

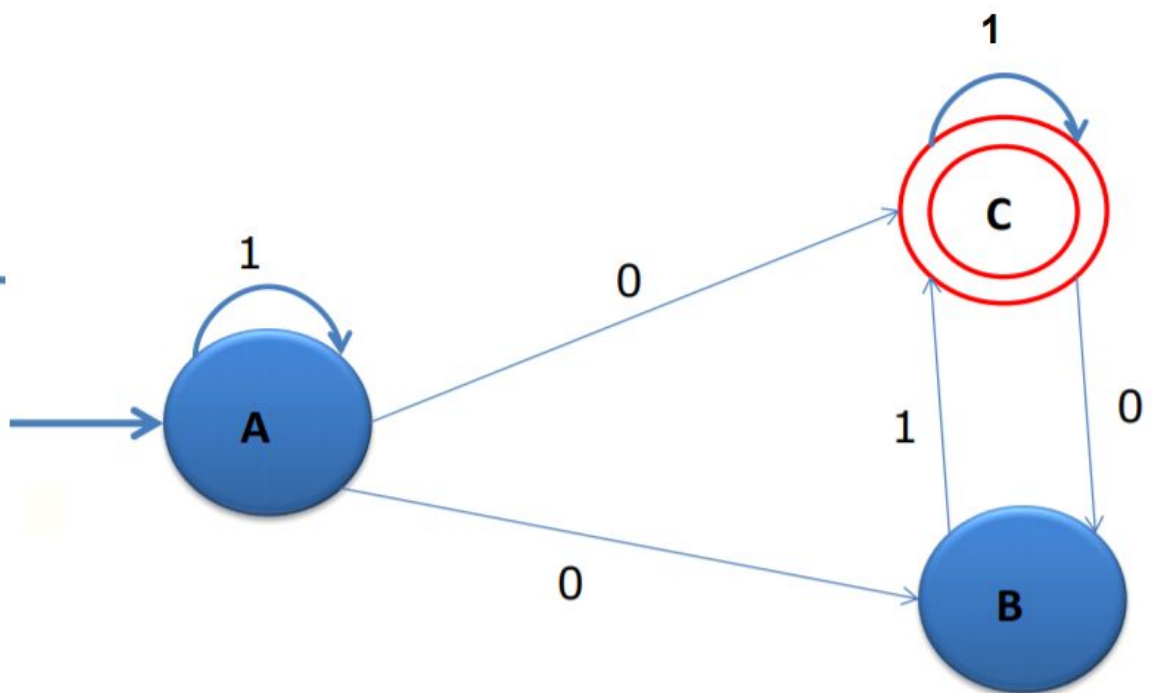
UNIVERSIDADE ESTADUAL DE PONTA GROSSA  
SETOR DE CIÊNCIAS AGRÁRIAS E DE TECNOLOGIA  
DEPARTAMENTO DE INFORMÁTICA

Exercício ERs AFNDs

PONTA GROSSA  
2019

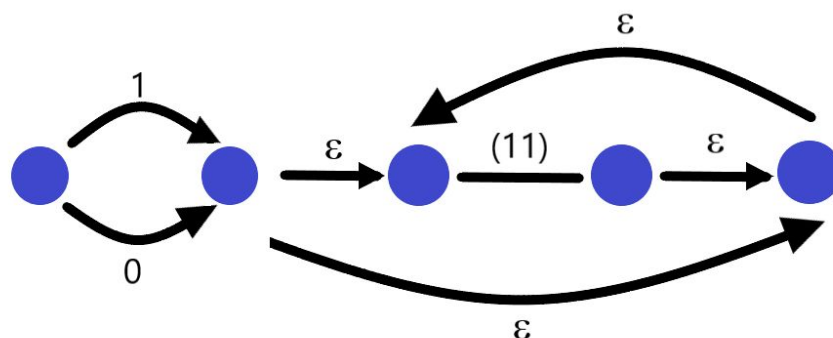
## 1. Defina AFND

São autômatos finitos em que pelo menos um símbolo (evento) de entrada possui várias transições de saídas para outros estados



Para o estado A, com o símbolo de entrada 0, leva o autômato para dois estados B ou C. Teria que se analisar independentemente o caminho para dois estados A->B ou A->C.

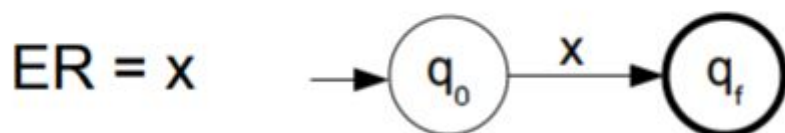
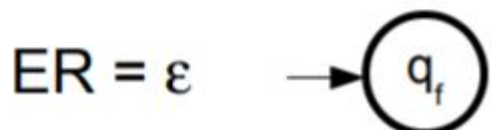
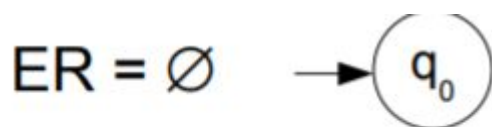
## 2. Converta a ER $(0|1)(11)^*$ em um AFND usando o algoritmo de Thompson.



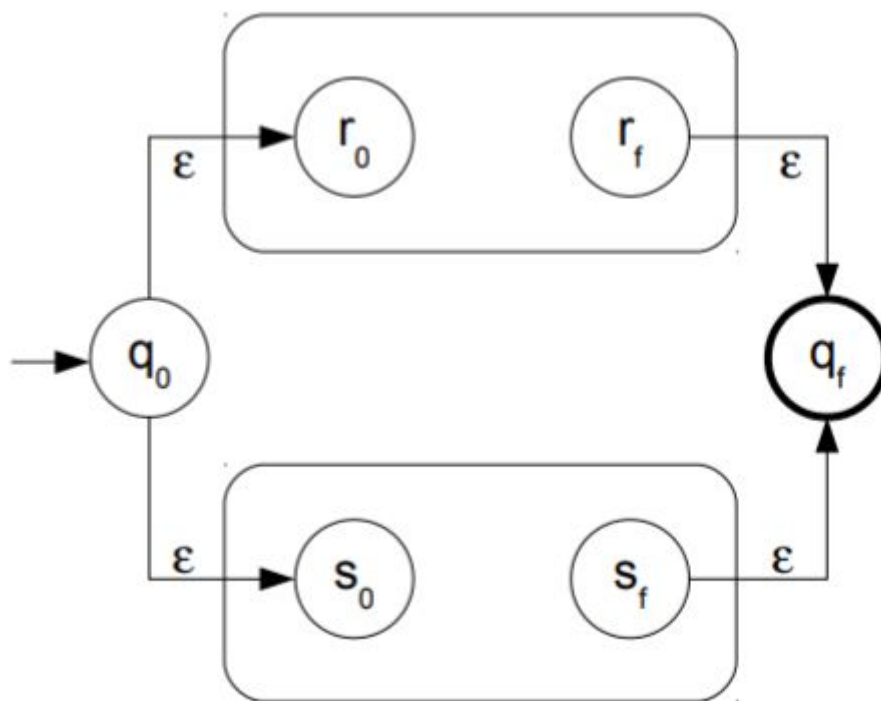
Fonte: <https://www.youtube.com/watch?v=RYNN-tb9Wxl>

3. Descreva os passos do processo de Thompson para converter ERs em AFNDs.

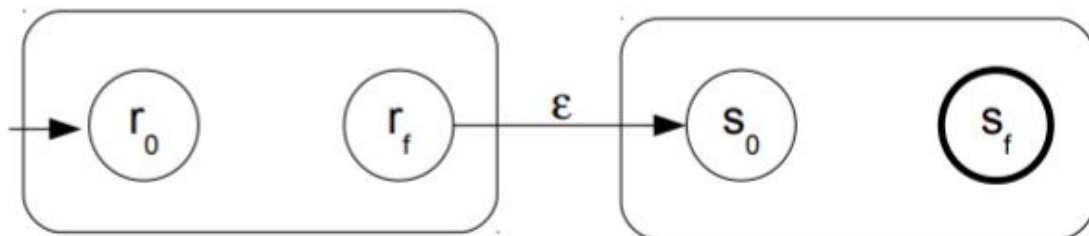
O algoritmo de Thompson define uma sequência de passos para, a partir de uma expressão regular, obter um autômato finito com movimentos vazios que reconheça sentenças a partir da correspondente linguagem regular.



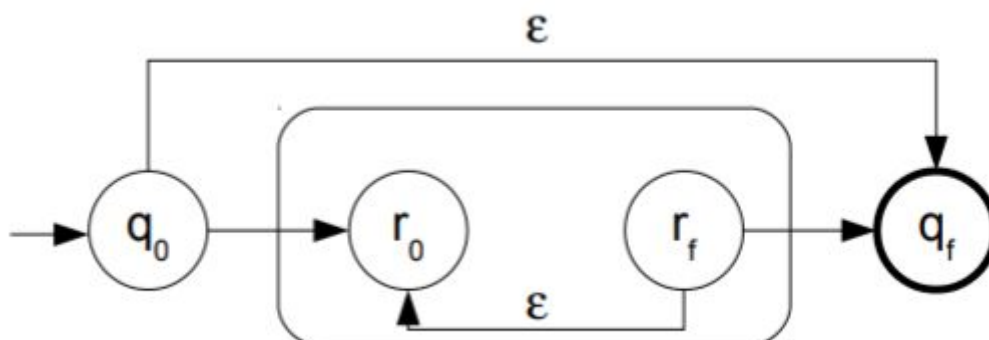
$ER = r + s$



ER = rs



ER =  $r^*$

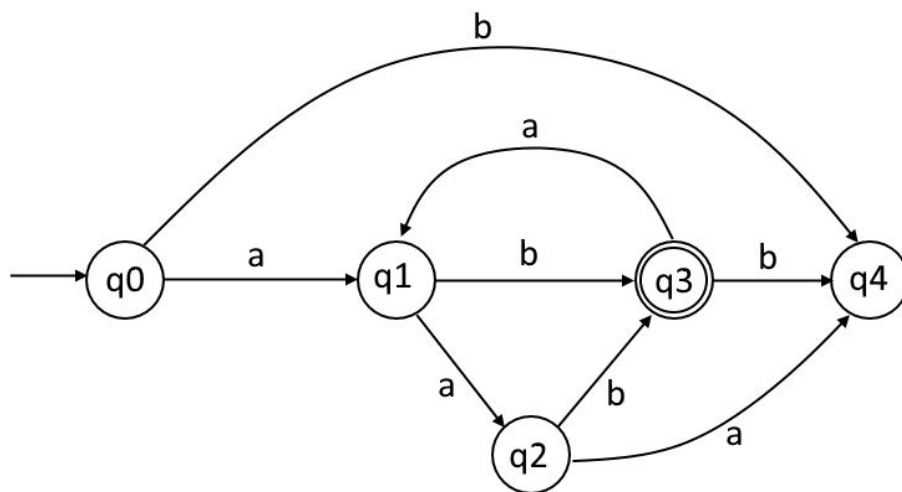


4. Elabore uma ERs para reconhecer (números inteiros/ identificadores em java/ strings em java).

5. Converta uma destas ERs em AFNDs.

1.  $(ab|aab)^*$

Essa expressão regular pode ser representada pelo seguinte autômato finito:



2.  $(0|1)^*111(0|1)^*$

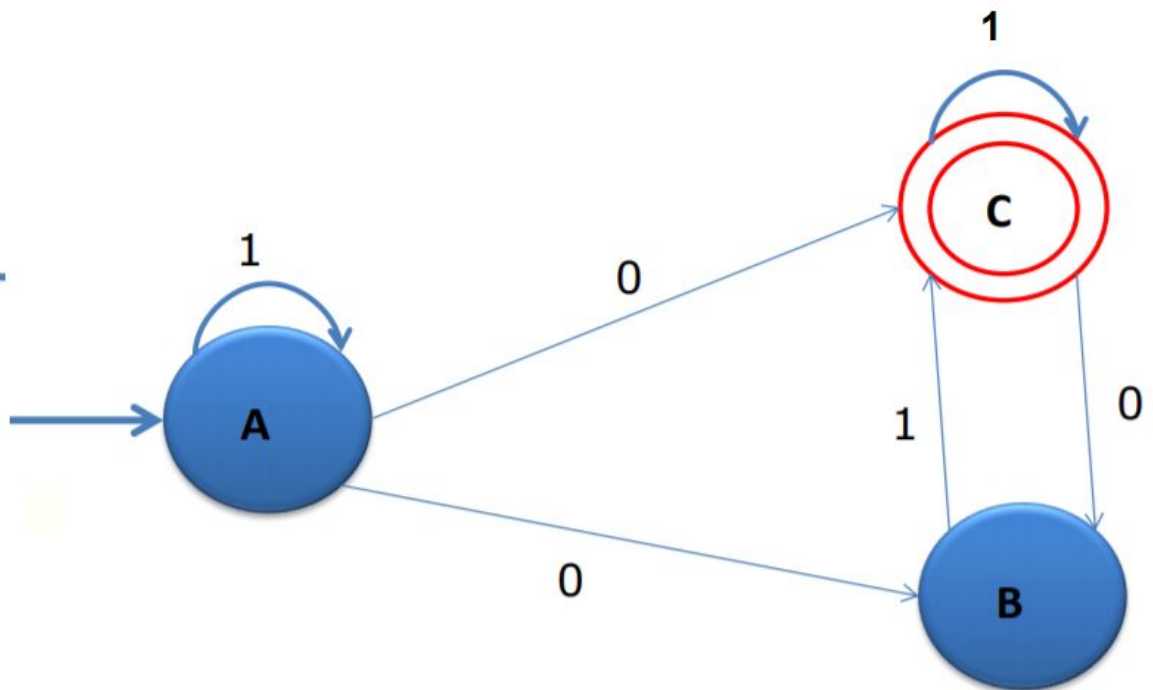
UNIVERSIDADE ESTADUAL DE PONTA GROSSA  
SETOR DE CIÊNCIAS AGRÁRIAS E DE TECNOLOGIA  
DEPARTAMENTO DE INFORMÁTICA

Exercício ERs AFNDs

PONTA GROSSA  
2019

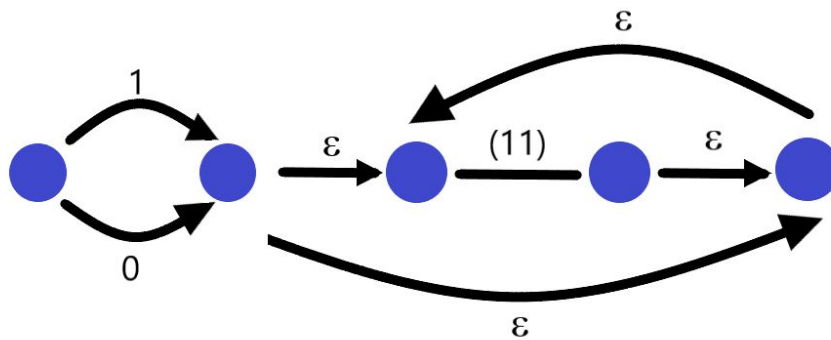
## 6. Defina AFND

São autômatos finitos em que pelo menos um símbolo (evento) de entrada possui várias transições de saídas para outros estados



Para o estado A, com o símbolo de entrada 0, leva o autômato para dois estados B ou C. Teria que se analisar independentemente o caminho para dois estados A→B ou A→C.

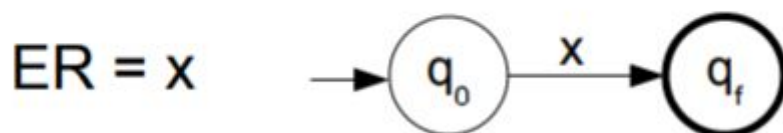
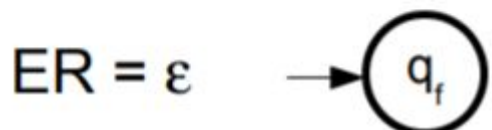
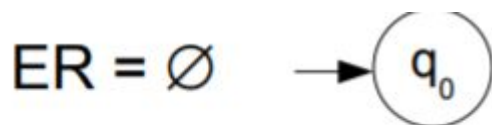
## 7. Converta a ER $(0|1)(11)^*$ em um AFND usando o algoritmo de Thompson.



Fonte: <https://www.youtube.com/watch?v=RYNN-tb9Wxl>

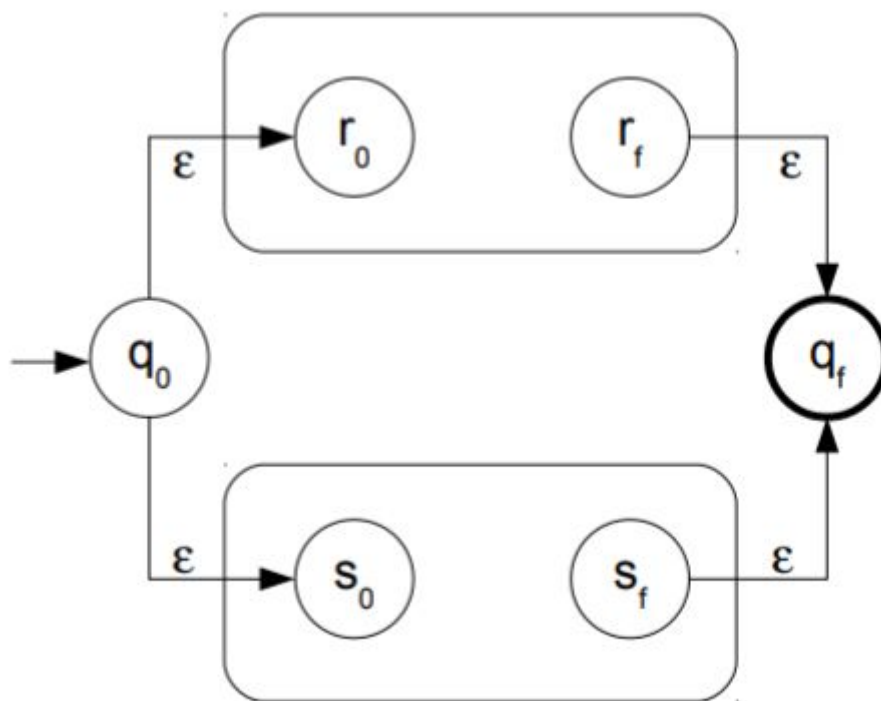
8. Descreva os passos do processo de Thompson para converter ERs em AFNDs.

O algoritmo de Thompson define uma sequência de passos para, a partir de uma expressão regular, obter um autômato finito com movimentos vazios que reconheça sentenças a partir da correspondente linguagem regular.

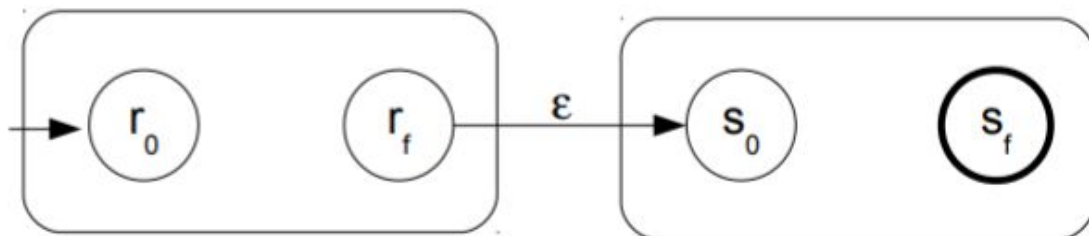


$ER = r + s$

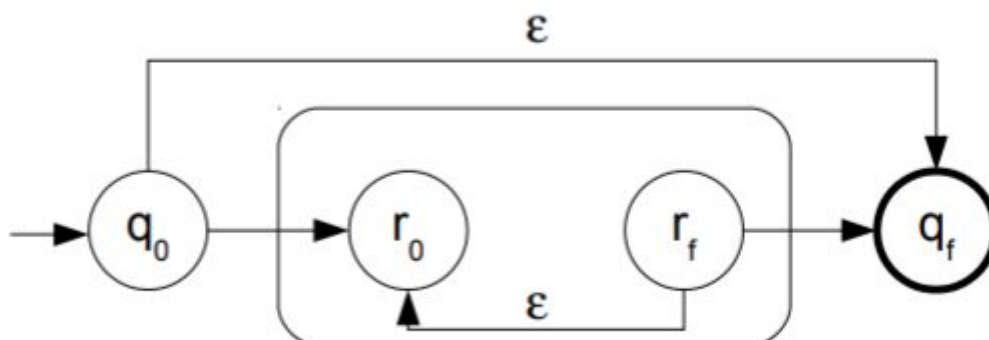




ER = rs



ER =  $r^*$

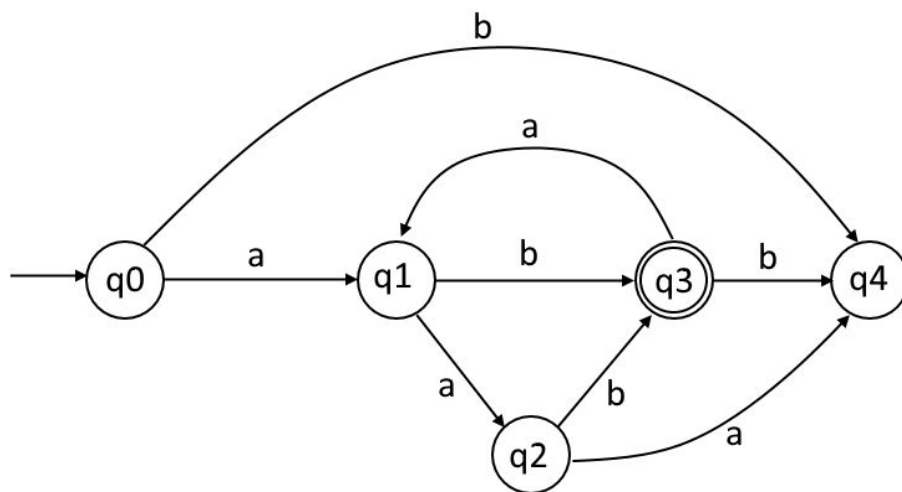


9. Elabore uma ERs para reconhecer (números inteiros/ identificadores em java/ strings em java).

10. Converta uma destas ERs em AFNDs.

1.  $(ab|aab)^*$

Essa expressão regular pode ser representada pelo seguinte autômato finito:



2.  $(0|1)^*111(0|1)^*$