

Universidade Federal de Alagoas
Instituto de Computação

Monitoria de Linguagens Formais, Autômatos e Computabilidade
Professor Leandro Dias

Luana Júlia Nunes Ferreira, Mateus Fernando Felismino da Silva
Patriota

Sumário

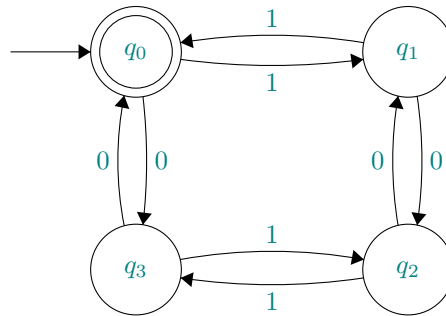
1	Linguagens Regulares	3
1.1	Autômato Finito Determinístico	3
1.2	Autômato Finito Não Determinístico	5
1.3	Expressão Regular	7
1.4	Gramática Regular	8
1.5	Autômato Finito com Saída	9
1.5.1	Máquina de Mealy	9
1.5.2	Máquina de Moore	11
1.6	Conversão de AFND para AFD	12
2	Linguagens Livres de Contexto	12
2.1	Autômato com Pilha	12
2.2	Árvore de Derivação	13
3	Linguagens Enumeradas e Recursivamente e Sensíveis ao Contexto	13
3.1	Máquina de Turing	13
3.2	Hipótese de Church	13
4	Computabilidade	14
4.1	Cálculo Lambda	14
5	Atividades 2020.1	14
5.1	Atividade 1	14
5.2	Atividade 2	14
5.3	Atividade 3	15
5.4	Atividade 4	17
5.5	Atividade 5	18
5.6	Atividade 6	19
5.7	Atividade 7	19
5.8	Atividade 8	20
6	Atividades 2021.2	20
6.1	Atividade 4	20
7	Avaliações Anteriores	21
7.1	Avaliação 01 - 2020.1	21
7.2	Avaliação 02 - 2020.1	27

1 Linguagens Regulares

1.1 Autômato Finito Determinístico

Questão 1. Construa um AFD tal que $L(M) = \{w \mid w \in \{0,1\}^* \text{ e possui um número par de ocorrências de } 0\text{'s e de } 1\text{'s}\}$.

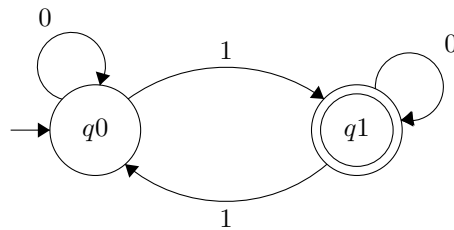
Solução 1. Uma representação possível é:



Questão 2. Construa um AFD que aceite cadeias sequenciais de "abc" em ocorrências pares.

Solução 2. O autômato pode ser representado por: todo

Questão 3. Para o autômato dado: Dê a definição formal; construa a tabela da função programa; mostre a computação da palavra 00100; que autômato é esse? Que linguagem ele reconhece?



Solução 3. Definição formal: $M = (\Sigma, Q, \delta, q_0, F)$, de modo que:

- $\Sigma = \{0, 1\}$
- $Q = \{q_0, q_1\}$
- $\delta = \{\delta(q_0, 0) = q_0; \delta(q_0, 1) = q_1; \delta(q_1, 0) = q_0; \delta(q_1, 1) = q_0\}$
- $q_0 = q_0$
- $F = \{q_1\}$

Função computação:

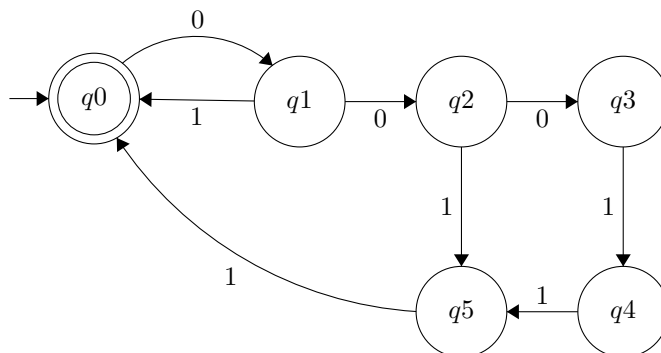
δ	0	1
q_0	q_0	q_1
q_1	q_1	q_0

Computação da palavra $w = 00100$:

$$\left\{ \begin{array}{lll} \delta^*(q_0, 00100) & = \delta^*(\delta(q_0, 0), 0100) & = \delta^*(q_0, 0100) \\ = \delta^*(\delta(q_0, 0), 100) & = \delta^*(q_0, 100) & = \delta^*(\delta(q_0, 1), 00) \\ = \delta^*(q_1, 00) & = \delta^*(\delta(q_1, 0), 0) & = \delta^*(q_1, 0) \\ = \delta^*(\delta(q_1, 0), \epsilon) & = \delta^*(q_1, \epsilon) & = q_1 \end{array} \right.$$

Esse autômato é um AFD (Autômato Finito Determinístico) e reconhece as linguagens sobre o alfabeto $\{0, 1\}$ que contenham um número ímpar de 1's.

Questão 4. Para o automato dado: Dê a definição formal; construa a tabela da função programa; mostre a computação da palavra 00100; que autômato é esse? Que linguagem ele reconhece?



Solução 4. A definição formal é $M = \{\Sigma, Q, \delta, q_0, F\}$, onde:

- $\Sigma = \{0, 1\}$
- $Q = \{q_0, q_1, q_2, q_3, q_4, q_5\}$
- $\delta = \{\delta(q_0, 0) \rightarrow q_1, \delta(q_1, 0) \rightarrow q_2, \delta(q_2, 0) \rightarrow q_3, \delta(q_3, 0) \rightarrow \emptyset, \delta(q_4, 0) \rightarrow \emptyset, \delta(q_5, 0) \rightarrow \emptyset, \delta(q_0, 1) \rightarrow q_1, \delta(q_1, 1) \rightarrow q_0, \delta(q_2, 1) \rightarrow q_5, \delta(q_3, 1) \rightarrow q_4, \delta(q_4, 1) \rightarrow q_5, \delta(q_5, 1) \rightarrow q_0\}$
- $q_0 = q_0$
- $F = \{q_0\}$

Tabela com a função programa:

δ	0	1
q0	q1	
q1	q2	
q2	q3	q5
q3		q4
q4		q5
q5		q0

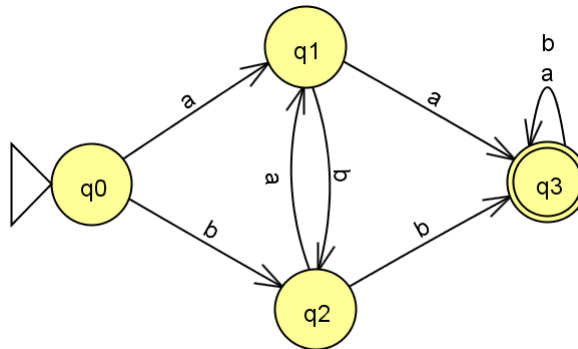
Computação da palavra $w = 00100$:

$$\left\{ \begin{array}{lll} \delta^*(q_0, 00100) & = \delta^*(\delta(q_0, 0), 0100) & = \delta^*(q_1, 0100) \\ = \delta^*(\delta(q_1, 0), 100) & = \delta^*(q_2, 100) & = \delta^*(\delta(q_2, 1), 00) \\ = \delta^*(q_5, 00) & = \delta^*(\delta(q_5, 0), 0) & = \emptyset \end{array} \right.$$

Esse autômato é um AFD (Autômato Finito Determinístico) e reconhece as linguagens sobre o alfabeto $\{0, 1\}$ que contenham o mesmo número de 1's precedidos de 0's.

Questão 5. Construa um autômato que aceite aa ou bb como subpalavra. Faça a função programa.

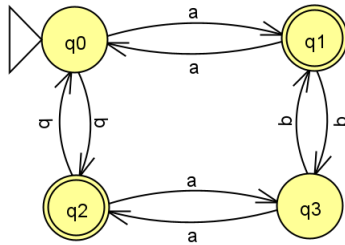
Solução 5. O autômato pode ser representado por:



1.2 Autômato Finito Não Determinístico

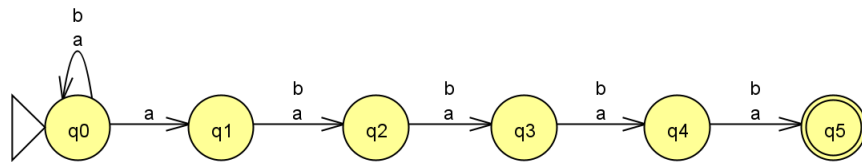
Questão 6. Construa um autômato que $L(M) = \{ w \mid w \text{ possui número par de 'a' e ímpar de 'b' ou par de 'b' e ímpar de 'a' } \}$

Solução 6. O autômato pode ser representado por:

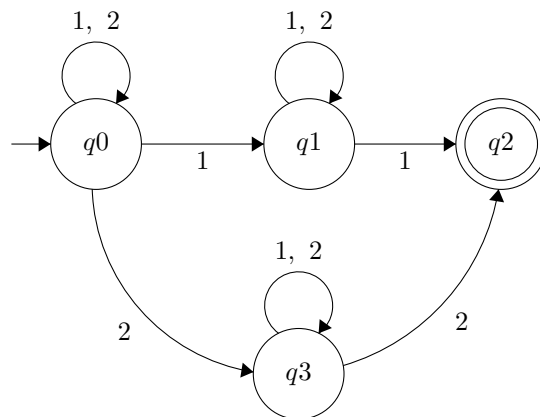


Questão 7. Construa um autômato que $L(M) = \{ w \mid w \text{ o quinto símbolo da direita para a esquerda de } w \text{ é 'a'} \}$. Compute a palavra 'abaaab'. Ela é aceita pelo autômato?

Solução 7. O autômato pode ser representado por:



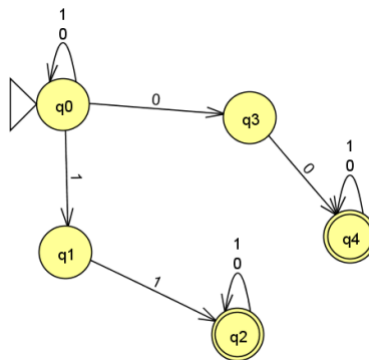
Questão 8. Para o automato dado: Dê a definição formal; construa a tabela da função programa; mostre a computação da palavra 00100; que autômato é esse? Que linguagem ele reconhece?



Solução 8. Aceita cadeias $\in \{1,2\}^*$ tal que o último símbolo na cadeia tenha aparecido anteriormente

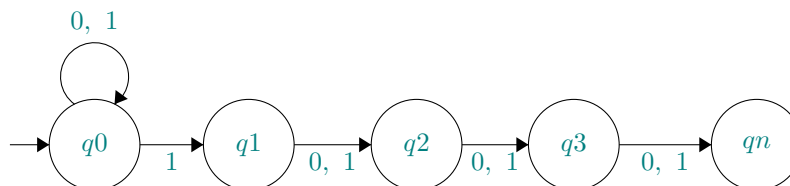
Questão 9. Construa um AFND tal que $L(M) = \{w \mid w \in \{0,1\}^* \text{ e tenha dois } 0\text{'s consecutivos ou dois } 1\text{'s consecutivos}\}$

Solução 9. O autômato pode ser representado por:



Questão 10. Considerando o alfabeto $\Sigma = \{a, b\}$, construa um autômato finito não determinístico que admite cadeias em que o termo n -ésimo, a partir da direita, é b .

Solução 10. O autômato pode ser representado por:



1.3 Expressão Regular

Questão 11. Considerando o alfabeto $\Sigma = \{a, b\}$, escreva uma expressão regular que represente a linguagem:

- (a) que começa com b e termina com uma quantidade indefinida de a , incluindo a palavra vazia;
- (b) que começa com a , termina com b e pode conter uma infinidade de letras, exceto a palavra vazia;
- (c) que tem sempre um número par de letras a ;

(d) que possui aa como subpalavra;

(e) que possui bbb como sufixo;

Solução 11.

Questão 12. Considerando o alfabeto $\Sigma = \{a, b\}$, descreva que tipo de linguagem cada expressão regular representa.

(a) ab^*

(b) $(a + b)^*$

(c) $(a + b)^*aa(a + b)^*$

(d) $(a + \varepsilon)(b + ba)^*$

(e) $(a + b)^*(aa + bb)$

Solução 12.

1.4 Gramática Regular

Questão 13. Dê a definição de Gramática Regular. Com isso, classifique e exemplifique os tipos de gramática regular.

Solução 13.

Questão 14. A partir da expressão regular $w = a(ba)^*$, determine as regras de produção de sua respectiva gramática linear à direita. Em seguida, exemplifique para a palavra $w = ababa$.

Solução 14.

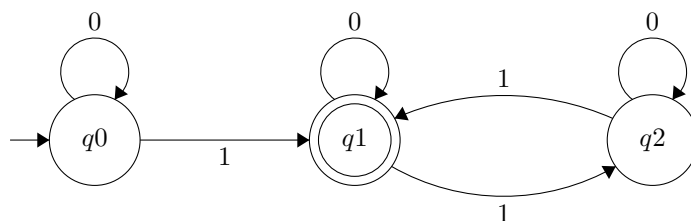
Questão 15. A partir da expressão regular $w = 01^*0^*$, determine as regras de produção de sua respectiva gramática regular. Em seguida, exemplifique para a palavra $w = 011100$.

Solução 15.

Questão 16. A partir da expressão regular $w = (x + y)^*$, determine as regras de produção de sua respectiva gramática regular. Em seguida, exemplifique para a palavra $w = xyx$.

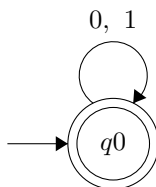
Solução 16.

Questão 17. Considerando o alfabeto $\Sigma = \{0, 1\}$, construa a gramática regular do seguinte autômato:



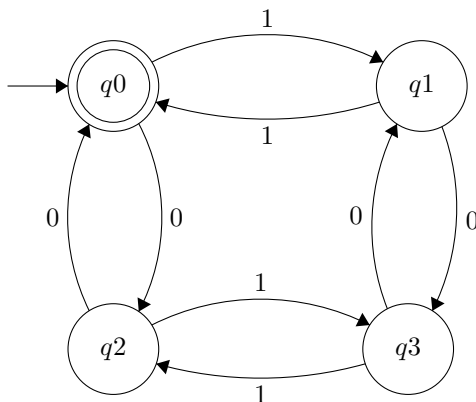
Solução 17.

Questão 18. Considerando o alfabeto $\Sigma = \{0, 1\}$, construa a gramática regular do seguinte autômato:



Solução 18.

Questão 19. Considerando o alfabeto $\Sigma = \{0, 1\}$, construa a gramática regular do seguinte autômato:



Solução 19.

1.5 Autômato Finito com Saída

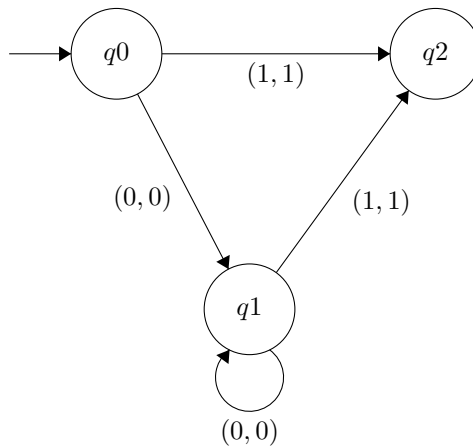
1.5.1 Máquina de Mealy

Questão 20. Dê a definição matemática de Máquina de Mealy.

Solução 20. A máquina de Mealy pode ser definida como uma Tupla dada por:
 $M = (\Sigma, Q, \delta, q_0, F, \Delta)$

- Σ é o alfabeto de símbolos de entrada.
- Q representa o conjunto de estados.
- δ é a função programa ou de transição.
- q_0 é o estado inicial.
- F é o conjunto de estados finais.
- Δ É o alfabeto de símbolos de saída.

Questão 21. Escreva a definição para a Máquina de Mealy a seguir e calcule a fita de saída para as fitas de entrada abaixo, caso a palavra seja aceita. Por fim, responda: que tipo de tarefa esse autômato desempenha? Dica: lógica booleana.



- (a) 01000
- (b) 0000
- (c) 1110011
- (d) 11

Solução 21.

Questão 22. Transforme a Máquina de Mealy da questão anterior na sua Máquina de Moore equivalente.

Solução 22.

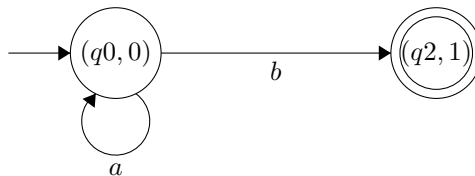
1.5.2 Máquina de Moore

Questão 23. Dê a definição matemática de Máquina de Moore.

Solução 23. A máquina de Moore pode ser definida como uma Tupla dada por:
 $M = (\Sigma, Q, \delta, q_0, F, \Delta, \delta_S)$

- Σ é o alfabeto de símbolos de entrada.
- Q representa o conjunto de estados.
- δ é a função programa ou de transição.
- q_0 é o estado inicial.
- F é o conjunto de estados finais.
- Δ é o alfabeto de símbolos de saída.
- δ_S é a função de saída.

Questão 24. Escreva a definição para a Máquina de Moore a seguir e calcule a fita de saída para as fitas de entrada abaixo, caso a palavra seja aceita. Por fim, responda: que tipo de tarefa esse autômato desempenha?

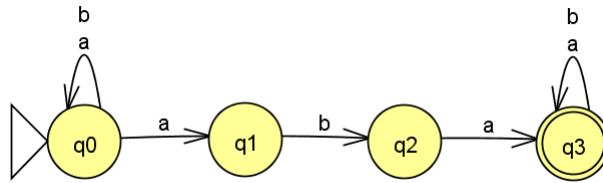
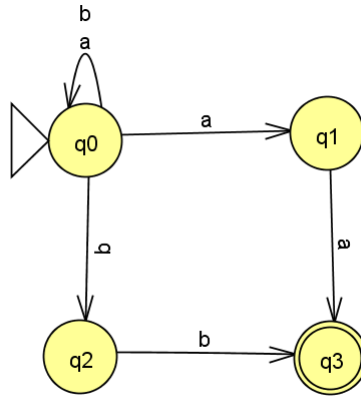


- (a) aaab
- (b) abab
- (c) aa

Solução 24.

Questão 25. Transforme a Máquina de Moore da questão anterior na sua Máquina de Mealy equivalente.

Solução 25.



1.6 Conversão de AFND para AFD

Questão 26. O que este autômato faz? Compute $w = 'baaa'$ e converta para AFD.

Solução 26.

Questão 27. O que este autômato faz? Compute $w = 'baba'$ e converta para AFD.

Solução 27.

2 Linguagens Livres de Contexto

2.1 Autômato com Pilha

Questão 28. Construa um autômato tal que $L(M) = \{wcw^r \mid w \in \{0,1\}^*\}$. Na prática, o que esse autômato reconhece?

Solução 28.

Questão 29. Construa um autômato tal que $L(M) = \{a^n b^m b^c \mid c = n + m \mid n \geq 0, m \geq 0 \in \{a,b\}^*\}$. Exemplo de palavra reconhecida $a^1 b^1 b^2 = abbb$.

Solução 29.

2.2 Árvore de Derivação

Questão 30. Construa a árvore de derivação da palavra $W = x+x^*[x-x]$ $G = \{\{E\}, \{+, -, *, [,], x\}, P, E\}$ e $P = \{E \rightarrow E + E \mid E * E \mid [E], x\}$.

Solução 30.

Questão 31. A palavra gerada pela árvore é ambigua? Por quê? Se sim, gera a palavra de outra forma diferente da anterior.

Solução 31.

3 Linguagens Enumeradas e Recursivamente e Sensíveis ao Contexto

3.1 Máquina de Turing

Questão 32. Defina formalmente a Máquina de Turing; construa um autômato de exemplo e, em seguida, compute uma palavra de sua escolha.

Solução 32. A máquina de turing é definida por uma tupla composta por 8 elementos:

$M = (\Sigma, Q, \delta, q_0, F, V, \beta, s)$, onde:

- Σ é o alfabeto de símbolos de entrada.
- Q é o conjunto de estados
- δ é a função programa
- q_0 é o estado inicial
- F é um subconjunto de Q , denominado de conjunto de estados finais.
- V é o alfabeto dito auxiliar.
- β é o símbolo branco.
- s é o marcador de início da fita.

3.2 Hipótese de Church

Questão 33. Descreva a hipótese Church-Turing.

Solução 33.

4 Computabilidade

4.1 Cálculo Lambda

Questão 34. Reduza as expressões a seguir:

(a) $(\lambda x. 2 * x + 1) 3$

(b) $((\lambda x. (\lambda y. x - y) 2) - 1)$

(c) $((\lambda x. \lambda y. - x y) 9) 4$

Solução 34. Reduções:

(a) $(\lambda x. 2 * x + 1) 3 = 2 * 3 + 1 = 7$

(b) $((\lambda x. (\lambda y. x - y) 2) - 1) = (\lambda y. -1 - y) 2 = -1 - 2 = -3$

(c) $((\lambda x. \lambda y. -xy) 9) 4 = \lambda y. -4y 9 = -4 * 9 = -36$

5 Atividades 2020.1

5.1 Atividade 1

Questão 35. Estudar autômatos finitos determinísticos. Algumas sugestões de materiais auxiliares:

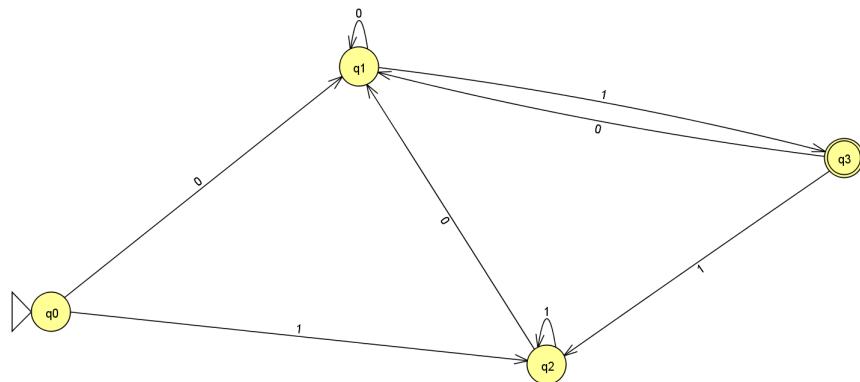
- <http://eaulas.usp.br/portal/video.action?idItem=17133>
- https://www.cin.ufpe.br/~gcb/tc/tc_automato_finitos.pdf
- <http://www2.fct.unesp.br/docentes/dmec/olivete/lfa/arquivos/Apostila.pdf> (página 77)

Solução 35. Leitura a cargo do(a) estudante.

5.2 Atividade 2

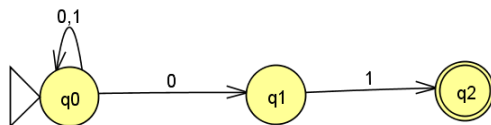
Questão 36. Faça o Autômato Finito Determinístico (AFD) que aceita a linguagem que contém palavras com sufixo "01".

Solução 36. O autômato pode ser representado por:



Questão 37. Faça o problema anterior com o Autômato Finito Não Determinístico (AFN).

Solução 37. O autômato pode ser representado por:



Questão 38. Converta, manualmente, o autômato AFN da questão anterior no AFD, sem usar o JFLAP (você pode usar o JFLAP para comparar com sua resposta).

Solução 38. TODO

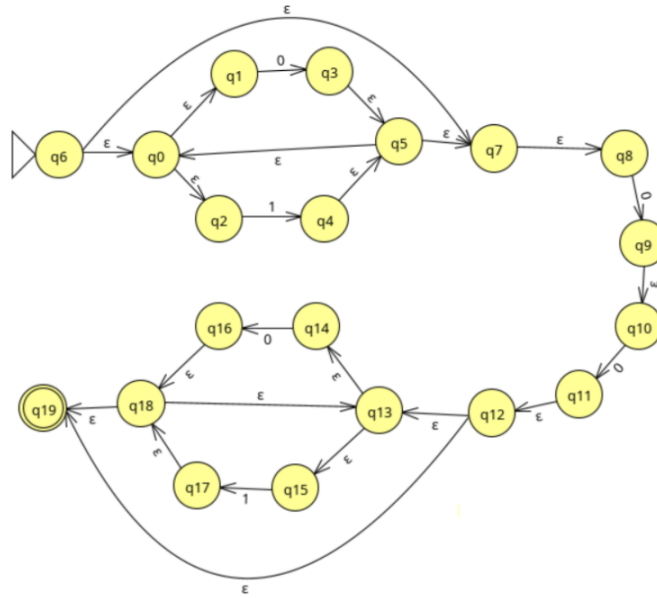
5.3 Atividade 3

Questão 39. Defina a expressão regular que denota todas as cadeias de 0's e 1's com ao menos dois 0's consecutivos.

Solução 39. A Expressão regular é dada por $ER = (0 + 1)^*00(0 + 1)^*$

Questão 40. Faça o autômato finito equivalente ao exercício 1 acima.

Solução 40. Podemos subdividir o autômato em $r1 = 0; r2 = 1; r3 = (0 + 1)^*; r4 = (0 + 1)^*00^*(0 + 1)^* = r3r1r1r3$; Logo, o autômato será definido por:



Questão 41. Defina a gramática regular para a linguagem sobre o alfabeto $\{0, 1\}$, $L = \{w \in \Sigma^* | w \text{ tem no máximo tamanho } 3\}$.

Solução 41. Definindo a gramática $G = (V, T, P, S)$. Tal que:

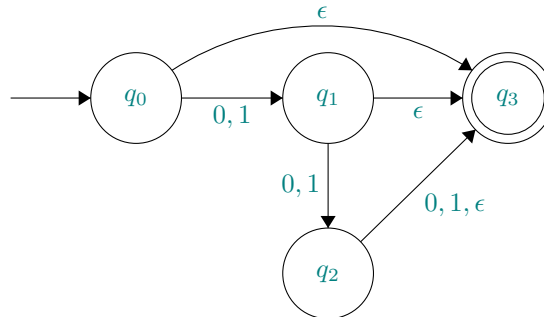
$$V = \{A, B\};$$

$$T = \{0, 1\};$$

$$P = \{S \rightarrow 0A|1A|\epsilon; A \rightarrow 0B|1B|\epsilon; B \rightarrow 0|1|\epsilon\};$$

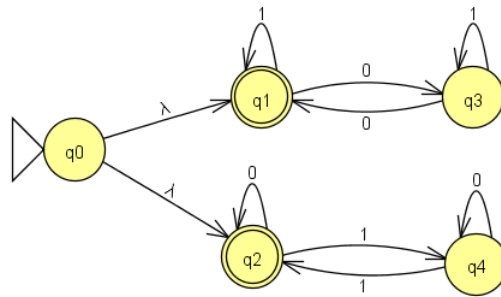
Questão 42. Construa o autômato equivalente ao exercício acima.

Solução 42. O autômato equivalente é dado por:



Questão 43. Construa o AFN com movimentos vazios, com alfabeto $\{0,1\}$, que reconheça a linguagem que tenha números pares de 0's ou de 1's.

Solução 43. O AFN pode ser representado por:



5.4 Atividade 4

Questão 44. Defina a gramática livre de contexto que gere palíndromos. Supondo o alfabeto $\{a,b\}$, mostre a derivação de uma palavra. Ex.: aba

Solução 44. Definindo a gramática $G = (V, T, P, S)$. Tal que:

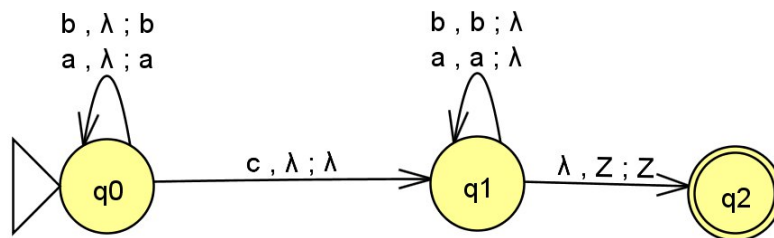
$$V = \{S\};$$

$$T = \{a, b\};$$

$$P = \{S \rightarrow \epsilon; S \rightarrow a; S \rightarrow b; S \rightarrow aSa; S \rightarrow bSb\};$$

Questão 45. Faça o autômato com pilha que reconheça a linguagem $L = wcwR, w \in \{a,b\}^*$ (R significa reversa). Ex.: abcba.

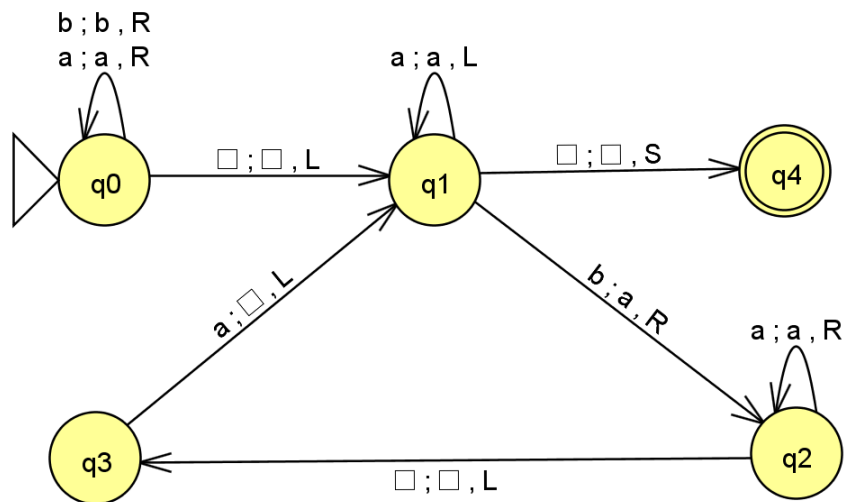
Solução 45. O autômato com pilha pode ser representado por:



5.5 Atividade 5

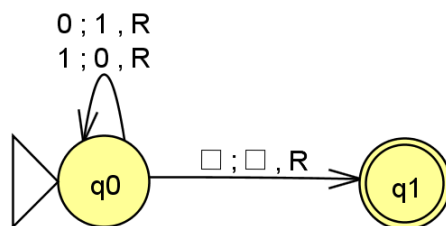
Questão 46. Faça a máquina de Turing com alfabeto $\Sigma = \{a, b\}$ que remova os símbolos b da entrada. Exemplo: $ababa \rightarrow aaa$ (sem brancos/vazios, somente a 's).

Solução 46. A máquina de Turing que resolve o problema pode ser representada por:



Questão 47. Faça a máquina de Turing para inverter os bits de uma entrada binária, ou seja, alfabeto $0, 1$. Exemplo: $[0001011] \rightarrow [1110100]$.

Solução 47. A máquina de Turing para inverter os bits de uma entrada binária é representada por:



5.6 Atividade 6

Questão 48. Suponha as funções *constzero* (número zero), *antecessor*, *id* (identidade), proj_3^3 (projeção), a função subtração *sub*: $N_2 \rightarrow N$ pode ser definida usando recursão primitiva:

- $\text{sub}(x, 0) = \text{id}(x)$
- $\text{sub}(x, y + 1) = \text{antecessor} \circ \text{proj}_3^3(x, y, \text{sub}(x, y))$

Determine o valor de $\text{sub}(2, 3)$

Solução 48. Desenvolvendo as funções, temos:

$$\begin{aligned}
 \text{sub}(2, 3) &= \text{antecessor} \circ \text{proj}_3^3(2, 2, \text{sub}(2, 2)) \\
 &= \text{antecessor} \circ \text{proj}_3^3(2, 2, \text{antecessor} \circ \text{proj}_3^3(2, 1, \text{sub}(2, 1))) \\
 &= \text{antecessor} \circ \text{proj}_3^3(2, 2, \text{antecessor} \circ \text{proj}_3^3(2, 1, \text{antecessor} \circ \text{proj}_3^3(2, 0, \text{sub}(2, 0)))) \\
 &= \text{antecessor} \circ \text{proj}_3^3(2, 2, \text{antecessor} \circ \text{proj}_3^3(2, 1, \text{antecessor} \circ \text{proj}_3^3(2, 0, \text{id}(2)))) \\
 &= \text{antecessor} \circ \text{proj}_3^3(2, 2, \text{antecessor} \circ \text{proj}_3^3(2, 1, \text{antecessor} \circ \text{proj}_3^3(2, 0, 2))) \\
 &= \text{antecessor} \circ \text{proj}_3^3(2, 2, \text{antecessor} \circ \text{proj}_3^3(2, 1, \text{antecessor} \circ \text{proj}_3^3(2, 0, 2))) \\
 &= \text{antecessor} \circ \text{proj}_3^3(2, 2, \text{antecessor} \circ \text{proj}_3^3(2, 1, \text{antecessor}(2))) \\
 &= \text{antecessor} \circ \text{proj}_3^3(2, 2, \text{antecessor} \circ \text{proj}_3^3(2, 1, 1)) \\
 &= \text{antecessor} \circ \text{proj}_3^3(2, 2, \text{antecessor}(1)) \\
 &= \text{antecessor} \circ \text{proj}_3^3(2, 2, 0) \\
 &= \text{antecessor}(0) \\
 &= 0
 \end{aligned}$$

5.7 Atividade 7

Questão 49. Faça as seguintes reduções:

- $((\lambda n. 2n)7)$
- $(\lambda x. (x^2 - 2x + 5))2$
- $(\lambda x. ((\lambda y. (xy))a))b$
- $(\lambda x. (\lambda y. (xy)))(\lambda y. x)$

Solução 49. Reduções:

- $((\lambda n. 2n)7) = (2 * 7) = 14$
- $(\lambda x. (x^2 - 2x + 5))2 = (2^2 - 2 * 2 + 5) = 5$
- $(\lambda x. ((\lambda y. (xy))a))b = ((\lambda y. (by))a) = ba$
- $(\lambda x. (\lambda y. (xy)))(\lambda y. x) = \lambda y. (\lambda y. xy) = \lambda y. x$

5.8 Atividade 8

Questão 50. Explique com suas palavras o que você entende por computabilidade.

Solução 50. Computabilidade está relacionada ao estudo de problemas e à resolução destes através de dado algoritmo em tempo determinado.

Questão 51. Explique com suas palavras o que você entende por princípio da redução.

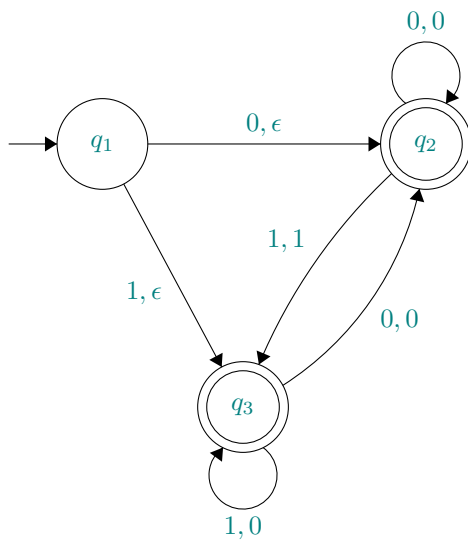
Solução 51. O princípio da redução é uma forma de solucionar problemas utilizando recursividade. Ou seja, é realizado a divisão de um problema menores partes, até que cada um desses pequenos pedaços/problemas possua uma solução conhecida.

6 Atividades 2021.2

6.1 Atividade 4

Questão 52. Faça o autômato com saída para o somador completo.

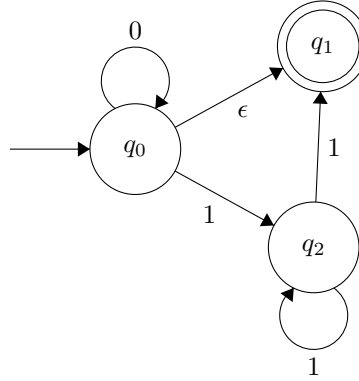
Solução 52. Uma possível representação é:



7 Avaliações Anteriores

7.1 Avaliação 01 - 2020.1

Questão 53. *Que tipo de autômato é ilustrado na figura abaixo? Que linguagem ele reconhece? Mostre a definição formal. Mostre a computação da palavra $w = 0111$. A palavra é aceita?*



Solução 53. *O autômato ilustrado nessa figura é um autômato não-determinístico com movimentos vazios. Esse autômato reconhece as seguintes linguagens: palavras vazias, palavras com sufixo 11 e prefixo ou 0 ou vazio. Além disso, nunca haverá 0 precedido de 1.*

- *Definição formal:* $M = (\{0, 1\}, \{q_0, q_1, q_2\}, \delta, q_0, \{q_1\})$
- *Computação vazia da palavra $w = 0111$:*
 - $\delta^*(\{q_0\}, 0111) = \delta_\epsilon(\{r | r \in \delta(s, 1) \text{ e } s \in \delta^*(\{q_0\}, 011)\})$
 - $\delta^*(\{q_0\}, 011) = \delta_\epsilon(\{r | r \in \delta(s, 1) \text{ e } s \in \delta^*(\{q_0\}, 01)\})$
 - $\delta^*(\{q_0\}, 01) = \delta_\epsilon(\{r | r \in \delta(s, 1) \text{ e } s \in \delta^*(\{q_0\}, 0)\})$
 - $\delta^*(\{q_0\}, 0) = \delta_\epsilon(\{r | r \in \delta(s, 0) \text{ e } s \in \delta^*(\{q_0\}, \epsilon)\})$
- *Sabendo que:*
 - $\delta^*(\{q_0\}, \epsilon) = \delta_\epsilon(\{q_0\}) = \{q_0, q_1\}$
 - $\delta^*(\{q_0\}, 0) = \{q_0, q_1\}$
 - $\delta^*(\{q_0\}, 01) = \{q_0, q_2\}$
 - $\delta^*(\{q_0\}, 011) = \{q_1, q_2\}$
- *A computação da palavra 0111 é $\delta^*(\{q_0\}, 0111) = \{q_1, q_2\}$ e a palavra é aceita.*

Questão 54. *Construa o autômato finito não determinístico equivalente ao da figura acima (questão anterior). Mostre os passos da conversão.*

Solução 54. A máquina de estados é: $M_2 = (\{0, 1\}, \{q_0, q_1, q_2\}, \delta_3, q_0, \{q_1\})$. A função computação pode ser representada da seguinte forma:

δ_1	0	1	ϵ
q_0	$\{q_0\}$	$\{q_2\}$	$\{q_1\}$
q_1			
q_2		$\{q_1, q_2\}$	

Daí, $M_3 = M_{2n} = (\{0, 1\}, \{q_0, q_1, q_2\}, \delta_{2n}, q_0, F_n)$. A computação da palavra vazia é $F_n = \{q_0, q_1, q_2\}$.

- $\delta_\epsilon(q_0) = \{q_0, q_1\}$
- $\delta_\epsilon(q_1) = \{q_1\}$
- $\delta_\epsilon(q_2) = \{q_2\}$

Função δ_n :

- $\delta_n(\{q_0\}, \epsilon) = \{q_0, q_1\}$
- $\delta_n(\{q_1\}, \epsilon) = \{q_1\}$
- $\delta_n(\{q_2\}, \epsilon) = \{q_2\}$

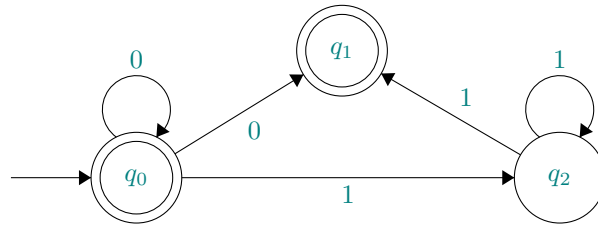
Computação:

- $\delta_n(q_0, 0) = \delta_3(\{q_0\}, 0) = \delta_\epsilon(\{r|r \in \delta(s, 0) \text{ e } s \in \delta(\{q_0\}, \epsilon)\}) = \{q_0, q_1\}$
- $\delta_n(q_0, 1) = \delta_3(\{q_0\}, 1) = \delta_\epsilon(\{r|r \in \delta(s, 1) \text{ e } s \in \delta(\{q_0\}, \epsilon)\}) = \{q_2\}$
- $\delta_n(q_1, 0) = \delta_3(\{q_1\}, 0) = \delta_\epsilon(\{r|r \in \delta(s, 0) \text{ e } s \in \delta(\{q_1\}, \epsilon)\}) = \text{indefinido}$
- $\delta_n(q_1, 1) = \delta_3(\{q_1\}, 1) = \delta_\epsilon(\{r|r \in \delta(s, 1) \text{ e } s \in \delta(\{q_1\}, \epsilon)\}) = \text{indefinido}$
- $\delta_n(q_2, 0) = \delta_3(\{q_2\}, 0) = \delta_\epsilon(\{r|r \in \delta(s, 0) \text{ e } s \in \delta(\{q_2\}, \epsilon)\}) = \text{indefinido}$
- $\delta_n(q_2, 1) = \delta_3(\{q_2\}, 1) = \delta_\epsilon(\{r|r \in \delta(s, 1) \text{ e } s \in \delta * (\{q_2\}, \epsilon)\}) = \{q_1, q_2\}$

Função computação:

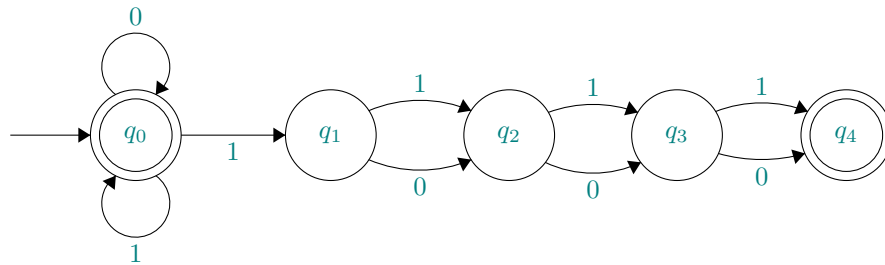
δ_3	0	1
q_0	$\{q_0, q_1\}$	$\{q_2\}$
q_1		
q_2		$\{q_1, q_2\}$

Representação do autômato:



Questão 55. Construa o autômato finito, definição formal e o diagrama, com $\Sigma = \{0, 1\}$, que aceita palavras que tem 1 na quarta posição da direita para a esquerda da palavra.

Solução 55. Representação do autômato:



Função Computação:

δ_2	0	1
q_0	$\{q_0\}$	$\{q_0, q_1\}$
q_1	$\{q_2\}$	$\{q_2\}$
q_2	$\{q_3\}$	$\{q_3\}$
q_3	$\{q_4\}$	$\{q_4\}$
q_4		

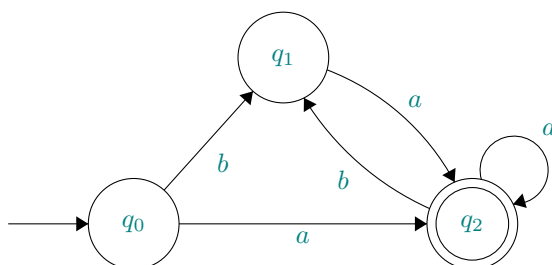
Definição formal: $M_2 = (\{0, 1\}, \{q_0, q_1, q_2, q_3, q_4, q_5\}, \delta_2, q_0, \{q_5\})$

Questão 56. Construa o autômato finito, com alfabeto $\Sigma = \{a, b\}$, que reconheça a linguagem em que qualquer ocorrência de b é imediatamente sucedida por a .

Solução 56. Função computação:

δ	a	b
q_0	q_2	q_1
q_1	q_2	
q_2	q_2	q_1

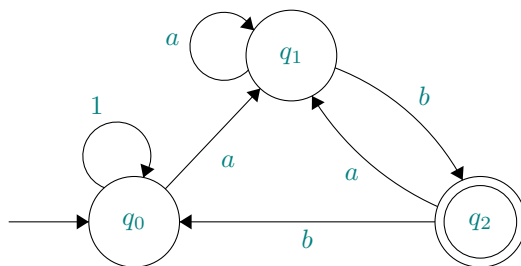
Representação do autômato:



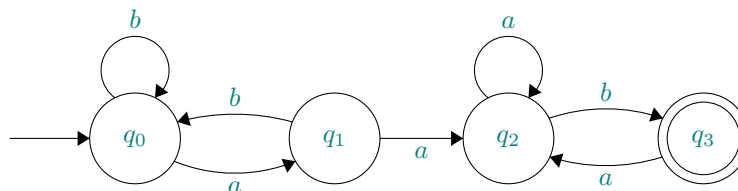
Questão 57. Desenvolva autômatos finitos determinísticos que reconheçam as seguintes linguagens sobre $\Sigma = \{a, b\}$:

- (a) $\{w \mid \text{o sufixo de } w \text{ é } ab\}$.
- (b) $\{w \mid w \text{ possui aa como subpalavra}\}$.

Solução 57. (a) Representação do autônomo:



(b) Representação do autômato:



Questão 58. Descreva a linguagem gerada pelas seguintes expressões regulares:

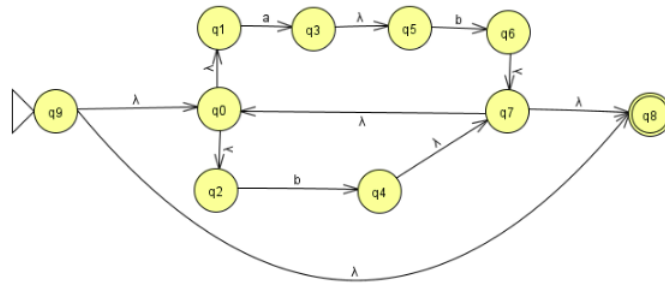
- (a) $(ab + b)^*$
- (b) $a^*(c + b)$
- (c) $a(b + c)^*$
- (d) $(a + \epsilon)(b + c)^*$

Solução 58. *Linguagem gerada:*

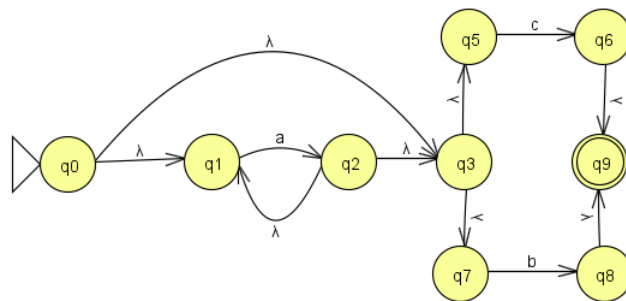
- (a) $(ab + b)^*$: Todas as palavras sobre ab, b , onde a ocorrência de a é sempre sucedida por b .
- (b) $a^*(c + b)$: Todas as palavras acabando exatamente com um b ou um c , e sempre começando com vazio, um a , ou uma sequência de a 's.
- (c) $a(b + c)^*$: Todas as palavras começam por a seguido de vazio ou sequências de b ou c ou de ambas intercaladas.
- (d) $(a + \epsilon)(b + c)^*$: Todas as palavras podem começar com vazio ou a , seguido de vazio ou sequências de b ou c ou de ambas intercaladas.

Questão 59. *Faça o autômato para os itens a) e b) acima.*

Solução 59. (a) *Representação do autômato:*



(b) *Representação do autômato:*



Questão 60. Defina a gramática G que gera a linguagem $L = \{w \mid w \text{ é palavra de } a(ab)^*\}$. Mostre os passos da derivação para palavra $w=aabab$.

Solução 60. A gramática regular é $G = (\{S, A, B\}, \{a, b\}, P, S)$, tal que P é:

- $S \Rightarrow aA$
- $A \Rightarrow aB \mid \epsilon$
- $B \Rightarrow bA$

Temos que:

- $G = \{V, T, P, S\}$:
- $V = \{S, A, B\}$
- $T = \{a, b\}$
- $P = \{S \Rightarrow aA; A \Rightarrow aB \mid \epsilon; B \Rightarrow bA\}$
- $S = \{S\}$

Derivação para a palavra $aabab$:

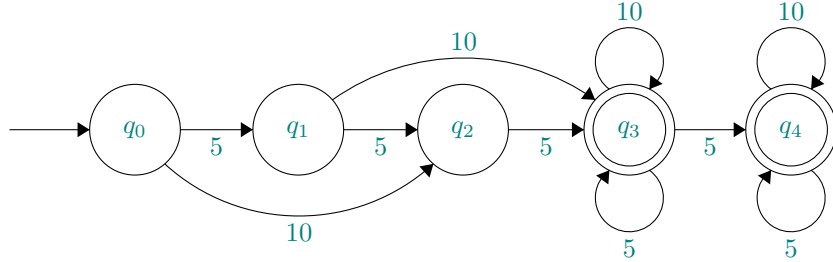
- $S \cdots S \rightarrow aA$
- $aA \cdots A \rightarrow aB$
- $aaB \cdots B \rightarrow bA$
- $aabA \cdots A \rightarrow aB$
- $aabaB \cdots B \rightarrow bA$
- $aababA \cdots A \rightarrow \epsilon$
- $aabab$

Questão 61. Explique com suas palavras o que significa computação no contexto dos autômatos estudados.

Solução 61. A computação tenta compreender as definições e propriedades dos modelos matemáticos como as linguagens, os autômatos e as gramáticas. Além disso, a computação estuda as máquinas abstratas com vários modelos, cada um com diferentes habilidades e limitações, a computação é responsável por descrever precisamente o poder computacional de cada máquina.

Questão 62. Suponha uma máquina que venda dois tipos de doces, um que custa R\$0,15 e outro que custa R\$0,20. Suponha que a máquina aceita somente moedas de R\$0,05 e de R\$0,10. Suponha, ainda, que a máquina não dá troco. Faça o autômato finito que modele o funcionamento da máquina.

Solução 62. *Representação do autômato:*



7.2 Avaliação 02 - 2020.1

Questão 63. *Faça o autômato com pilha que reconheça a linguagem $L = \{0^m 1^n | m \leq n\}$. Mostre um exemplo de computação incluindo a pilha.*

Solução 63. *Definindo a gramática em L1, temos a tupla $G = (V, T, P, S)$. Tal que:*

V : conjunto finito de variáveis ou não-terminais;

T : conjunto finito de símbolos terminais disjunto do conjunto V ;

P : conjunto com regras de produção;

S : simbolo inicial distinguido de V ;

Onde:

$$V = \{S\};$$

$$T = \{0, 1\};$$

$$P = \{S \rightarrow \epsilon; S \rightarrow 1; S \rightarrow 0S1; S \rightarrow S1\};$$

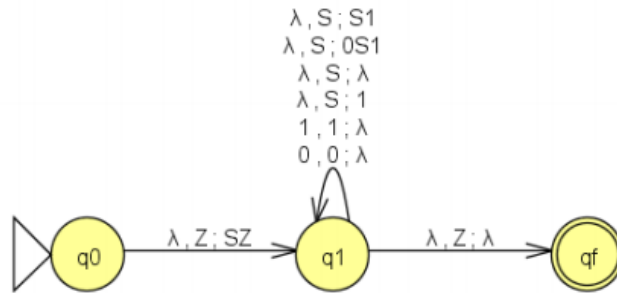
O autômato com pilha pode ser definido pela tupla $M = (\Sigma, Q, \Gamma, q_0, F, \Delta)$. Logo em M1, temos: $M1 = (\Sigma, Q, \delta, q_0, \{q_f\}, \Delta)$.

$$\Sigma = \{0, 1\};$$

$$Q = \{q_0, q_1, q_f\};$$

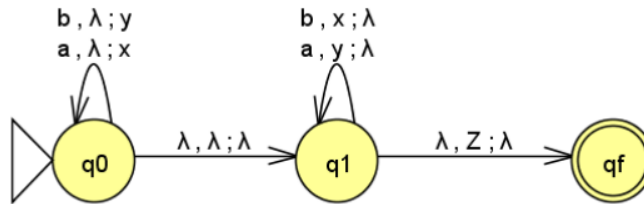
$$\Delta = \{0, 1, S, Z\};$$

Podemos representar o autômato por:



Questão 64. Faça o autômato com pilha que reconheça a linguagem que tenha o mesmo número de as e bs. Mostre um exemplo de computação incluindo a pilha.

Solução 64. Podemos representar o autômato por:



O autômato com pilha pode ser definido pela tupla $M = (\Sigma, Q, \Gamma, q_0, F, \Delta)$. Logo em $M2$, temos: $M2 = (\Sigma, Q, \delta, q_0, \{q_f\}, \Delta)$.

$$\Sigma = \{a, b\};$$

$$Q = \{q_0, q_1, q_f\};$$

$$\Delta = \{a, b, x, y, Z\};$$

Questão 65. Defina a gramática livre de contexto, incluindo as regras de produção, que gera $L2 = \{wcw^R, w \in \{a, b, c\}^*\}$. Mostre a derivação da palavra $w = abcba$. Mostre a árvore de derivação. Existe outra árvore possível? Se sim, mostre pelo menos mais uma árvore. O que significa isso?

Solução 65. Definindo a gramática livre de contexto em $L2$, temos a tupla $G2 = (V, T, P, S)$. Tal que:

$$V = \{S\};$$

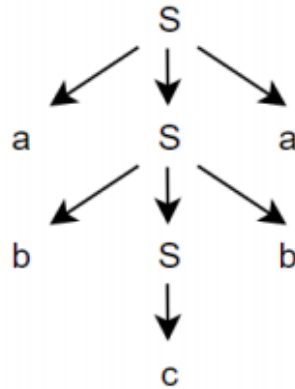
$$T = \{0, 1\};$$

$$P = \{S \rightarrow aSa; S \rightarrow bSb; S \rightarrow cSc; S \rightarrow c\};$$

Derivando a palavra $abcba$:

- $S \rightarrow aSa$
- $aSa \rightarrow abSba$
- $abSba \rightarrow abcba$

A árvore de derivação é definida por:



Existe apenas uma árvore de derivação possível. Podemos concluir que a linguagem livre de contexto não é ambígua.

Questão 66. Defina a gramática livre de contexto, incluindo as regras de produção, que gera expressões algébricas com alfabeto $\{x, y, +, -, *, /, (,)\}$. Mostre a derivação da palavra $w = (x + y)^*x/(y - x)^*y$. Mostre a árvore de derivação. Existe outra árvore possível? Se sim, mostre pelo menos mais uma árvore. O que significa isso?

Solução 66. Definindo a gramática livre de contexto, temos a tupla $G3 = (V, T, P, S)$. Tal que: $G3 = (\{S\}, \{x, y, +, -, *, /, (,)\}, P, S)$.

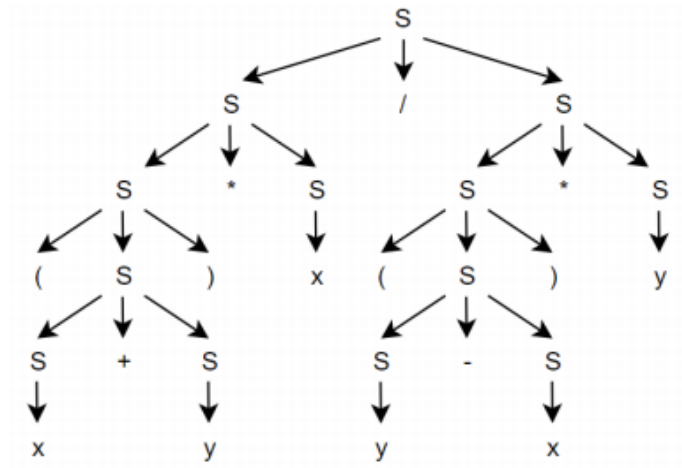
$P = \{S \rightarrow S + S; S \rightarrow S - S; S \rightarrow S * S; S \rightarrow S/S; S \rightarrow (S); S \rightarrow x; S \rightarrow y; \}$;

Derivando a palavra $(x+y)^*x/(y-x)^*y$:

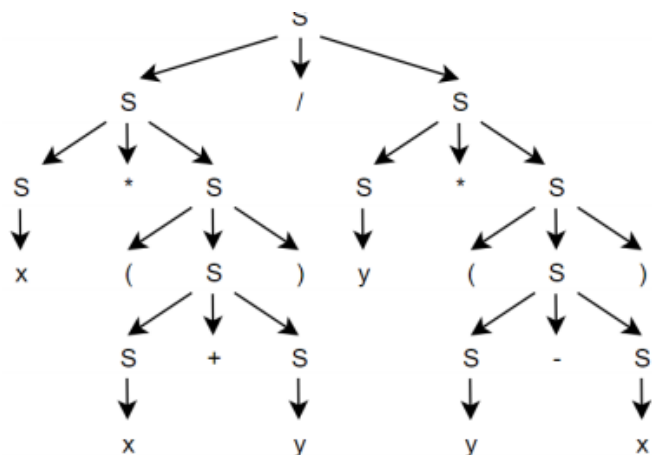
- $S \rightarrow S/S$
- $S/S \rightarrow S * S/S$
- $S * S \rightarrow S * S/S * S$
- $S * S/S * S \rightarrow (S) * S/S * S$

- $(S) * S / S * S \rightarrow (S) * S / (S) * S$
- $(S) * S / (S) * S \rightarrow (S + S) * S / (S) * S$
- $(S + S) * S / (S) * S \rightarrow (S + S) * S / (S - S) * S$
- $(S + S) * S / (S - S) * S \rightarrow (x + S) * S / (S - S) * S$
- $(x + S) * S / (S - S) * S \rightarrow (x + y) * x / (S - S) * S$
- $(x + y) * x / (S - S) * S \rightarrow (x + y) * x / (y - S) * S$
- $(x + y) * x / (y - S) * S \rightarrow (x + y) * x / (y - x) * S$
- $(x + y) * x / (y - x) * S \rightarrow (x + y) * x / (y - x) * y$

A árvore de derivação é definida por:



A árvore de derivação não é única, temos também:



Logo, a gramática livre de contexto é ambígua.

Questão 67. O que faz a máquina de Turing da Figura 1? Mostre a definição formal e uma computação, incluindo a fita.

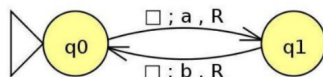


Figura 1

Solução 67. A máquina de turing dada escreve palavras com a's e b's intercalados.

Pode ser definida formalmente pela tupla: $M = (\{ \}, \{q_0, q_1\}, \delta, q_0, \{ \}, \{a, b\}, \beta, b)$

Questão 68. O que faz a máquina de Turing da Figura 2? Mostre a definição formal, a computação da palavra aba, incluindo a fita.

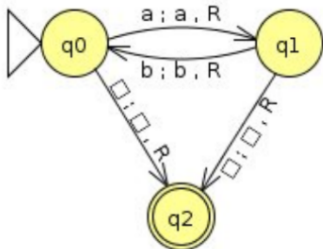


Figura 2

Solução 68. A máquina de turing dada aceita palavras vazias, com a ou a's e bs's intercalados. Pode ser definida formalmente pela tupla: $M = (Q, \Sigma, \delta, q_0, \beta, F)$;

Onde: $M = (\{q_0, q_1, q_2\}, \{a, b\}, \delta, q_0, \beta, \{q_2\})$

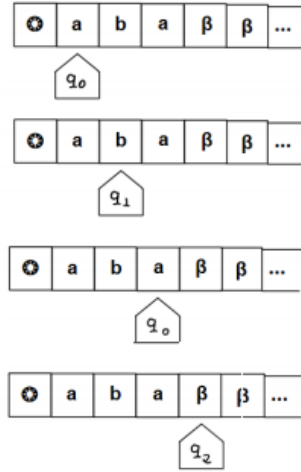
Computando a palavra aba:

$$\delta(q_0, a) = (q_1, a, R)$$

$$\delta(q_1, b) = (q_0, b, R)$$

$$\delta(q_0, a) = (q_1, a, R)$$

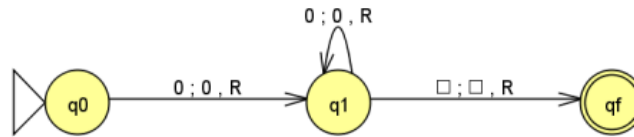
$$\delta(q_1, \beta) = (q_2, \beta, R)$$



Questão 69. Faça a máquina de Turing que reconheça palavras na linguagem $L = 00^*$, onde $\Sigma = \{0, 1\}$.

Solução 69. A máquina de turing pode ser definida pela tupla:

$M = (\{0, 1\}, \{q_0, q_1, q_f\}, \delta, q_0, \{q_f\}, \{0\}, \beta, b)$

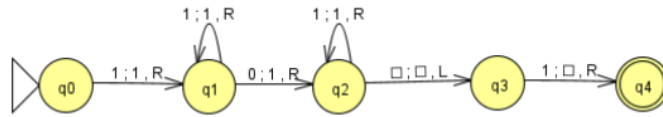


Questão 70. Faça a máquina de Turing que realize soma de números representados por quantidade de uns. Por exemplo, $2=11$, $5=11111$, etc. Os dois operandos são separados por um zero. Exemplo: $2+3=110111=11111$.

Solução 70. O autômato pode ser definido pela tupla:

$M = (\{0, 1\}, \{q_0, q_1, q_2, q_3, q_4\}, \delta, q_0, \{q_4\}, \{1\}, \beta, b)$.

Assim, temos o autômato:



Questão 71. Explique a diferença entre:

- (a) Gramática irrestrita e gramática sensível ao contexto;
- (b) Máquina de Turing como reconhecedor de linguagens recursivamente enumeráveis e de linguagens sensíveis ao contexto.

Solução 71. (a) A gramática irrestrita gera o conjunto das linguagens recursivamente enumeráveis, logo suas regras de produção não possuem restrições. O formalismo reconhecedor equivalente à gramática irrestrita é a Máquina de Turing. - A gramática sensível ao contexto gera o conjunto das linguagens sensíveis ao contexto, suas regras de produção possuem restrições e essas restrições estão ligadas a um contexto, este pode ser definido como um elemento associado ao domínio das regras de produção que possui mais de um elemento em conjunto, podendo ser uma sequência de variáveis e/ou terminais que possuem derivações de tamanho maior ou igual a esta sequência. O formalismo reconhecedor equivalente à gramática sensível ao contexto é a Máquina de Turing com fita limitada.

- (b) A principal diferença entre as duas ocorre no fato de que para uma Linguagem Recursivamente Enumerável a fita é ilimitada, enquanto que para uma Linguagem Sensível ao Contexto ela possui restrição no limite de sua fita. Além disso, a LRE é gerada por uma gramática Irrestrita e tem 3 destinos possíveis: parar e aceitar a palavra, parar e recusar a palavra ou entrar em um loop infinito, no qual não será possível dizer se a palavra é aceita ou não. A Linguagem Sensível ao Contexto está contida na Linguagem Recursivamente Enumerável e, como já dito anteriormente, tem tamanho limitado e sempre irá parar, aceitando ou rejeitando a palavra.