Expressões Regulares

EDUARDO FREIRE NAKAMURA

Instituto de Computação
Universidade Federal do Amazonas
nakamura@icomp.ufam.edu.br

[1]Este material utiliza conteúdo das aulas fornecidas pelo Prof. Vilar da Câmara Neto (disponível em http://http://prof.vilarneto.com).
[2]Permissão de uso fornecida pelos autores.
[3]As figuras utilizadas neste material são de domínio público, disponíveis na Internet sem informações de direitos autorais.

Introdução

- AFDs, AFNs, AFNEs e AFNλs são formalismos úteis para a construção de diversas linguagens de uso prático
- As Gramáticas, por sua vez, permitem descrever facilmente linguagens artificiais usadas em computadores, como as linguagens de programação
- Ambos os formalismos são difíceis de serem expressados de maneira compacta

Introdução

- As Expressões Regulares (ERs) são uma ferramenta para a expressão de linguagens de maneira compacta
- As ERs são tão úteis na prática que são implementadas em diversas ferramentas computacionais:
  - Editores de texto
  - Comandos de sistemas operacionais
  - Linguagens de programação
  - ...

Expressões regulares

- As ERs são representações de linguagens por meio de operações sobre conjuntos, porém permitem apenas uma pequena gama de operações
- Na forma mais simples, as ERs permitem apenas as seguintes operações
  - Agrupamento
  - Fecho de Kleene
  - Concatenação
  - União
- Nas ERs conjuntos não são denotados por meio de { e }

Operadores

- A representação de símbolos é feita simplesmente pelo próprio símbolo

  Exemplo: $L = \{0\}$, temos $r = 0$

- A representação de $\{\lambda\}$ é feita simplesmente por λ

  Exemplo: $L = \{\lambda\}$, temos $r = \lambda$

- A concatenação, assim como na representação por conjuntos, é feita pela sequência daquilo que se quer concatenar
  - *Exemplo:* $L = \{0\}\{1\}\{0\}\{1\}$, temos $r = 0101$

Operadores

- A união é representada pelo símbolo de adição("+")
  - *Exemplo 1:* $L = \{a, b\}$, temos $r = a+b$
  - *Exemplo 2:* $L = \{1, 01, 23\}$, temos $r = 1+01+23$
  - *Exemplo 3:* $L = \{12\} \cup \{21\}$, temos $r = 12+21$

- O agrupamento é representado por um par de parênteses
  - *Exemplo 1:* $L = \{aa\}(\{bb\} \cup \{cc\})$, temos $r = aa(bb+cc)$
  - *Exemplo 2:* $L = \{1\}\{2, 3\}$, temos $r = 1(2+3)$
  - *Exemplo 3:* $L = \{0,1\}\{111\}\{0,00,1,11\}$, temos $r =(0+1)111(0+00+1+11)$

Operadores

- O fecho de Kleene é representado por um asterisco (“*”)
  - *Exemplo 1: L* = {x}*, temos *r* = x*
  - *Exemplo 2: L* = {0}*{1}*, temos *r* = 0*1*
  - *Exemplo 3: L* = {00}*, temos *r* = (00)*

- Atenção
  - O fecho de Kleene tem precedência sobre a concatenação!
  - E a concatenaçãotem precedência sobre a união!
  - Portanto, a ER 01* é o mesmo que 0(1*), e não (01)*

ER nula

- A ER que não aceita nenhuma palavra é denotada por $\varnothing$
  - $r = \varnothing$ é diferente de $r = \lambda$

- Propriedades da ER nula
  - A concatenação de qualquer ER com a ER nula equivale àER nula

$$r = r_1 \varnothing r_2 \Rightarrow r = \varnothing$$

  - A união da ER nula com qualquer outra ER não altera o resultado

$$r = r1 + \varnothing \Rightarrow r = r_1$$

$$r = r_1 + r_2 + \varnothing + r_3 \Rightarrow r = r_1 + r_2 + r_3$$

  - O fecho de Kleene sobre a ER nula gera a ER λ

$$\varnothing^* = \lambda$$

Exemplos

Usando conjuntos		Usando ER
$L = \{0,1\}^*\{\lambda,0,1\}$	$\Rightarrow$	$r = (0+1)^*(\lambda+0+1)$
$L = (\{01\}\{0,1\})^*$	$\Rightarrow$	$r = (01(0+1))^*$
$L = \{01\}^+$	$\Rightarrow$	$r = 01(01)^*$
$L = \{ab\}\{a,b\}\{aa,bb\}^*\{aaa\}$	$\Rightarrow$	$r = ab(a+b)(aa+bb)^*aaa$
$L = (\{ab\}\{c\}^*)^+$	$\Rightarrow$	$r = abc^*(abc^*)^*$
$L = (\{ab\}\{c\}^+)^+$	$\Rightarrow$	$r = abcc^*(abcc^*)^*$

A forma básica das ERs não permite repetições do tipo "uma ou mais vezes"

Eduardo Freire Nakamura (nakamura@icomp.ufam.edu.br)

Linguagens Associadas com Expressões Regulares

- Exemplos: exemplos de ERs em $\Sigma = \{a,b\}$ e suas linguagens:
  - r = aa → Somente a palavra aa
  - r = ba* → Palavras que iniciam com b, seguido por zero ou mais a´s
  - r = (a+b)* → Todas as palavras sobre {a,b}
  - r = (a+b)*aa(a+b)* → Todas as palavras contendo aa como subpalavra
  - r = a*ba*ba* → Todas as palavras contendo exatamente dois b´s
  - r = (a+b)*(aa+bb) → Todas as palavras que terminam com aa ou bb

Linguagens Associadas com Expressões Regulares

- Exemplos: Dado o alfabeto $\Sigma = \{0, 1\}$, diga quais são as linguagens denotadas pelas seguintes ERs:

a) 0^*10^*
b) $\Sigma^*1\Sigma^*$
c) $\Sigma^*001\Sigma^*$
d) $(\Sigma\Sigma)^*$
e) $(\Sigma\Sigma\Sigma)^*$

Eduardo Freire Nakamura (nakamura@icomp.ufam.edu.br)

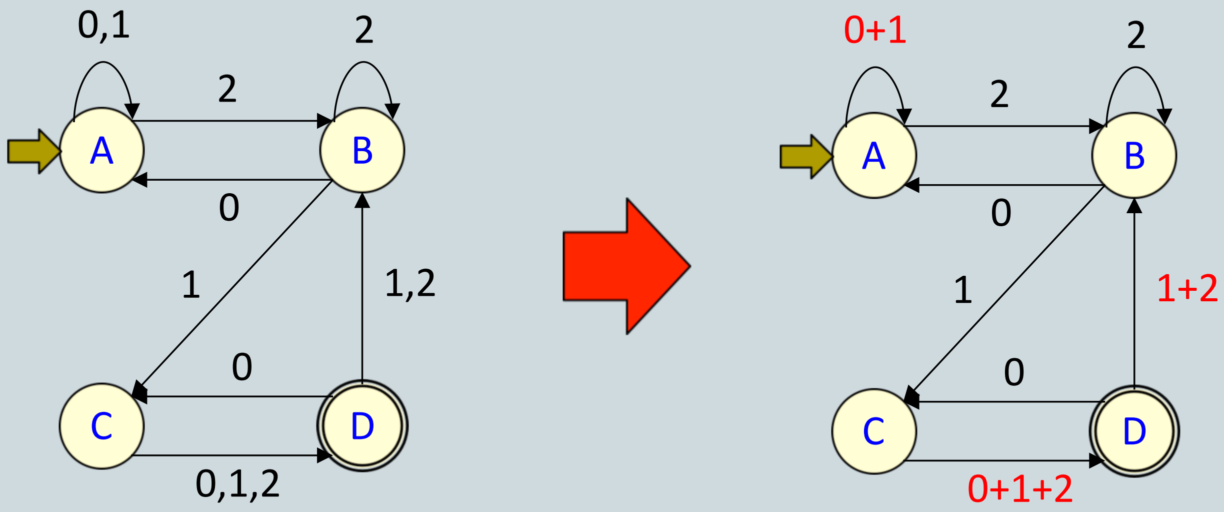
Representatividade

- Uma característica importante das ERs é que elas possuem o mesmo poder de representação de linguagens dos AFDs
  - Qualquer AF estudado até agora pode ser transformado em um ER e vice-versa
  - O procedimento toma por base que há apenas um estado final, identificado por *f* a partir de agora
  - A generalização para um AF com vários estados finais será vista posteriormente
  - O processo de transformação de um AF em ER é dividido em três passos

Passo #01: Transformar o AF em um diagrama de ER

- Um diagrama de ER é um diagrama de estados (similar a um AF) em que as transições são feitas sobre expressões regulares, ao invés de serem feitas sobre símbolos ou palavras do alfabeto

Exemplo

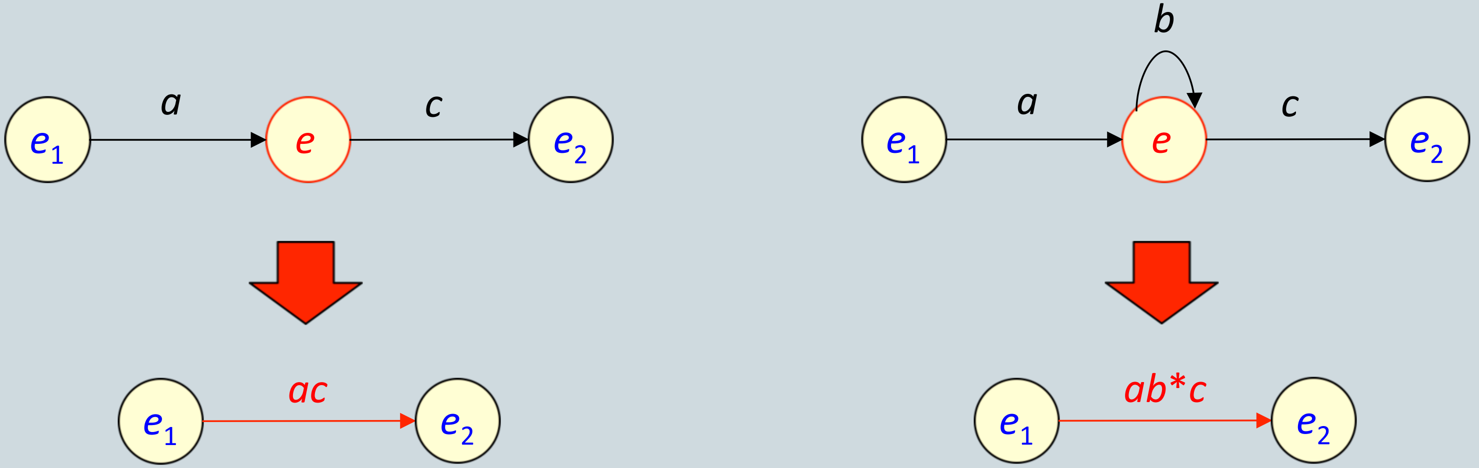


Eduardo Freire Nakamura (nakamura@icomp.ufam.edu.br)

Transformação de AFs em ERs

Passo #02: Eliminar estados que não são iniciais nem finais

- O objetivo é considerar todas as combinações de transições de entrada e de saída para o estado que será eliminado (aqui representado por *e*)

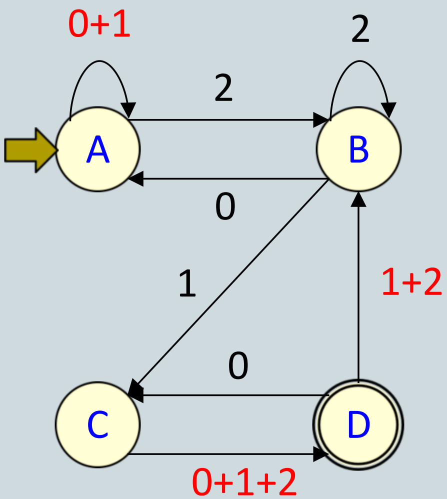


*Se não houver transição de saída do estado que será eliminado ("estado lixeira"), o estado pode ser simplesmente eliminado.

Transformação de AFs em ERs

Passo #02: Eliminar estados que não são iniciais nem finais

- De volta ao exemplo

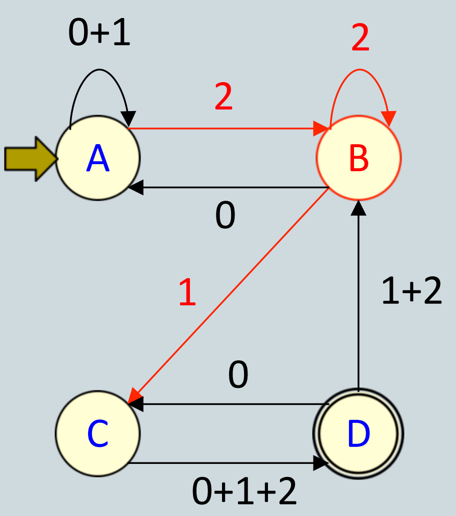


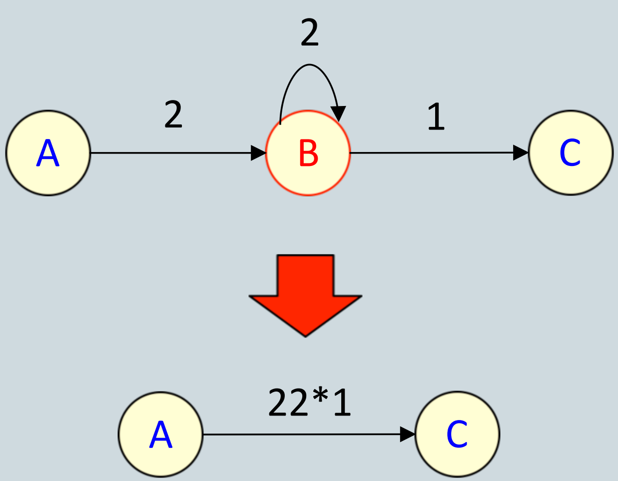
Transformação de AFs em ERs

Passo #02: Eliminar estados que não são iniciais nem finais

- De volta ao exemplo

Eliminando o estado B (caminho A-B-C)





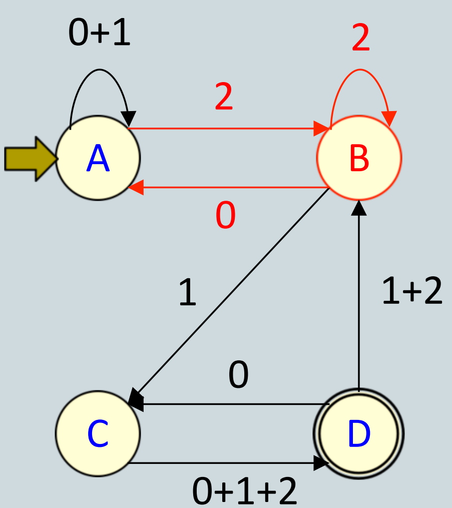
Eduardo Freire Nakamura (nakamura@icomp.ufam.edu.br)

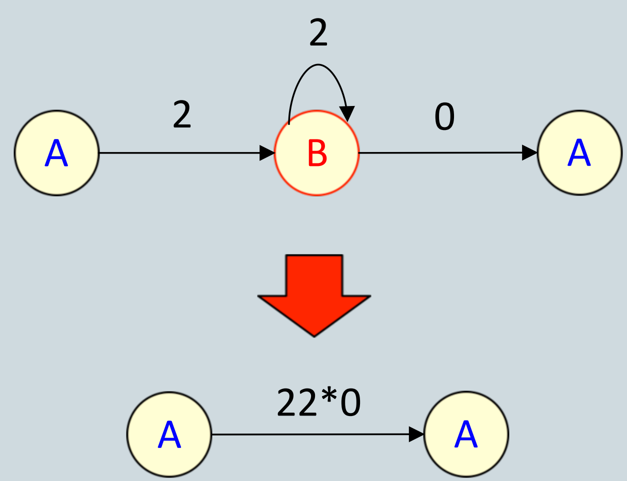
Transformação de AFs em ERs

Passo #02: Eliminar estados que não são iniciais nem finais

- De volta ao exemplo

Eliminando o estado B (caminho A-B-A)



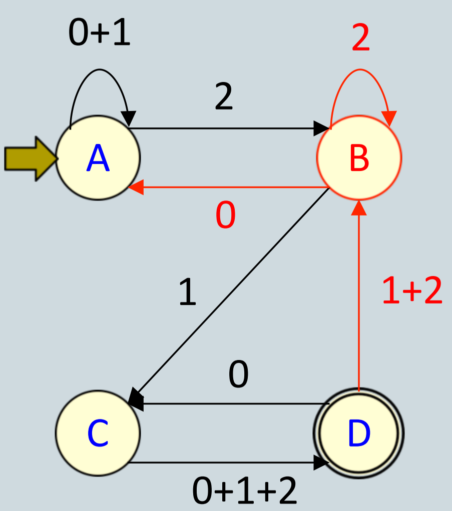


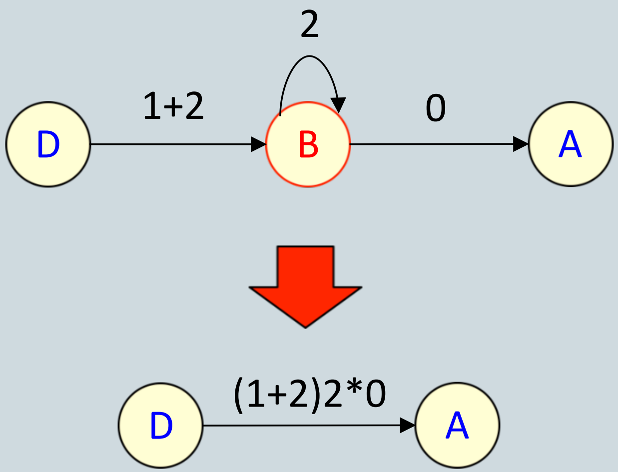
Transformação de AFs em ERs

Passo #02: Eliminar estados que não são iniciais nem finais

- De volta ao exemplo

Eliminando o estado B (caminho D-B-A)



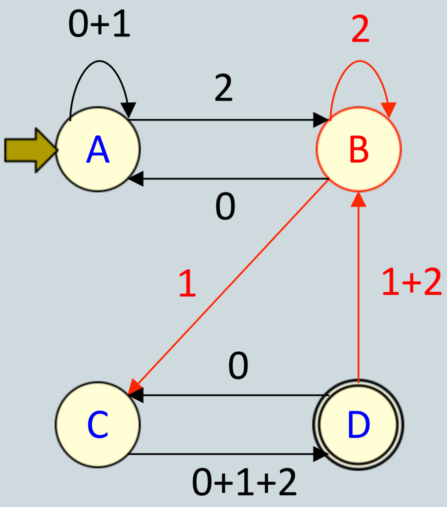


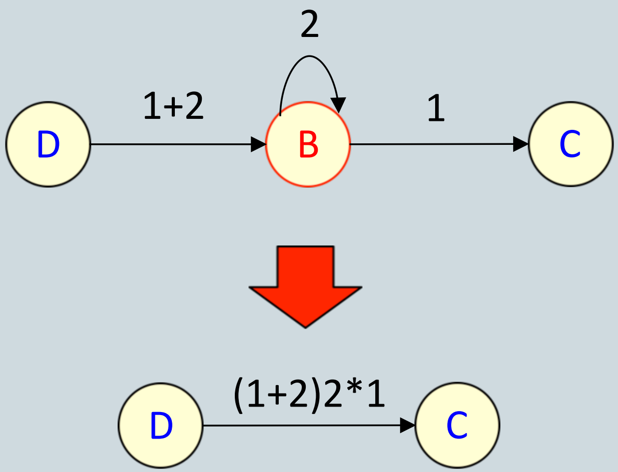
Transformação de AFs em ERs

Passo #02: Eliminar estados que não são iniciais nem finais

- De volta ao exemplo

Eliminando o estado B (caminho D-B-C)



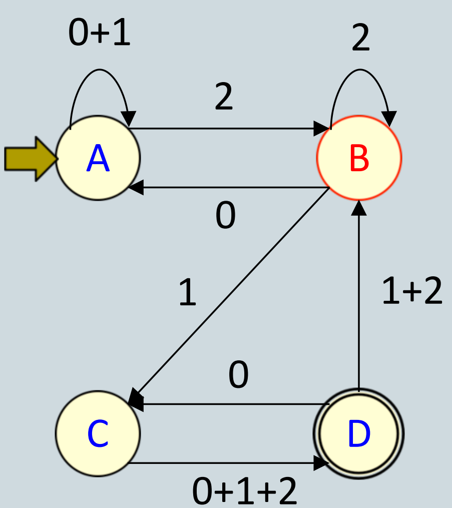


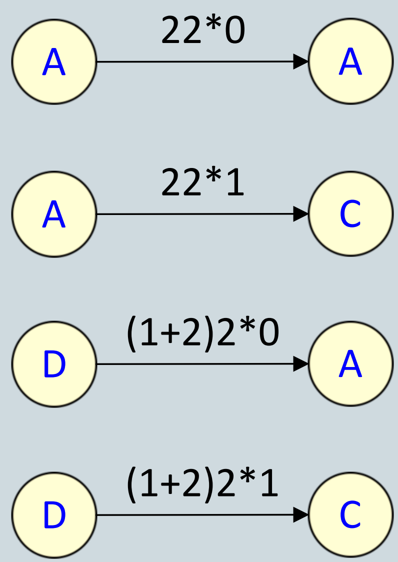
Transformação de AFs em ERs

Passo #02: Eliminar estados que não são iniciais nem finais

- De volta ao exemplo

Eliminando o estado B



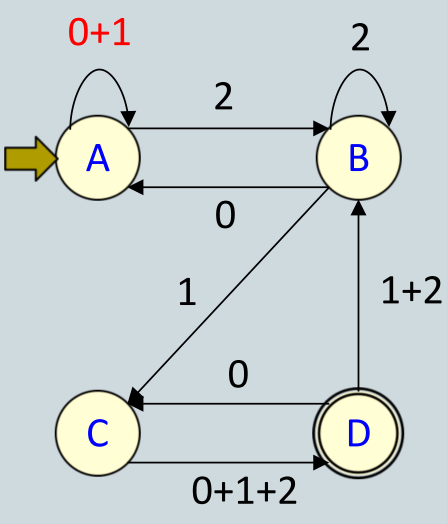


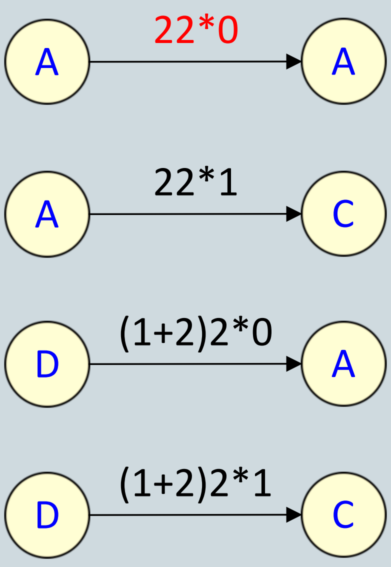
Transformação de AFs em ERs

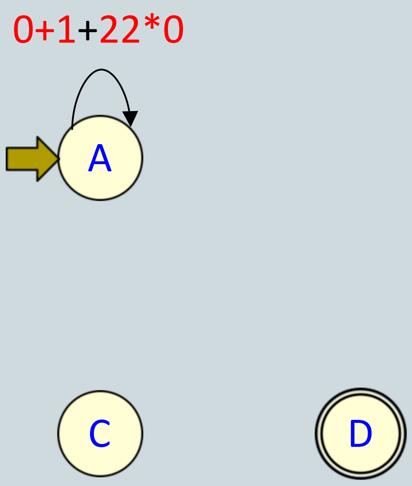
Passo #02: Eliminar estados que não são iniciais nem finais

- De volta ao exemplo

Eliminando o estado B



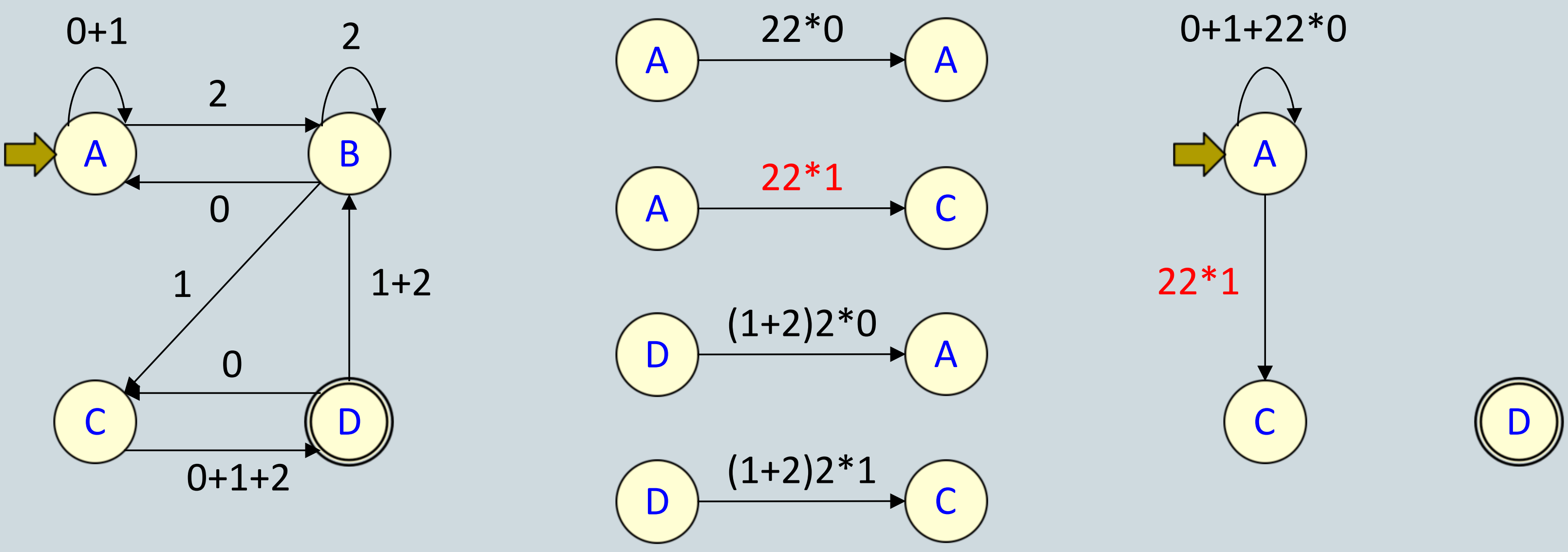




Passo #02: Eliminar estados que não são iniciais nem finais

- De volta ao exemplo

Eliminando o estado B

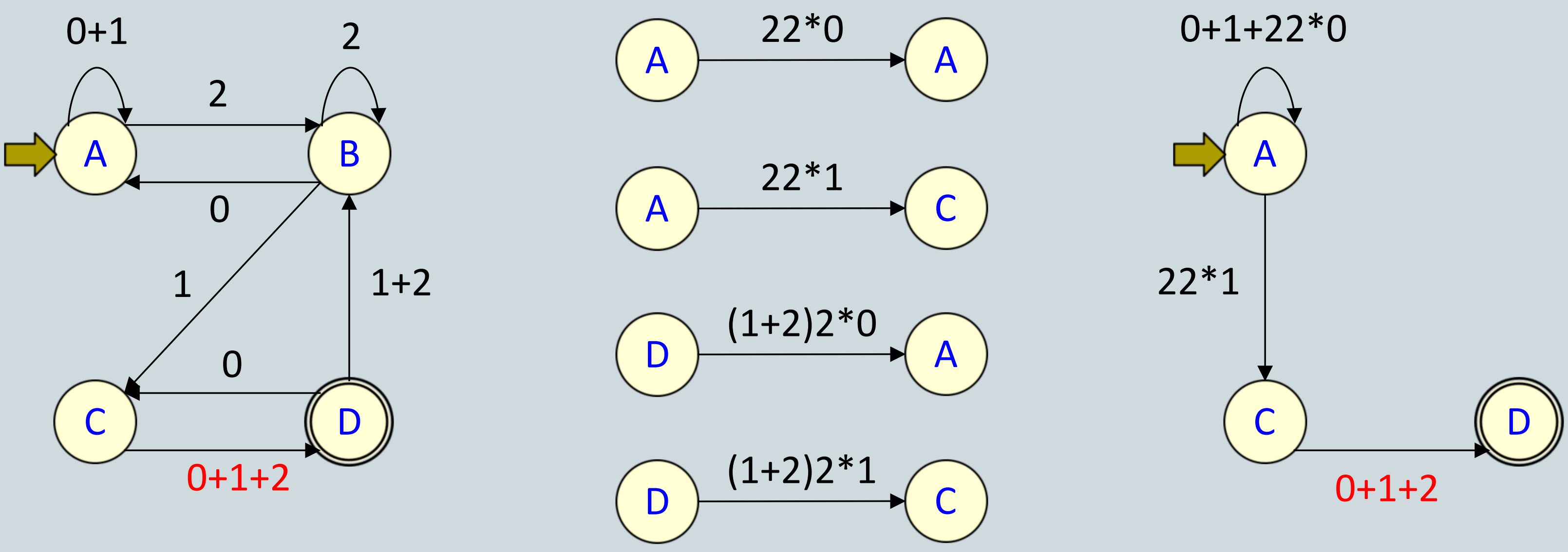


Transformação de AFs em ERs

Passo #02: Eliminar estados que não são iniciais nem finais

- De volta ao exemplo

Eliminando o estado B

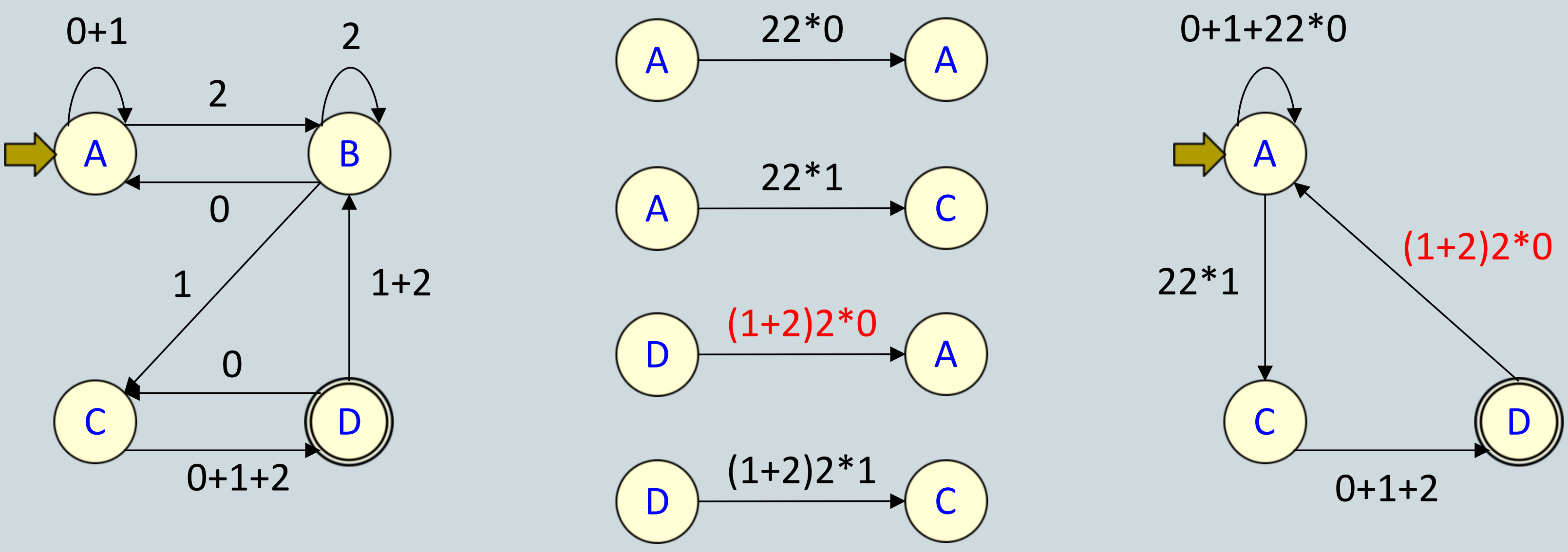


Transformação de AFs em ERs

Passo #02: Eliminar estados que não são iniciais nem finais

- De volta ao exemplo

Eliminando o estado B



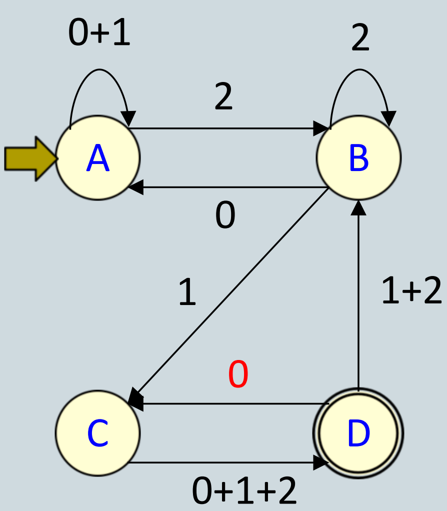
Eduardo Freire Nakamura (nakamura@icomp.ufam.edu.br)

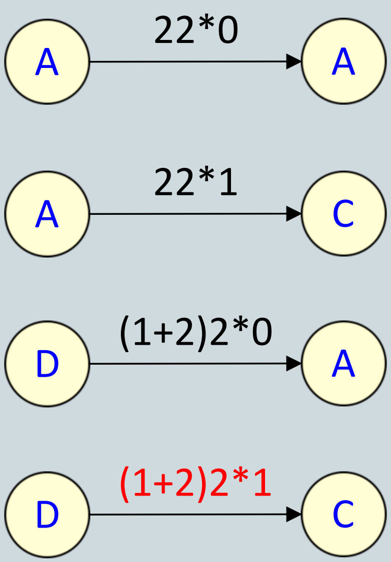
Transformação de AFs em ERs

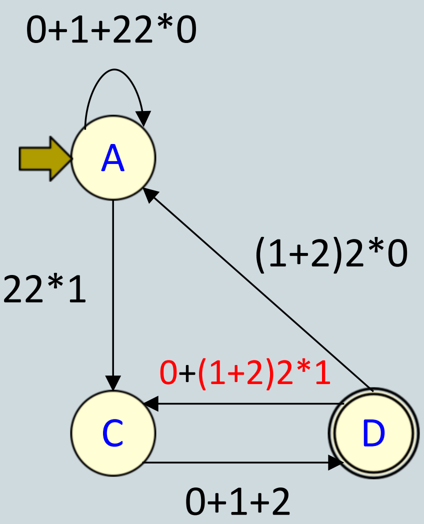
Passo #02: Eliminar estados que não são iniciais nem finais

- De volta ao exemplo

Eliminando o estado B





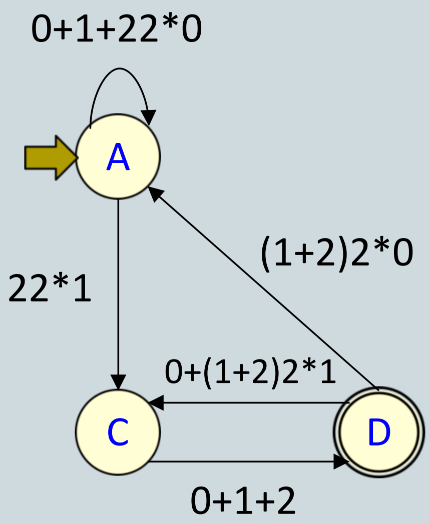


Transformação de AFs em ERs

Passo #02: Eliminar estados que não são iniciais nem finais

- De volta ao exemplo

  Eliminando o estado C

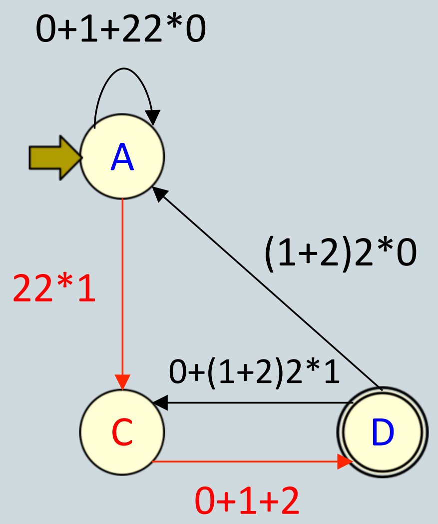


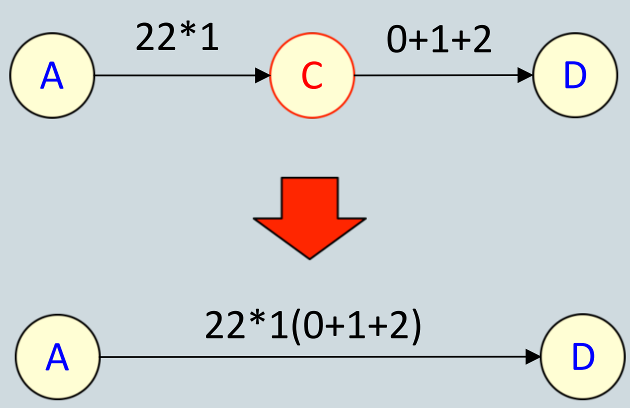
Transformação de AFs em ERs

Passo #02: Eliminar estados que não são iniciais nem finais

- De volta ao exemplo

Eliminando o estado C (caminho A-C-D)



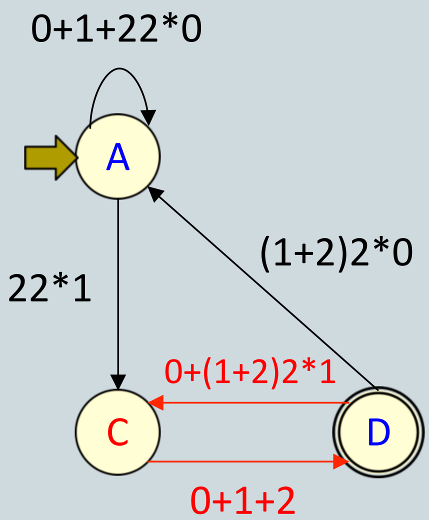


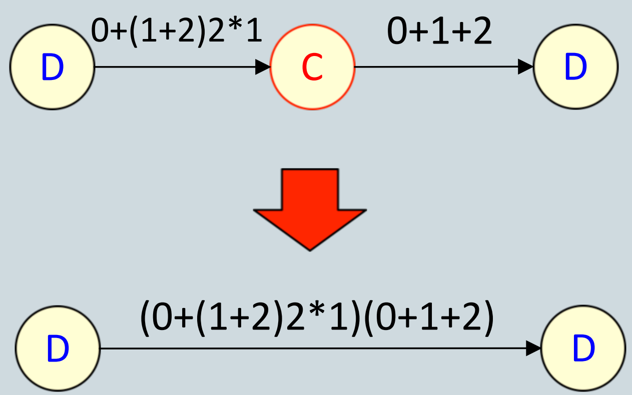
Transformação de AFs em ERs

Passo #02: Eliminar estados que não são iniciais nem finais

- De volta ao exemplo

Eliminando o estado C (caminho D-C-D)



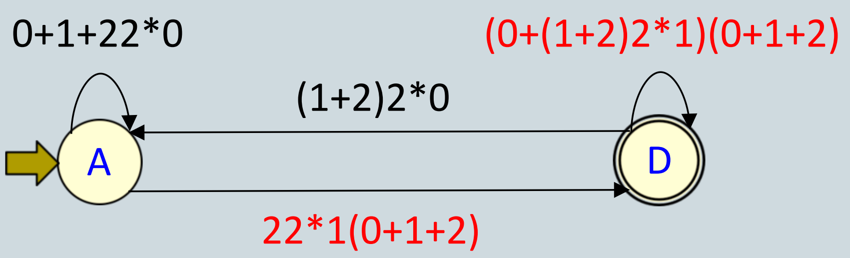


Transformação de AFs em ERs

Passo #02: Eliminar estados que não são iniciais nem finais

- De volta ao exemplo

Diagrama ER eliminando os estados B e C

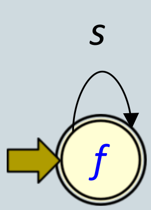


Transformação de AFs em ERs

Passo #03: Unindo estado inicial e final *f* do diagrama de ER

1. Possibilidade #01

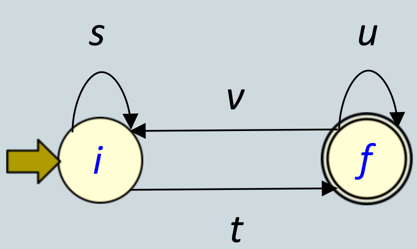
   Estado final igual ao inicial



   A ER resultante é $r = s^*$

2. Possibilidade #02

   Estado final diferente do inicial



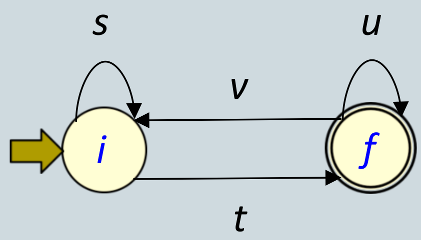
   A ER resultante é $r = s^*t(u + vs^*t)^*$

Transformação de AFs em ERs

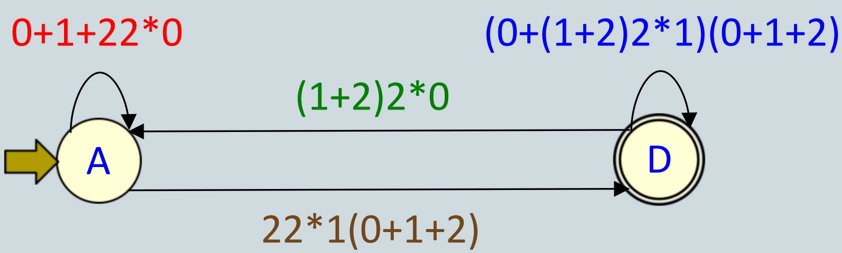
Passo #03: Unindo estado inicial e final *f* do diagrama de ER

- De volta ao exemplo

ER resultante eliminando os estados inicial e final



$r = s^*t(u + vs^*t)^*$



$r =$ (0+1+22*0)*22*1(0+1+2)((0+(1+2)2*1)(0+1+2) + (1+2)2*0(0+1+22*0)*22*1(0+1+2))*

Equivalências para simplificação

1. $r + s = s + r$
2. $r + \varnothing = r$
3. $r + r = r$
4. $r\lambda = \lambda r = r$
5. $r\varnothing = \varnothing r = \varnothing$
6. $(r + s)t = rt + st$
7. $r(s + t) = rs + rt$
8. $(r + s)^* = (r^*s)^*r^*$
9. $(r + s)^* = r^*(sr^*)^*$
10. $(rs)^* = \lambda + r(sr)^*s$
11. $r^{**} = r^*$
12. $r^* = (rr)^*(\lambda + r)$
13. $\varnothing^* = \lambda$
14. $\lambda^* = \lambda$
15. $r^*r^* = r^*$
16. $rr^* = r^*r$
17. $(r^* + s)^* = (r + s)^*$
18. $(r^*s^*)^* = (r + s)^*$
19. $r^*(r + s)^* = (r + s)^*$
20. $(r + s)^*r^* = (r + s)^*$

Transformação de AFs em ERs

- Quando há vários estados finais, $F = \{ f_1, f_2, \ldots, f_n \}$
  1. Para cada um dos estados finais f_k, considera-se um AF de entrada em que f_k é o único estado final
  2. Obtém-se a ER equivalente a esse AF, que será chamada de r_k
  3. Após calcular todas as ERs r_k, obtém-se a expressão regular geral fazendo

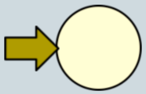
$$r = r_1 + r_2 + \ldots + r_n$$

Transformação de ERs em AFs

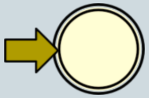
- O processo de transformação de ERs em AFs é bem mais simples do que o caminho contrário
- Para compreender o processo de transformação, serão apresentados os casos para o reconhecimento de ERs básicas

Transformação de ERs em AFs

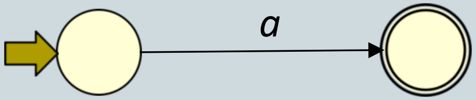
- AF para reconhecer $r = \varnothing$



- AF para reconhecer $r = \lambda$



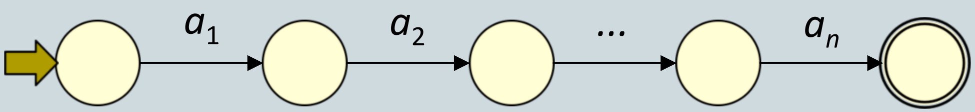
- AF para reconhecer $r = a$, sendo $a \in \Sigma$



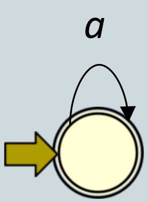
Transformação de ERs em AFs

Consequentemente,

- AF para reconhecer $r = a_1a_2 \ldots a_n$, sendo $a_k \in \Sigma$



- AF para reconhecer $r = a^*$, sendo $a \in \Sigma$



Transformação de ERs em AFs

- A partir dos casos básicos, já sabemos como compor AFs para realizar as operações permitidas por ERs
  - Concatenação, união e fecho de Kleene (utilizar transições λ)
  - Agrupamento (apenas observar as alterações de precedência das operações)

Eduardo Freire Nakamura (nakamura@icomp.ufam.edu.br)

Transformação de ERs em AFs

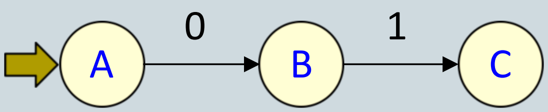
Exemplo: Transformar *r* = (01)*00+10*1* em um AF

- AF para 01
- AF para (01)*
- AF para 00
- AF para (01)*00
- AF para 1
- AF para 0*
- AF para 1*
- AF para 10*1*
- AF para (01)*00+10*1*

Transformação de ERs em AFs

Exemplo: Transformar *r* = (01)*00+10*1* em um AF

- AF para 01
- AF para (01)*
- AF para 00
- AF para (01)*00
- AF para 1
- AF para 0*
- AF para 1*
- AF para 10*1*
- AF para (01)*00+10*1*

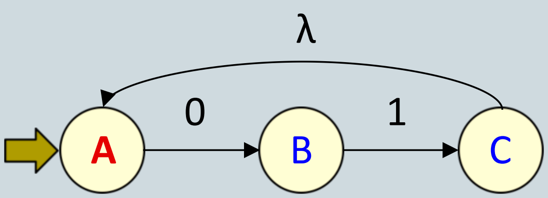


Eduardo Freire Nakamura (nakamura@icomp.ufam.edu.br)

Transformação de ERs em AFs

Exemplo: Transformar *r* = (01)*00+10*1* em um AF

- AF para 01
- AF para (01)*
- AF para 00
- AF para (01)*00
- AF para 1
- AF para 0*
- AF para 1*
- AF para 10*1*
- AF para (01)*00+10*1*

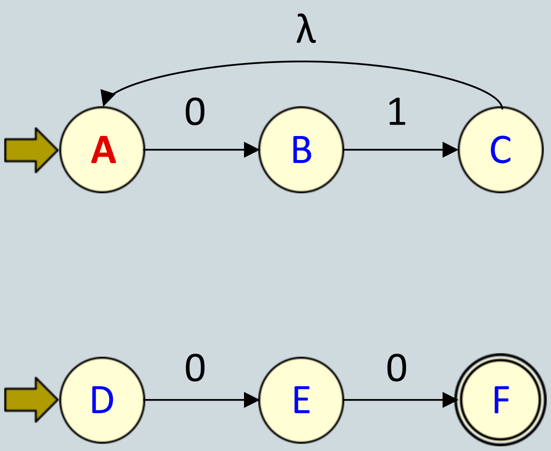


Eduardo Freire Nakamura (nakamura@icomp.ufam.edu.br)

Transformação de ERs em AFs

Exemplo: Transformar r = (01)*00+10*1* em um AF

- AF para 01
- AF para (01)*
- AF para 00
- AF para (01)*00
- AF para 1
- AF para 0*
- AF para 1*
- AF para 10*1*
- AF para (01)*00+10*1*

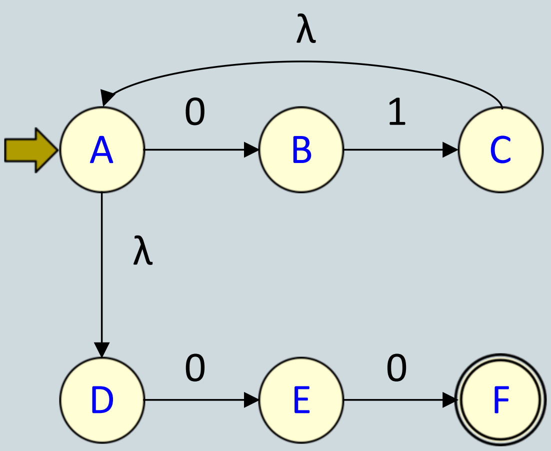


Eduardo Freire Nakamura (nakamura@icomp.ufam.edu.br)

Transformação de ERs em AFs

Exemplo: Transformar *r* = (01)*00+10*1* em um AF

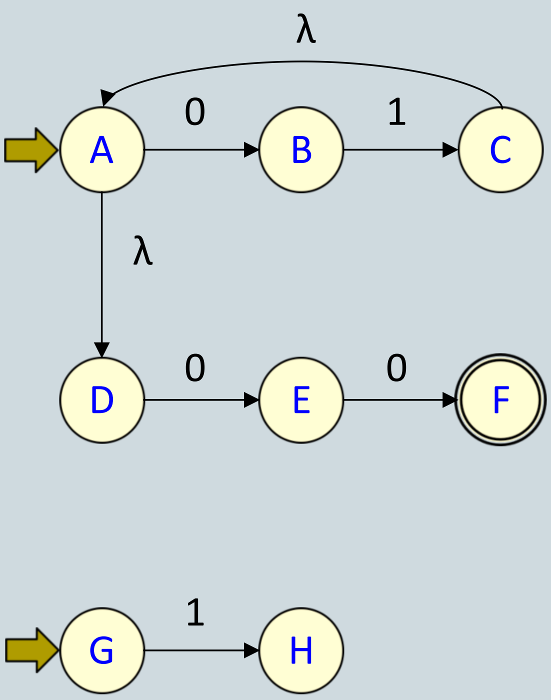
- AF para 01
- AF para (01)*
- AF para 00
- AF para (01)*00
- AF para 1
- AF para 0*
- AF para 1*
- AF para 10*1*
- AF para (01)*00+10*1*



Transformação de ERs em AFs

Exemplo: Transformar *r* = (01)*00+10*1* em um AF

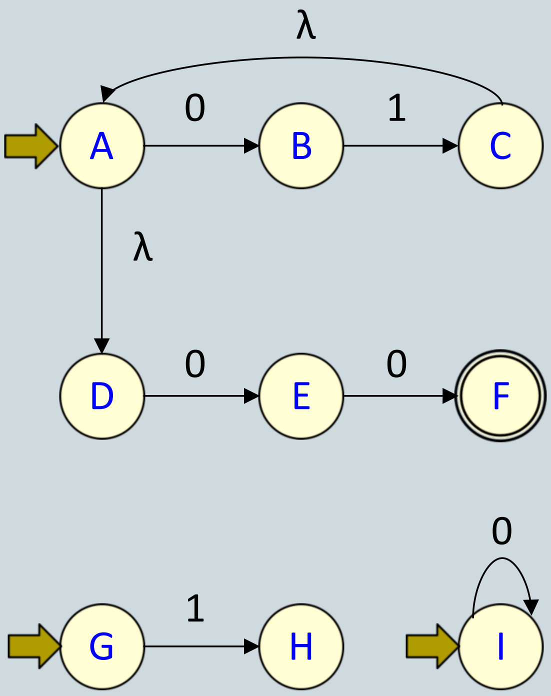
- AF para 01
- AF para (01)*
- AF para 00
- AF para (01)*00
- AF para 1
- AF para 0*
- AF para 1*
- AF para 10*1*
- AF para (01)*00+10*1*



Transformação de ERs em AFs

Exemplo: Transformar *r* = (01)*00+10*1* em um AF

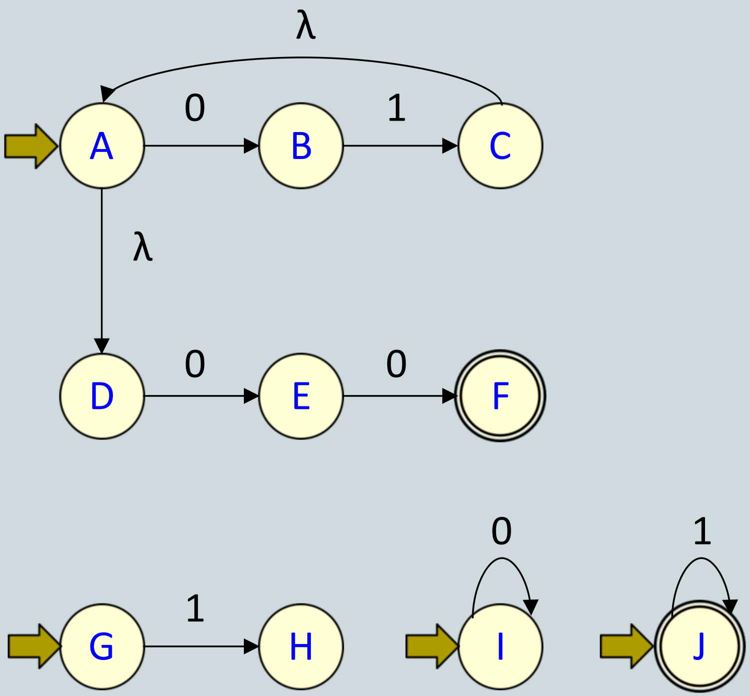
- AF para 01
- AF para (01)*
- AF para 00
- AF para (01)*00
- AF para 1
- AF para 0*
- AF para 1*
- AF para 10*1*
- AF para (01)*00+10*1*



Transformação de ERs em AFs

Exemplo: Transformar *r* = (01)*00+10*1* em um AF

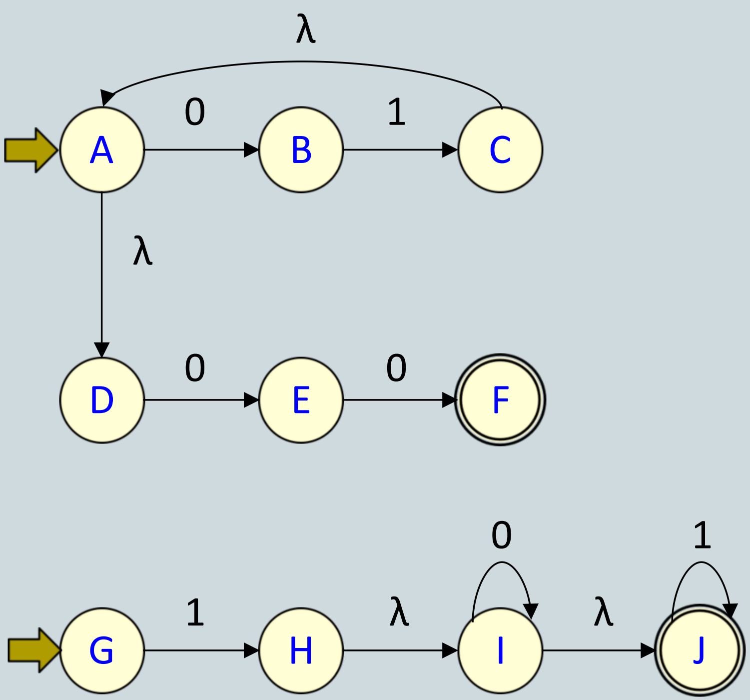
- AF para 01
- AF para (01)*
- AF para 00
- AF para (01)*00
- AF para 1
- AF para 0*
- AF para 1*
- AF para 10*1*
- AF para (01)*00+10*1*



Transformação de ERs em AFs

Exemplo: Transformar *r* = (01)*00+10*1* em um AF

- AF para 01
- AF para (01)*
- AF para 00
- AF para (01)*00
- AF para 1
- AF para 0*
- AF para 1*
- AF para 10*1*
- AF para (01)*00+10*1*



Transformação de ERs em AFs

Exemplo: Transformar *r* = (01)*00+10*1* em um AF

- AF para 01
- AF para (01)*
- AF para 00
- AF para (01)*00
- AF para 1
- AF para 0*
- AF para 1*
- AF para 10*1*
- AF para (01)*00+10*1*