**Autômatos Finitos Não determínisticos (AFND)**

**Aluno: Dericson Pablo Calari Nunes**

**André Bernardo**

**Gabriel Simioni**

Defina AFND (Autômato Finito Não-determinístico) (Revisão da última aula)

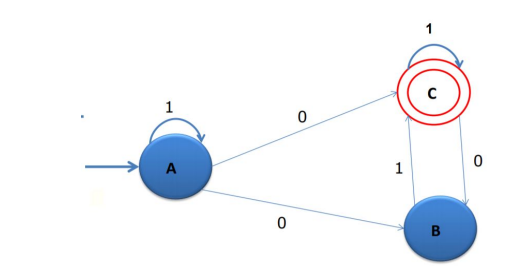
* 1. Definição formal
  2. Defina transições-ε.
  3. Quando uma cadeia é aceita por um AFND? Quando ela é rejeitada?
  4. Dê uma definição recursiva da função de transição estendida de um AFND. Qual a relação desta definição com a aceitação de uma cadeia?

1. Converta a ER (0|1)(11)\* em um AFND usando o algoritmo de Thompson.
2. Descreva os passos do processo de Thompson para converter ERs em AFNDs.
3. Elabore ERs para reconhecer números inteiros, identificadores e strings na sua linguagem preferida e converta-as em AFNDs usando o algoritmo de Thompson.
4. Converta uma destas ERs em AFNDs.
   1. (ab|aab)\*.
   2. (0|1)\*111(0|1)\*
   3. (b|c)\*(a|b)\*
   4. c\*b\*a\*

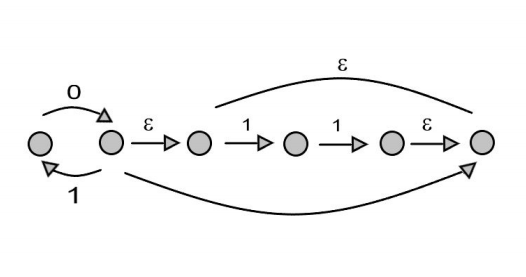
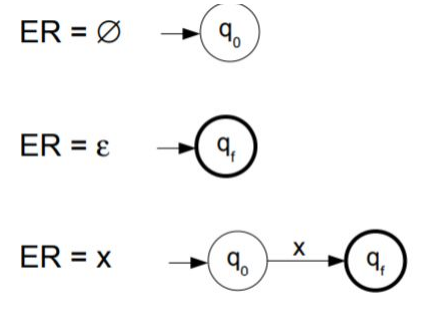
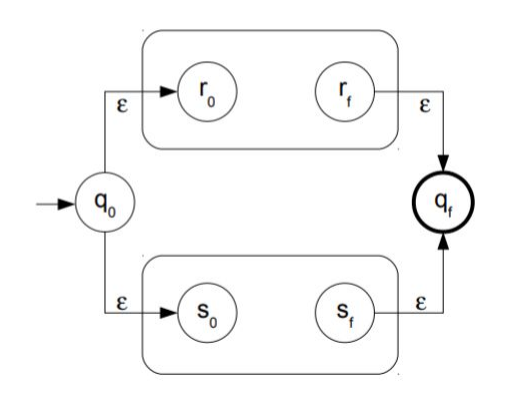
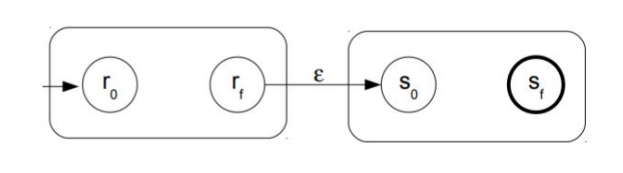
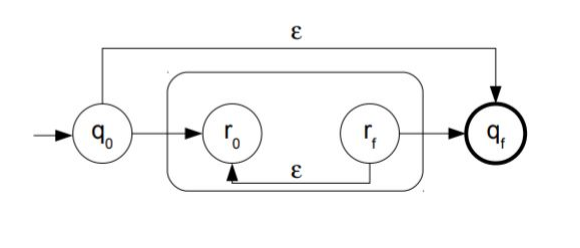
7) Descreva o procedimento para converter AFNDs em AFDs e dê um exemplo de sua aplicação.

**RESPOSTAS:**

1. 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.

1. ****
2. 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\*  
   ****

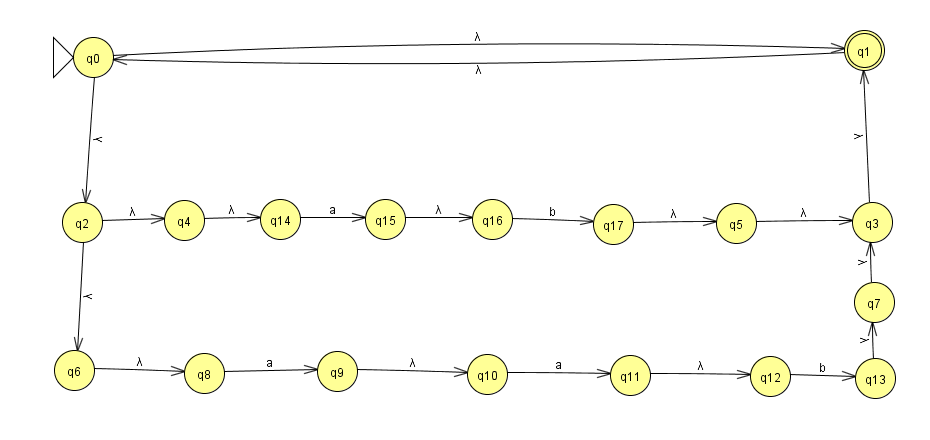
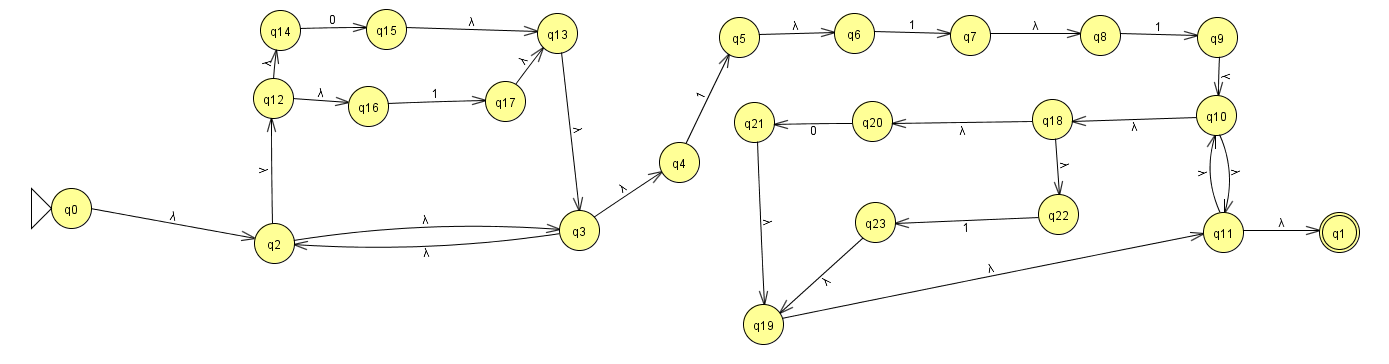
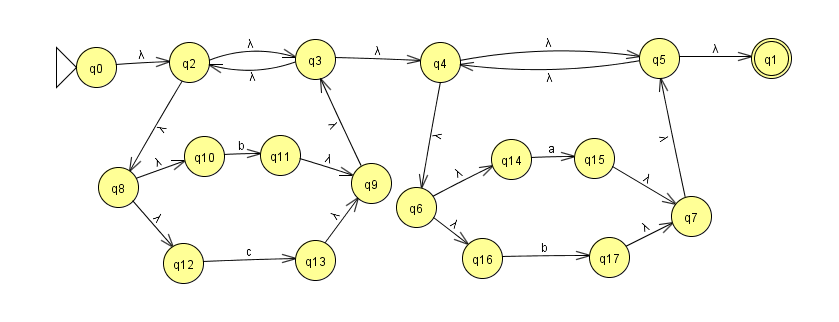
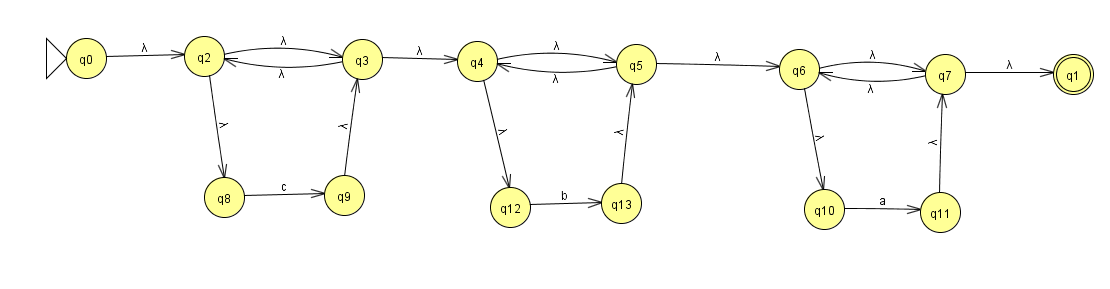
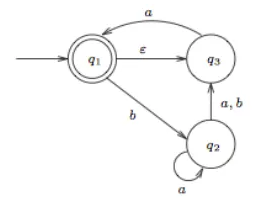
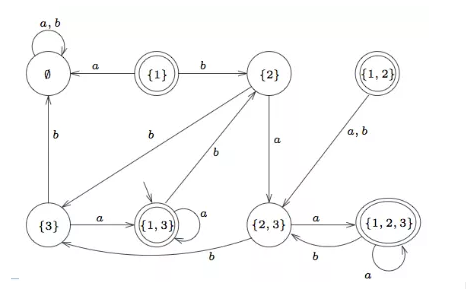
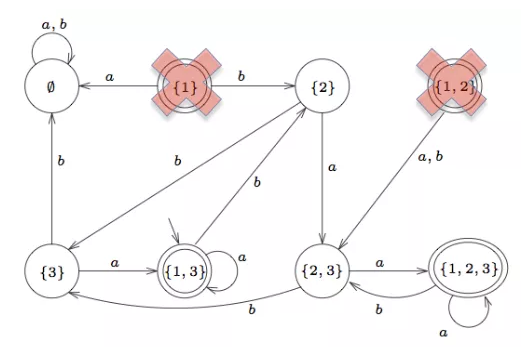
**NUMEROS INTEIROS:  
function** validarCampoNumerico(numero){

**var** er = **new** RegExp("[0-9]+");

**return** (er.test(numero));

}

**STRINGS:**  
var reg = /[a-zA-Z\u00C0-\u00FF ]+/i

1. **A)   
     
   B)  
     
   C)  
     
   D)  
   **
2. Supondo o autômato Sendo M o autômato abaixo, N tem 3 estados {1, 2, 3} e M1 terá todos os possíveis  subconjuntos de estados de M. {{}, {1}, {2}, {3}, {1, 2}, {1, 3}, {2,3}, {1, 2, 3}}  
     
   ****  
   O estado inicial é E({1}) = {1, 3} → conjunto de estados atingíveis a partir de 1 viajando ao longo de setas ε. Os novos estados de aceitação são aqueles que contem o estado de aceitação de M → {{1}, {1, 2}, {1, 3}, {1, 2, 3}}  
     
   ****  
   Podemos Simplificar essa máquina observando que nenhuma transição aponta para os estados {1} e {1, 2}.  
   ****