FACULDADE DE CIÊNCIAS E TECNOLOGIA

DEPARTAMENTO DE MATEMÁTICA E COMPUTAÇÃO

CIÊNCIA DA COMPUTAÇÃO

JUAN CARDOSO DA SILVA

LUIZ FILIPE MIGUEL MONGE

VINÍCIUS VEDOVOTTO

**Trabalho de LFA – Parte 2**

Relatório e Manual do usuário.



PRESIDENTE PRUDENTE

2019

# Sumário

[Sumário 2](#_Toc25749566)

[Introdução 3](#_Toc25749567)

[Estrutura 4](#_Toc25749568)

[Interface 5](#_Toc25749569)

[Importação e Exportação de Gramática 5](#_Toc25749570)

[Importação e Exportação de Autômatos 6](#_Toc25749571)

[Conversões 7](#_Toc25749572)

[Implementação da importação 7](#_Toc25749573)

[Gramática Regular e Autômatos Finitos 7](#_Toc25749574)

[Implementação da exportação 8](#_Toc25749575)

[Gramática Regular e Autômatos Finitos 8](#_Toc25749576)

[Conversão entre Gramáticas e Autômatos 10](#_Toc25749577)

[Conversão entre Expressões Regulares e Autômatos 10](#_Toc25749578)

[Conversão AFND e AFD 11](#_Toc25749579)

# Introdução

Na segunda parte do trabalho, foi pedido implementação de funcionalidades que permitissem a exportação de gramática, expressões regulares e autômatos, como também a conversão entre gramática-autômato, e autômatos finitos e determinísticos. Utilizando a mesma base da primeira parte do trabalho.

# Estrutura

Na pasta de Exemplos na raiz do projeto existe novos tipos de arquivos, os arquivos XML do tipo .jff com as marcações específicas do Jflap para leitura das exportações a serem realizadas.

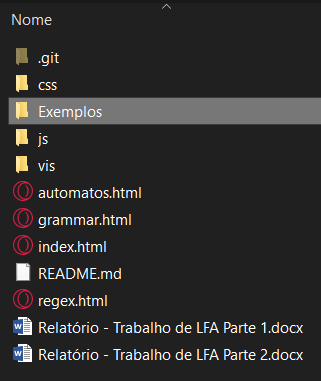
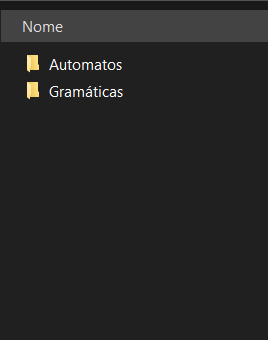


Figura 1 - exemplos de arquivos .jff

Foram adicionados 3 novos arquivos: grammar-importer.js, responsável por importar a gramática, af-er.js, responsável pelas conversões entre o autômatos e expressões regulares; . Também 2 novas bibliotecas, xmlToJson.js e FileSaver.js, que serão detalhadas nos tópicos abaixo.

# Interface

## Importação e Exportação de Gramática

Ao entrar na grammar.html, existe um novo botão chamado exportar e importar, ele abre um modal que contem mais dois botões exportar e importar, exportar uma gramática resulta em um arquivo com nome, data e hora de quando a exportação foi realizada como na figura 3.



Figura 2 - imagem mostrando as novas funcionalidades

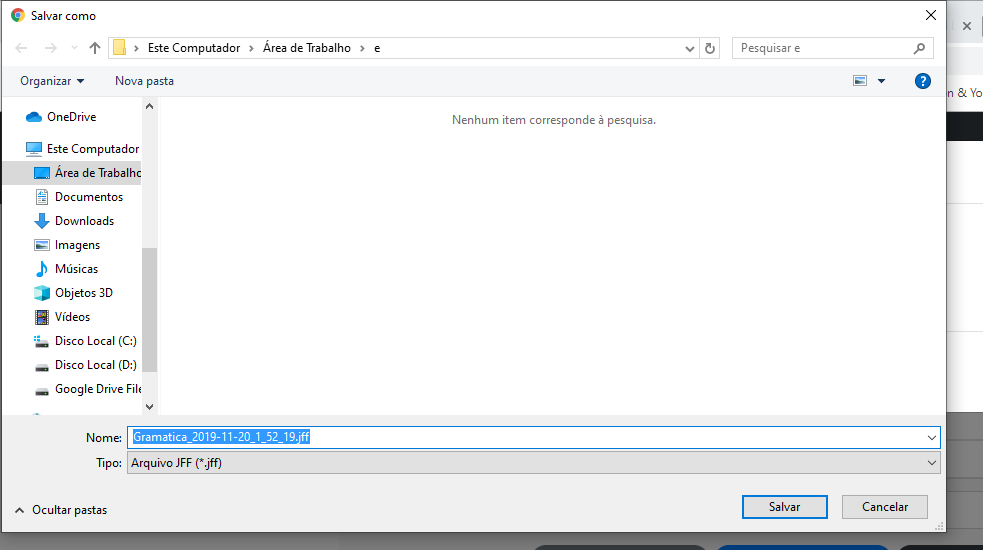


Figura 3 – exemplo de como o arquivo é nomeado ao ser exportado, é possível renomear ele caso queira.

O botão de importar carrega uma gramática e criar as linhas das tabelas conforme a gramática escrita.

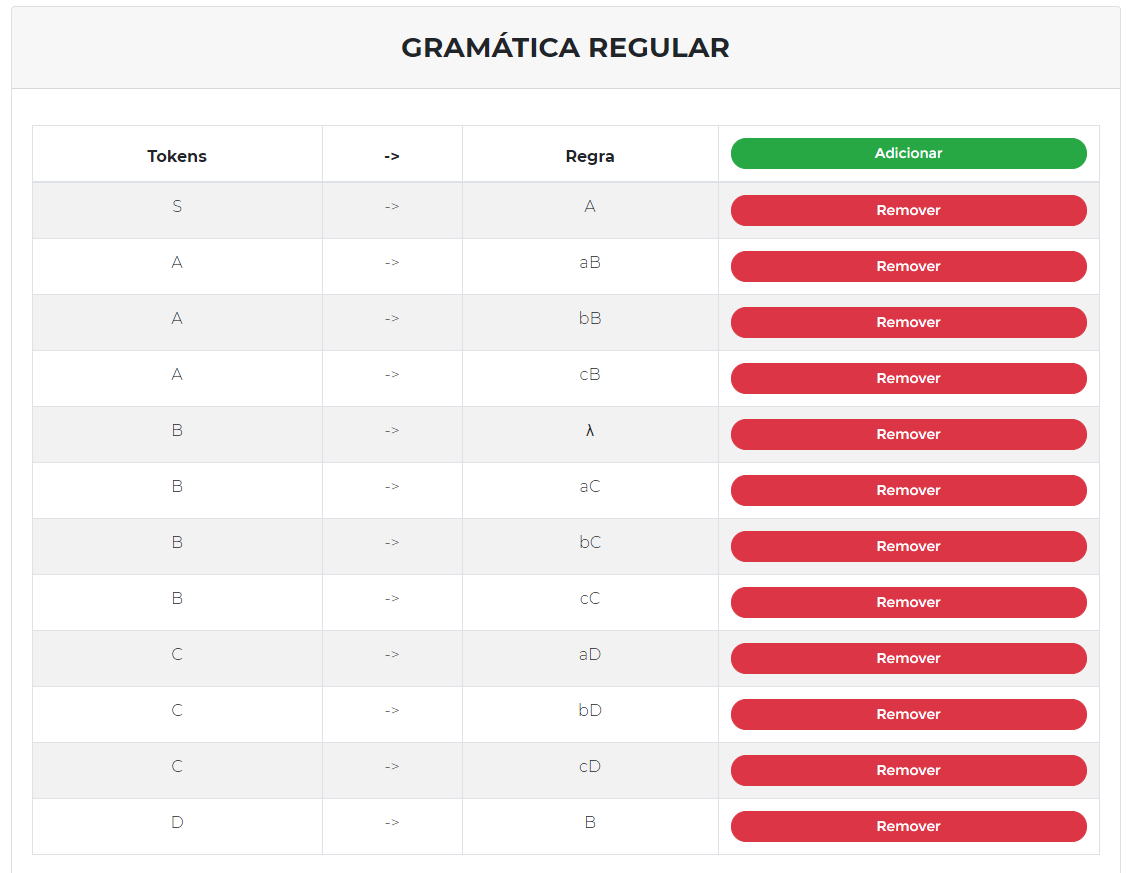


Figura 4 - Exemplo de uma gramática lida, permite cadeias ímpares.

## Importação e Exportação de Autômatos

Na página de autômatos foram adicionados junto ao menu de utilização do Canvas foram adicionados os botões para exportar e importar autômatos, representados pelos ícones de seta para cima e para baixo, respectivamente. O bot

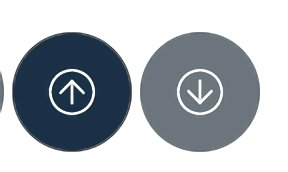


Figura 5 - botões para exportar e importar autômatos

## Conversões

Na página de autômatos fora adicionado a aba “Conversões”, que contém os botões que exibem as janelas das respectivas operações de conversão.



Figura 6 – aba e botões das janelas de conversão

# Implementação da importação

## Gramática Regular e Autômatos Finitos

A importação para a gramática e os autômatos ocorre usando a biblioteca xmlToJson, que realiza a de-serialização do arquivo XML-Jflap e gera um objeto do qual podemos extrair as informações das gramáticas para a tabela.

A partir do objeto gerado, utilizamos as marcações específicas dos autômatos ou gramática para importar as informações nas estruturas e interfaces de gramática e autômatos já implementadas.

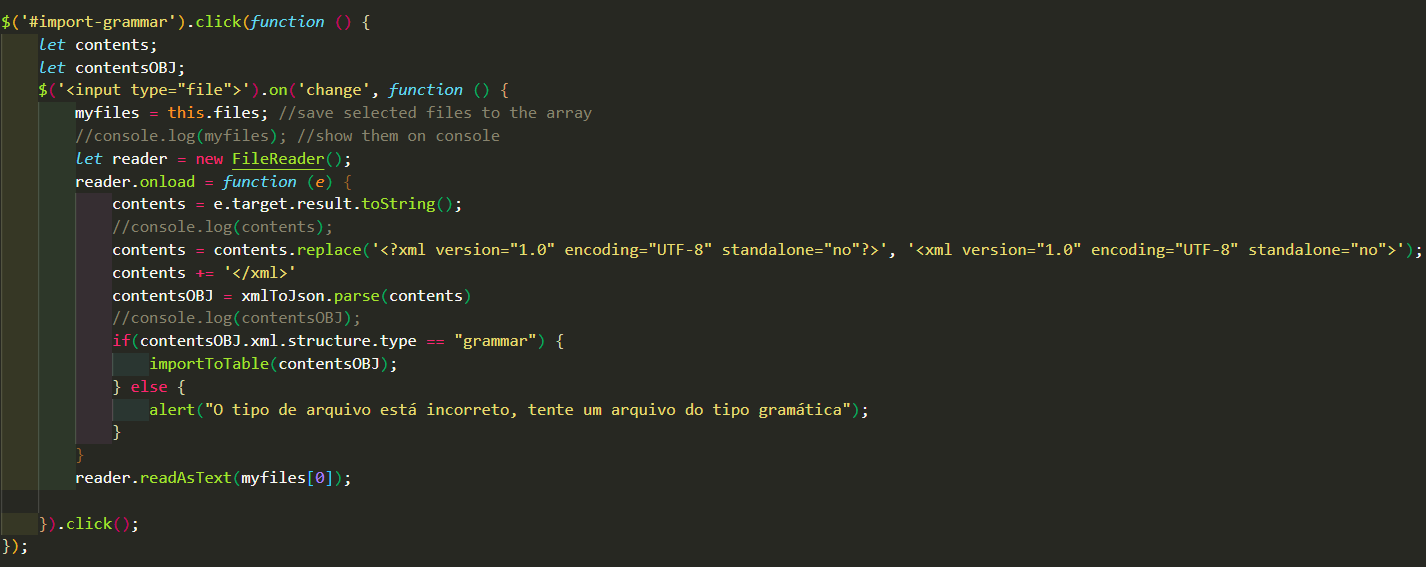
****

Figura 7 - importação da gramática

****

# Implementação da exportação

## Gramática Regular e Autômatos Finitos

Para gerar o conteúdo a ser exportado, em ambos os casos é utilizado uma string onde serão armazenados os valores das estruturas implementadas com as respectivas marcações XML Jflap para que, ao final, esta seja armazenada como arquivo. Utilizando biblioteca chamada FileSaver, definimos a string gerada como os dados a serem contidos no arquivo, e também, o formato, nome e tipo do arquivo.

****

Figura 9 - parte principal do algoritmo de exportação de gramáticas

# Conversão entre Gramáticas e Autômatos

As conversões de gramática para autômato são realizadas na mesma lógica de conversão vista em aula, porem são utilizados objetos para gerar cada parte transitória da gramática para o autômato. São, em geral, três transições sendo elas gramática para tabela de produção do autômato, da tabela para uma lista com os nomes de cada autômato e na ultima parte a criação dos autômatos utilizando a biblioteca Cytoscape junto com a lista e a tabela de objetos para gerara as transições e os estados.

A transformação do autômato para gramática é dada em uma serialização do autômato em XML e a biblioteca de parse para Json e utilizar o objeto gerado que será convertido em um objeto que conterá as transições, variáveis e terminais da tabela, em seguida eles são colocados na tabela. É necessário exportar a gramática gerada para um arquivo .jff para testar depois na área de gramática.

# Conversão entre Expressões Regulares e Autômatos

As conversões entre Expressões Regulares são realizadas de acordo com o que foi visto em aula. Na conversão de Autômatos para Expressões Regulares, é formado um objeto que reduz as transições para instancias únicas, contendo as entradas permitidas para as transições. As transições que repetem para um mesmo nó são convertidas em grupos de alternativa ‘|’ entre as entradas ativas com o delimitador ‘\*’, transições para nós a partir de uma mesma origem são mapeados e diferidos com o delimitador ‘|’ e, por último, ‘loops’ entre nós são mapeados e delimitados com ‘\*’. As expressões são geradas e agrupadas conforme o objeto é analisado.

Para converter a Expressão Regular para um Autômato, a expressão é mapeada utilizando como agrupamento a delimitação de parênteses e os delimitadores adicionais que os acompanham. A partir do objeto gerado pelo mapeamento, são geradas as transições e nós de acordo com o que é especificado. Por fim, os nós e transições são adicionados ao canvas de autômatos com a biblioteca Cytoscape.

# Conversão AFND e AFD