

## Lista Extra - Autômatos e Máquinas de Turing

- (AFD\*) Palavras com número par de "a"s: Construa um Autômato Finito Determinístico que reconheça a linguagem de todas as palavras sobre o alfabeto {a, b} que possuem um número par de "a"s.
  - Estados: {q0, q1}
  - Alfabeto: {a, b}
  - Estado inicial: q0
  - Estado final: q0
  - Transições:
    - o  $\delta(q0, a) = q1$
    - o  $\delta(q0, b) = q0$
    - $\circ$   $\delta(q1, a) = q0$
    - $\circ$   $\delta(q1, b) = q1$

**Explicação:** O autômato alterna entre dois estados. O estado q0 representa um número par de "a"s lidas até o momento, enquanto q1 representa um número ímpar.

- 2. (AFND\*) Palavras que começam e terminam com o mesmo símbolo: Construa um Autômato Finito Não-Determinístico que reconheça a linguagem de todas as palavras sobre o alfabeto {a, b} que começam e terminam com o mesmo símbolo.
  - Estados: {q0, q1, q2, q3}
  - Alfabeto: {a, b}
  - Estado inicial: q0
  - Estados finais: {q1, q3}
  - Transições:

$$\circ$$
  $\delta(q0, a) = \{q1\}$ 

$$\circ$$
  $\delta(q0, b) = \{q3\}$ 

$$\circ$$
  $\delta(q1, a) = \{q1\}$ 

$$\circ$$
  $\delta(q1, b) = \{q1\}$ 

$$\circ$$
  $\delta(q3, a) = \{q3\}$ 

$$o \delta(q3, b) = \{q3\}$$



**Explicação:** O autômato utiliza transições não-determinísticas para "adivinhar" qual será o símbolo final da palavra.

**3. (AFD) Múltiplos de 3:** Construa um Autômato Finito Determinístico que reconheça a linguagem de todas as palavras sobre o alfabeto {0, 1} que representam números binários múltiplos de 3.

• Estados: {q0, q1, q2}

• Alfabeto: {0, 1}

• Estado inicial: q0

• Estado final: q0

Transições:

$$\circ \quad \delta(q0, 0) = q0$$

$$\circ$$
  $\delta(q0, 1) = q1$ 

$$\circ$$
  $\delta(q1, 0) = q2$ 

$$\circ \quad \delta(q1,\,1)=q0$$

$$\circ$$
  $\delta(q2, 0) = q1$ 

$$\circ$$
  $\delta(q2, 1) = q2$ 

**Explicação:** O autômato utiliza o conceito de resto da divisão por 3. Cada estado representa um resto possível (0, 1 ou 2).

**4. (AFND\*) Substring "ab":** Construa um Autômato Finito Não-Determinístico que reconheça a linguagem de todas as palavras sobre o alfabeto {a, b} que contêm a substring "ab".

• Estados: {q0, q1, q2}

• Alfabeto: {a, b}

Estado inicial: q0

Estado final: q2

• Transições:

$$\circ$$
  $\delta(q0, a) = \{q0, q1\}$ 

$$\delta(q0, b) = \{q0\}$$

$$\circ$$
  $\delta(q1, b) = \{q2\}$ 

$$\circ$$
  $\delta(q2, a) = \{q2\}$ 



$$o \delta(q2, b) = \{q2\}$$

**Explicação:** O autômato permanece no estado q0 enquanto não encontra um 'a' seguido de um 'b'.

- **5. (AFND-AFD\*) Conversão AFN-AFD:** Converta o seguinte Autômato Finito Não-Determinístico para um Autômato Finito Determinístico equivalente:
- Estados: {q0, q1, q2}
- Alfabeto: {a, b}
- Estado inicial: q0
- Estados finais: {q2}
- Transições:
  - $\circ$   $\delta(q0, a) = \{q0, q1\}$
  - $o \delta(q0, b) = \{q0\}$
  - $\circ$   $\delta(q1, a) = \{q2\}$
  - $\circ$   $\delta(q1, b) = {}$
  - $\circ$   $\delta(q2, a) = {}$
  - $\circ \quad \delta(q2, b) = \{\}$

## Resposta:

- Estados: {{q0}, {q0, q1}, {q0, q2}, {q0, q1, q2}}
- Alfabeto: {a, b}
- Estado inicial: {q0}
- Estados finais: {{q0, q2}, {q0, q1, q2}}
- Transições:
  - $\circ$   $\delta(\{q0\}, a) = \{q0, q1\}$
  - $\circ$   $\delta(\{q0\}, b) = \{q0\}$
  - $\circ$   $\delta(\{q0, q1\}, a) = \{q0, q1, q2\}$
  - $\circ$   $\delta(\{q0, q1\}, b) = \{q0\}$
  - $\circ$   $\delta(\{q0, q2\}, a) = \{q0, q1\}$



- $\circ$   $\delta(\{q0, q2\}, b) = \{q0\}$
- $\circ$   $\delta(\{q0, q1, q2\}, a) = \{q0, q1, q2\}$
- $\circ$   $\delta(\{q0, q1, q2\}, b) = \{q0\}$

**Explicação:** A conversão AFN-AFD gera um AFD onde cada estado representa um conjunto de estados do AFN original. As transições são definidas com base nas transições do AFN original, considerando todos os estados presentes no conjunto.

- **6. (AP\*) Palíndromos:** Construa um Autômato de Pilha que reconheça a linguagem de todos os palíndromos sobre o alfabeto {a, b}. Um palíndromo é uma palavra que pode ser lida da mesma forma da esquerda para a direita e da direita para a esquerda (ex: "arara", "ovo").
  - Estados: {q0, q1}
  - Alfabeto de entrada: {a, b}
  - Alfabeto da pilha: {a, b, Z}
  - Símbolo inicial da pilha: Z
  - Estado inicial: q0
  - Estado final: q1
  - Transições:
    - o  $\delta(q0, a, Z) = (q0, aZ) // Empilha 'a'$
    - o  $\delta(q0, b, Z) = (q0, bZ) // Empilha 'b'$
    - o  $\delta(q0, a, a) = (q0, aa) // Empilha 'a'$
    - o  $\delta(q0, b, b) = (q0, bb) // Empilha 'b'$
    - $\circ$  δ(q0, ε, Z) = (q1, Z) // Transição para estado final se a pilha só tiver Z
    - o  $\delta(q0, a, a) = (q1, \epsilon) // Desempilha 'a' se corresponder ao topo$
    - o  $\delta(q0, b, b) = (q1, \epsilon) // Desempilha 'b' se corresponder ao topo$

**Explicação:** O autômato empilha os símbolos lidos até encontrar o meio da palavra. Em seguida, ele começa a desempilhar, verificando se os símbolos correspondem à segunda metade da palavra.

7. (AP\*) Balanceamento de Parênteses: Construa um Autômato de Pilha que reconheça a linguagem de todas as expressões com parênteses balanceados, ou seja, cada "(" possui um ")" correspondente e vice-versa. Considere apenas os símbolos "(" e ")".



• Estados: {q0, q1}

• Alfabeto de entrada: {(, )}

• Alfabeto da pilha: {(, Z}

Símbolo inicial da pilha: Z

• Estado inicial: q0

Estado final: q1

• Transições:

 $\circ$   $\delta(q0, (, Z) = (q0, (Z) // Empilha '('$ 

 $\circ$   $\delta(q0, (, () = (q0, (() // Empilha '('$ 

o  $\delta(q0, ), () = (q0, \epsilon) // Desempilha '(')$ 

 $\circ$   $\delta(q0, \epsilon, Z) = (q1, Z) // Transição para estado final se a pilha só tiver Z$ 

**Explicação:** O autômato empilha os parênteses abertos '(' e os desempilha quando encontra um parêntese fechado ')'. A pilha vazia no final indica que todos os parênteses foram balanceados.

**8. (MT\*) Cópia de uma String:** Construa uma Máquina de Turing que, dada uma fita contendo uma string "w" seguida por um branco, duplique a string, resultando em "ww" na fita.

Estados: {q0, q1, q2, q3, q4}

Alfabeto de entrada: {a, b, \_}

Alfabeto da fita: {a, b, , X, Y}

Estado inicial: q0

Estado final: q4

Transições:

o  $\delta(q0, a) = (q1, X, R)$  // Marca o primeiro 'a' com X e move para a direita

o  $\delta(q0, b) = (q1, Y, R)$  // Marca o primeiro 'b' com Y e move para a direita

o  $\delta(q1, a) = (q1, a, R) // Move para a direita até encontrar o branco$ 

o  $\delta(q1, b) = (q1, b, R) // Move para a direita até encontrar o branco$ 

o  $\delta(q1, \underline{\ }) = (q2, \underline{\ }, \underline{\ }) // Encontrou o branco, volta para a esquerda$ 

o  $\delta(q^2, a) = (q^2, a, L)$  // Move para a esquerda procurando X ou Y



- o  $\delta(q2, b) = (q2, b, L) // Move para a esquerda procurando X ou Y$
- o  $\delta(q2, X) = (q0, X, R) // Achou X, volta para o início para copiar o próximo símbolo$
- o  $\delta(q2, Y) = (q0, Y, R) // Achou Y, volta para o início para copiar o próximo símbolo$
- o  $\delta(q0, _) = (q3, _, R)$  // Se encontrar branco no início, significa que copiou tudo
- o  $\delta(q3, a) = (q3, a, R)$  // Move para a direita até o final da primeira cópia
- o  $\delta(q3, b) = (q3, b, R) // Move para a direita até o final da primeira cópia$
- $\circ$   $\delta(q3, _) = (q4, _, L) // Chegou ao final, move para a esquerda e termina$

**Explicação:** A máquina percorre a fita marcando cada símbolo da string original e copiando-o para o final da fita.

- 9. (MT\*) Adição de números unários: Construa uma Máquina de Turing que receba como entrada dois números unários separados por um branco (ex: "111 11" representa 3 + 2) e calcule a sua soma, deixando o resultado na fita (ex: "11111", representando 5).
  - Estados: {q0, q1, q2, q3}
  - Alfabeto de entrada: {1, \_}
  - Alfabeto da fita: {1, \_}
  - Estado inicial: q0
  - · Estado final: q3
  - Transições:
    - o  $\delta(q0, 1) = (q0, 1, R)$  // Move para a direita até encontrar o branco
    - o  $\delta(q0, \underline{\ }) = (q1, 1, R) // Encontrou o branco, substitui por 1 (somando 1)$
    - o  $\delta(q1, 1) = (q1, 1, R)$  // Move para a direita até o final do segundo número
    - o  $\delta(q1, ) = (q2, , L) // Encontrou o final, volta para apagar o último 1$
    - o  $\delta(q2, 1) = (q3, \_, L) // Apaga o último 1 e termina$

**Explicação:** A máquina percorre a fita até o final do primeiro número, substitui o branco separador por '1' (somando 1 ao total), percorre o segundo número e apaga o último '1', deixando o resultado na fita.