



Universidade Federal de Viçosa - Campus Florestal
Fundamentos da Teoria da Computação - CCF 131
Professor: Daniel Mendes Barbosa

Trabalho Prático 01

Fundamentos da Teoria da Computação

Mariana Souza - 3898

13 de Setembro de 2021

Sumário

1. Introdução	3
2. Desenvolvimento	4
2.1. Linguagens	4
2.2 Questão 11	12
3. Conclusão	14
4. Referências	14

1. Introdução

Para o presente trabalho prático serão aplicados os conhecimentos adquiridos na disciplina de Fundamentos da Teoria da Computação, onde será construído um diagrama de estados para cada linguagem predeterminada na especificação do trabalho prático, assim, para cada linguagem foi desenvolvido o diagrama e o resultado das computações, utilizando a ferramenta Jflap. Além disso, uma breve descrição sobre o que foi realizado para construção do diagrama.

2. Desenvolvimento

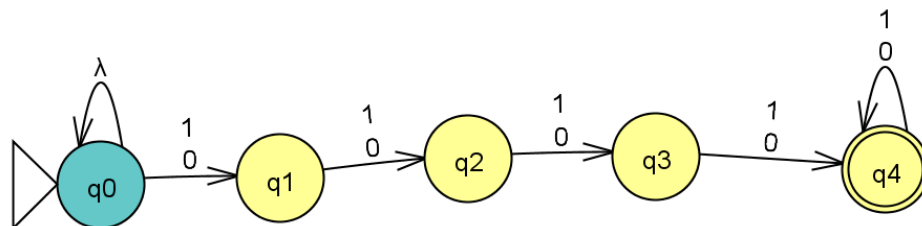
Para alcançar o objetivo do trabalho proposto foi utilizada o ([Link para Download](#)), para a instalação da ferramenta jflap, além do vídeo encontrado no youtube ([Link para o vídeo](#)) que auxilia na instalação do jflap no Windows e também aborda um pouco da forma como funciona a ferramenta assim, para entender mais como é utilizado a construção para os autômatos finitos do Jflap consultei o material ([Link tutorial Jflap](#)), na secção “Finite Automata” constam várias explicações e screenshots da forma como pode ser utilizado.

2.1 Linguagens

Linguagens apresentadas na especificação do trabalho prático proposto. Para cada uma foi anexado um diagrama de estados e resultados das computações utilizando a opção “Finite Automaton” no Jflap, contendo seis entradas, onde três são entradas aceitas e as outras três são rejeitadas.

1. O conjunto das palavras de tamanho maior do que 3.

Diagrama de estados:



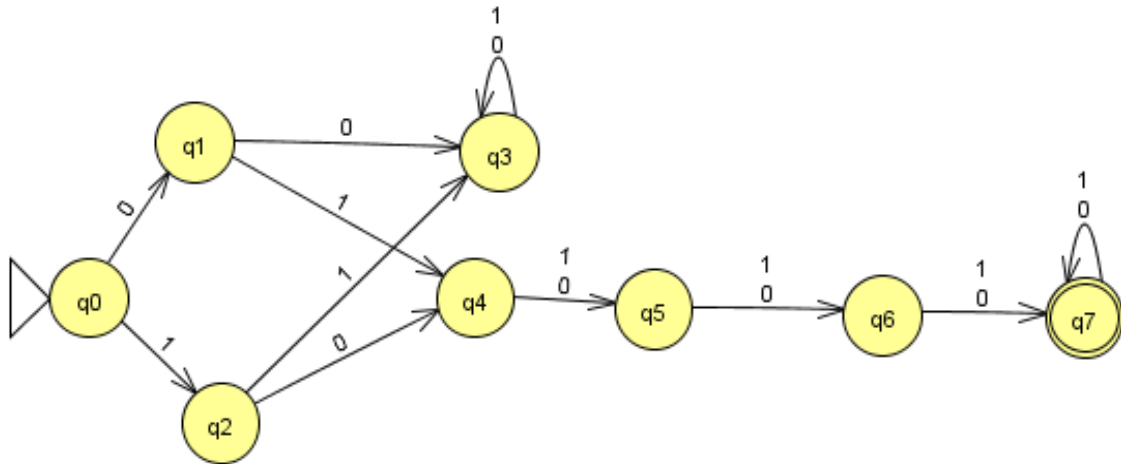
Resultado das computações:

Input	Result
0	Reject
01	Reject
001	Reject
0111	Accept
11000000	Accept
0101010111	Accept

Para o conjunto das palavras de tamanho maior que 3, o diagrama de estados apresenta 4 estados, fazendo as transições com 0 ou 1, onde apenas as palavras maiores que 3 serão aceitas sendo assim o estado final é o q4, que também apresenta um loop, pois caso a palavra seja maior também é considerada. O resultado das computações mostra que para os valores que são maiores que 3 tem a saída aceita.

2. $\{w \in (0,1)^* \mid |w| \geq 5, \text{ e cujos dois primeiros s\u00edmbolos s\u00e3o sempre diferentes um do outro.}$

Diagrama de estados:



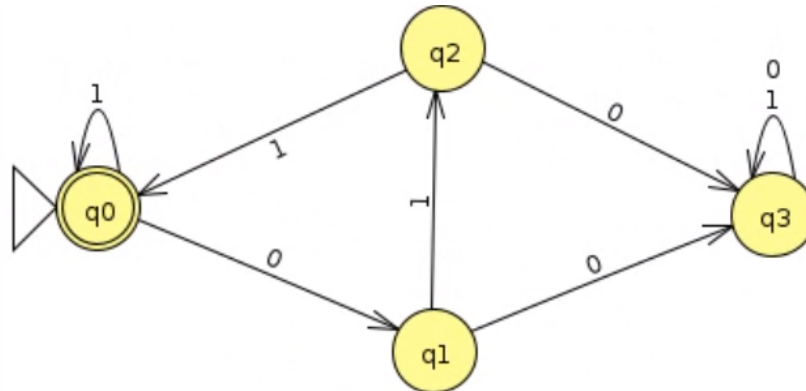
Resultado das computa\u00e7\u00f5es:

Input	Result
1000	Reject
01111	Accept
11001	Reject
1010101010	Accept
010010	Accept
1100011	Reject

Para o diagrama de estados com tamanho maior ou igual a 5, onde os dois primeiros s\u00edmbolos s\u00e3o diferentes um do outro, temos que, foram criados 7 estados, sendo que, para atender a condi\u00e7\u00e3o de s\u00edmbolos diferentes foram criadas transi\u00e7\u00f5es distintas para os estados q1 e q2, at\u00e9 o q7 que garante o final da palavra, tamb\u00e9m foi implementado o q3 que \u00e9 o estado de erro para esse diagrama.

3. $\{w \in (0,1)^* \mid \text{cada } 0 \text{ de } w \text{ é imediatamente seguido de, no mínimo, dois } 1\text{'s}\}$

Diagrama de estados:



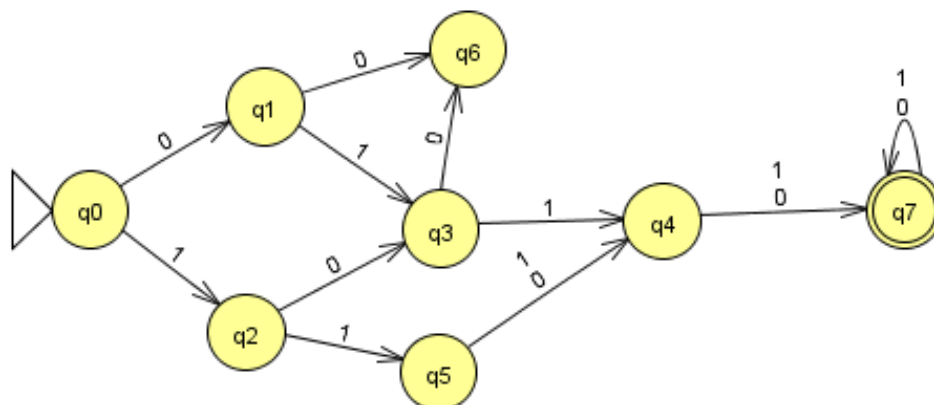
Resultado das computações:

Input	Result
011	Accept
0001	Reject
101011	Reject
011011	Accept
01011	Reject
011011011011	Accept

Onde 0 e W é seguido de no mínimo dois 1's, para o diagrama de estados, foi inserido o q0 que é o estado inicial e o final, onde faz uma transição 0 com o estado q1, e o q1 uma transição 1 para o estado q2, garantindo assim a condição de ter no mínimo dois 1's seguidos pelo 0, tendo o q3 como estado de erro. O resultado das computações mostram os resultados aceitos apenas para o input onde cada zero é seguido de no mínimo dois 1's como '011'.

4. $\{w \in (0,1)^* \mid |w| > 3 \text{ e os primeiros 3 símbolos de } w \text{ contêm, no mínimo, dois } 1\text{'s}\}$

Diagrama de estados:



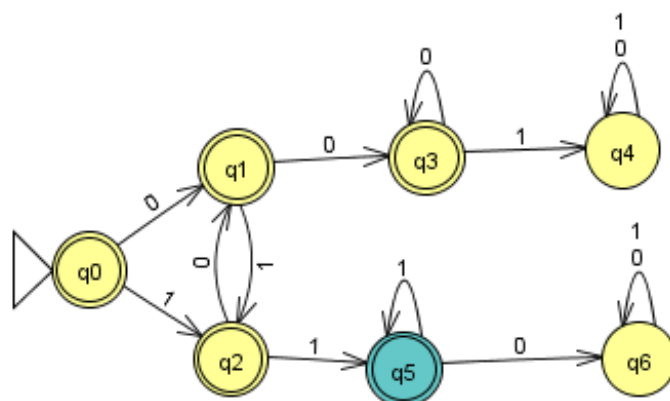
Resultado das computações:

Input	Result
0100	Reject
011	Reject
1100	Accept
01100	Accept
001	Reject
1100	Accept

No diagrama de estados foi inserido 7 estados onde para garantir que os três primeiros símbolos tinham contêm no mínimo dois 1's foi colocado a transição q0 a q2 com 1 e de q2 a q3 com 1, as demais transições são para garantir que outros tipos de palavras serão abrangidas, no resultados das computações temos que todos os inputs que tem dois 1's no início são aceitos.

5. $\{w \in (0,1)^* \mid w \text{ não possui a subpalavra 110 e nem a subpalavra 001}\}$

Diagrama de estados:



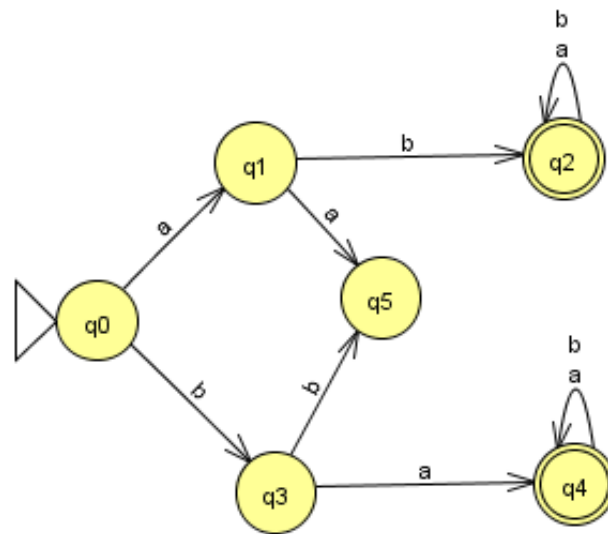
Resultado das computações:

Input	Result
0111	Accept
00100	Reject
1110	Reject
1111	Accept
0000	Accept
1100	Reject

Na questão 5 com o diagrama de estados apresentado acima possui os estados finais para q0, q1, q2, q3 e q5, sendo assim temos que se a subpalavra 110 ou 001 aparecer o resultado será rejeitado, como o apresentado nas computações que foram realizadas, tendo que palavras que não possuem as subpalavras citadas serão aceitas pelo diagrama de estados.

6. $\{w \in (a,b)^* \mid w \text{ sempre começa com dois símbolos diferentes e } |w| > 1\}$

Diagrama de estados:



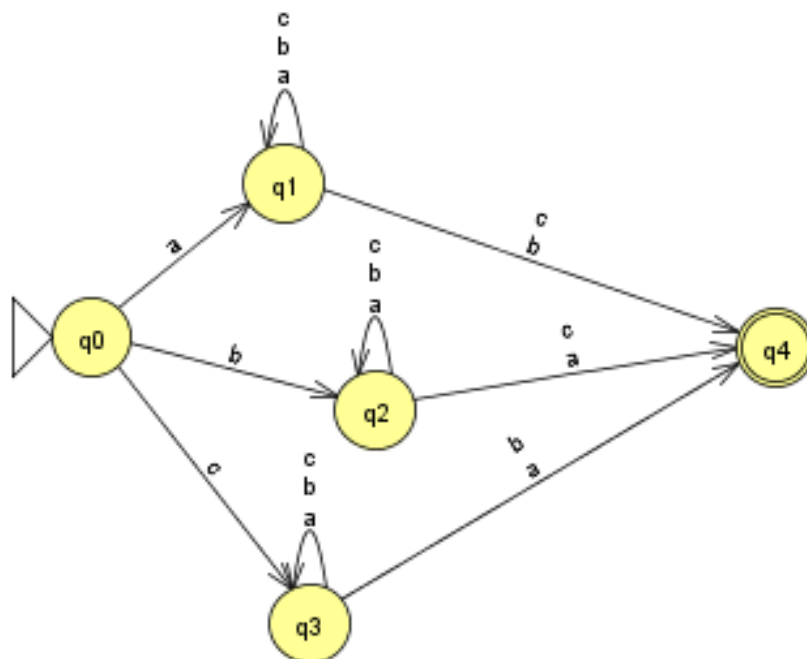
Resultado das computações:

Input	Result
a	Reject
ab	Accept
aab	Reject
abbba	Accept
bbaa	Reject
abaaa	Accept

O diagrama de estados que sempre começa com dois símbolos diferentes foi criado duas transições diferentes para o estado q1 e q2, onde q1 faz transição com “a” e q2 faz transição com “b”, e após fazem transições distintas em vão para o estado final onde garante que a palavra será sempre maior que 1 para ser aceita, isso pode ser visto no resultado das computações onde são aceitos as entradas de palavras maiores que 1 que tem dois símbolos distintos no início, por exemplo ‘abaa’.

7. $\{w \in (a,b,c)^* \mid \text{o primeiro símbolo de } w \text{ é sempre diferente do último}\}$

Diagrama de estados:



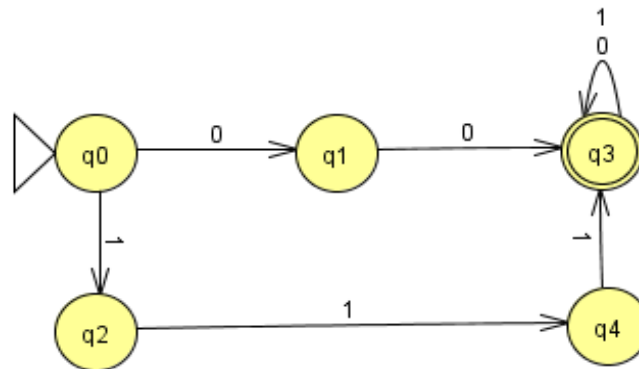
Resultado das computações:

Input	Result
ab	Accept
aa	Reject
abc	Accept
abb	Accept
aaa	Reject
aba	Reject

Para o diagrama de estados onde o primeiro símbolo é sempre diferente do último, foi criado 4 estados onde o estado q0 faz transições com os estados q1, q2 e q3 cada um para cada símbolo, e a transição para o estado final tendo sempre símbolos diferentes dos que foram inseridos na transição inicial, assim, o resultado das computações apresentam aceito para primeiro e último símbolo diferentes.

8. $\{w \in (0,1)^* \mid w \text{ tem como prefixo } 00 \text{ ou } 111\}$

Diagrama de estados:



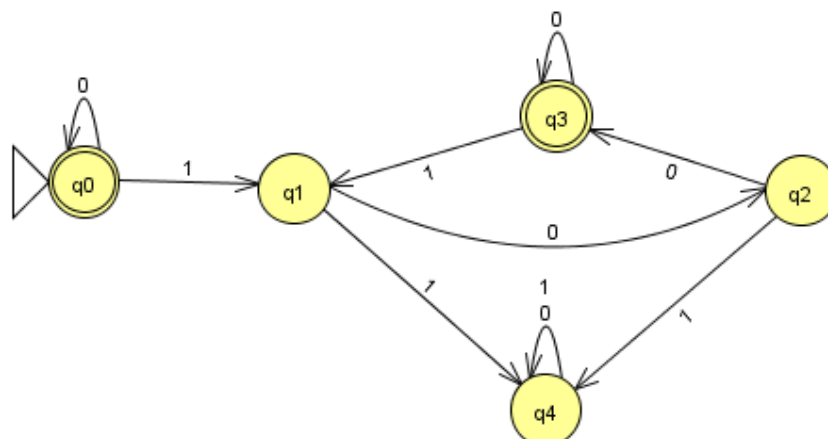
Resultado das computações:

Input	Result
00	Accept
11	Reject
0001	Accept
111	Accept
0101010	Reject
1100	Reject

Nessa questão 8, foi criado no diagrama de estados duas transições com q1 e q2, pois no enunciado mostra que será um prefixo 00 ou 111, sendo assim foi criada transições com 00, caso seja será o estado final e o mesmo para transições 111, no resultado das computações é apresentado resultado certo para palavras com prefixo 00 ou com prefixo 111, como '111'.

9. $\{w \in (0,1)^* \mid w \text{ tem no mínimo dois 0's consecutivos após cada 1}\}$

Diagrama de estados:



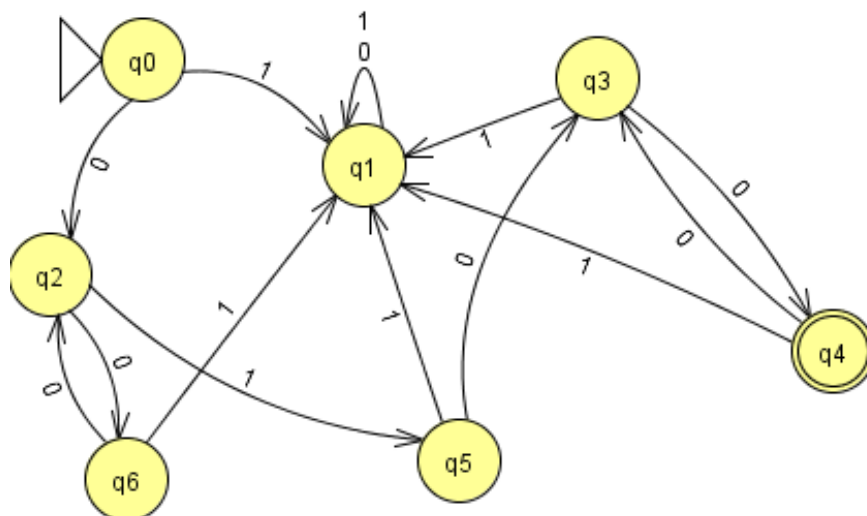
Resultado das computações:

Input	Result
100	Accept
10	Reject
001	Reject
1010	Reject
100100	Accept
0100	Accept

Para a condição de ter no mínimo dois zeros consecutivos a cada , foram implementados 4 estados, onde cada transição entre os estados possui dois zeros para cada um, ou seja, em cada estado possui a opção de dois zeros após uma transição 1, sendo assim o resultado das computações apresentam resultado aceito apenas quando tem dois zeros para cada símbolo 1.

10. $\{w \in \{0,1\}^* \mid w \text{ possui um único } 1 \text{ e um número ímpar de } 0\text{'s antes deste } 1 \text{ e um número par de } 0\text{'s após este } 1\}$

Diagrama de estados:



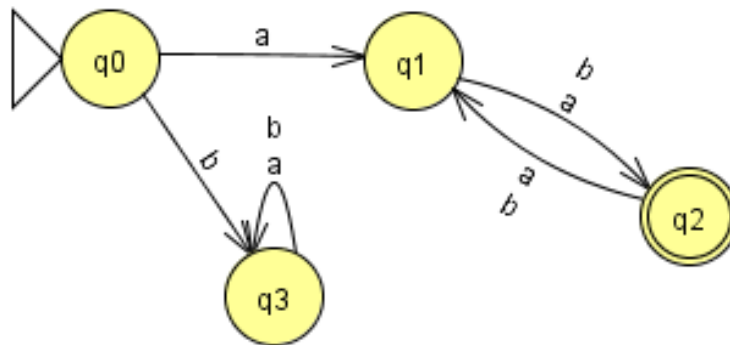
Resultado das computações:

Input	Result
000100	Accept
00100	Reject
00010	Reject
00010000	Accept
0100	Accept
101010	Reject

2.2 Questão 11.

1. $\{ w \in \{a, b\}^* \mid w \text{ começa com } a \text{ e tem tamanho par} \}$

Diagrama de estados:



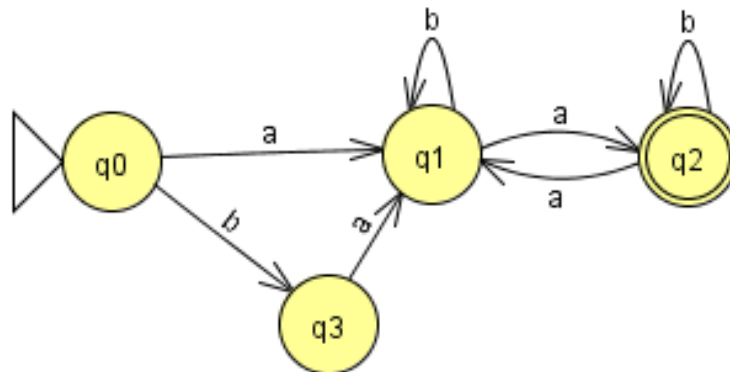
Resultado das computações:

Input	Result
aa	Accept
aba	Reject
aab	Reject
ababab	Accept
ba	Reject
abbbbb	Accept

Conjunto de palavras com o alfabeto $\{a,b\}$ que possuem “a” como símbolo no início e tem tamanho par, para isso, foi implementado um diagrama de estados, onde faz a transição “a” transição q0 para q1, garantindo assim, que ela possui o símbolo inicial “a” e para ter tamanho par foi criada a transição de q1 para q2 onde q2 é estado final, já o q3 é um estado de erro caso a palavra não comece com “a”, nos resultados das computações é perceptível que foram aceitas somente palavras que começam com “a” e são pares, como exemplo: ‘ababab’.

2. $\{ w \in \{a, b\}^* \mid w \text{ tem um número par de } a\text{'s} \}$

Diagrama de estados:



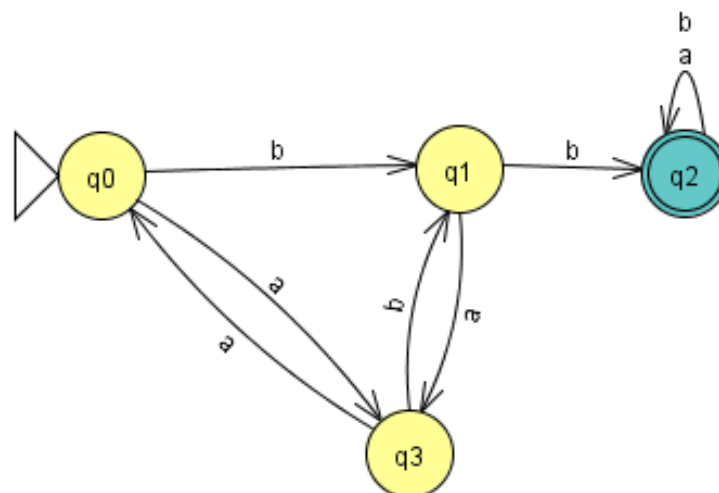
Resultado das computações:

Input	Result
aa	Accept
aba	Accept
aab	Accept
ababab	Reject
ba	Reject
abbbbb	Reject

Conjunto de palavras do alfabeto $\{a,b\}$ que possuem um número par de a 's, sendo assim, o diagrama de estados apresenta transições que tem um número par de a 's, no resultado das computações tem o resultado aceito as palavras que tem um número par de a 's.

3. $\{ w \in \{a, b\}^* \mid w \text{ contém } bb \}$

Diagrama de estados:



Resultado das computações:

Input	Result
ab	Reject
abab	Reject
aabb	Accept
bb	Accept
aabbaabab	Accept
ababa	Reject

Conjunto das palavras com o alfabeto {a,b} que possuem bb, então no diagrama de estados nas transições q1 e q2 garantem essa condição com o estado final em q2, para o resultado das computações só são aceitas as palavras que possuem bb seguidos, como: ‘aabb’ e ‘aabbaabab’.

3. Conclusão

Conclui-se com o trabalho prático que a ferramenta Jflap é muito eficaz para realizar o diagrama de estados, além disso, muito prática, ou seja, não demora muito tempo para ser compreendida, fácil instalação, também, apresenta o resultado das computações o que ajuda a fazer casos de teste para o diagrama que implementado, conferindo assim se a palavra desejada realmente é aceita ou rejeitada pelo diagrama que foi criado. Portanto, auxiliando no entendimento da matéria apresentada em sala de aula da disciplina.

4. Referências

<https://www.jflap.org/jflaptmp/>

<https://www.youtube.com/watch?v=y6xG62WcNYg>

<http://www.jflap.org/>

<https://www.youtube.com/watch?v=2wLtdCzqlh8&t=118s>