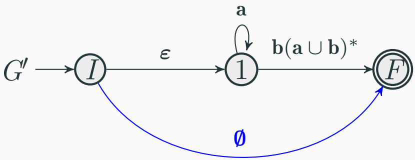
Como $|Q| = 4 > 2$, vamos remover o estado $C = 2$. Nesse caso, A pode assumir valores em $\{I, 1\}$ e B pode assumir valores em $\{1, F\}$.

Assim, as novas transições serão



Autômatos e expressões regulares

Agora temos

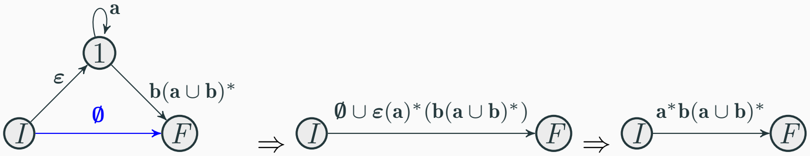


Vamos executar $\textsc{Converte\_AFNG\_em\_ER}(G')$.

Autômatos e expressões regulares

Como $|Q| = 3 > 2$, vamos remover o estado $C = 1$. Nesse caso, A pode assumir valores em $\{I\}$ e B pode assumir valores em $\{F\}$.

Assim, as novas transições serão



Autômatos e expressões regulares

Agora temos

$$G'' \longrightarrow I \xrightarrow{\mathbf{a}^*\mathbf{b}(\mathbf{a} \cup \mathbf{b})^*} F$$

Ao executar $\textsc{Converte\_AFNG\_em\_ER}(G'')$, entramos no caso base, pois $|Q| == 2$.

O algoritmo então devolve $\mathbf{a}^*\mathbf{b}(\mathbf{a} \cup \mathbf{b})^*$.

Implicações

Implicações

Corolário

Uma linguagem é regular se e somente se alguma expressão regular a descreve.

Demonstração.

Seja A uma linguagem qualquer.

$(\Rightarrow)$ Suponha que A é regular.

Então pelo lema mais recente, existe uma expressão regular que a descreve.

Corolário

Uma linguagem é regular se e somente se alguma expressão regular a descreve.

Demonstração.

Seja A uma linguagem qualquer.

($\Rightarrow$) Suponha que A é regular.

Então pelo lema mais recente, existe uma expressão regular que a descreve.

($\Leftarrow$) Suponha que uma expressão regular descreve A. Então pelo primeiro lema, A é regular. $\square$

Implicações