



PEDRO COBIANCHI BORGES PAIVA

**FERRAMENTAS DE APOIO PARA A
DISCIPLINA DE LINGUAGENS FORMAIS E
AUTÔMATOS:
UMA PROPOSTA DE USO**

LAVRAS – MG

2023

PEDRO COBIANCHI BORGES PAIVA

**FERRAMENTAS DE APOIO PARA A DISCIPLINA DE LINGUAGENS
FORMAIS E AUTÔMATOS:
UMA PROPOSTA DE USO**

Monografia apresentada à Universidade Federal de Lavras, como parte das exigências do Programa de Graduação em Ciência da Computação, para a obtenção do título de Bacharel.

Prof. Dr. Ricardo Terra
Orientador

Prof. Dr. Maurício Ronny de Almeida Souza
Coorientador

LAVRAS – MG

2023

**Ficha catalográfica elaborada pela Coordenadoria de Processos Técnicos
da Biblioteca Universitária da UFLA**

Paiva, Pedro Cobianchi Borges

Ferramentas de apoio para a disciplina de Linguagens Formais e
Autômatos : Uma proposta de uso / Pedro Cobianchi Borges Paiva.
– Lavras : UFLA, 2023.

54 p. :

Monografia–Universidade Federal de Lavras, 2023.

Orientador: Prof. Dr. Ricardo Terra.

Bibliografia.

1. TCC. 2. Monografia. 3. Dissertação. 4. Tese. 5. Trabalho
Científico – Normas. I. Universidade Federal de Lavras. II. Título.

CDD-808.066

PEDRO COBIANCHI BORGES PAIVA

**FERRAMENTAS DE APOIO PARA A DISCIPLINA DE LINGUAGENS
FORMAIS E AUTÔMATOS: UMA PROPOSTA DE USO**

Monografia apresentada à Universidade Federal de Lavras, como parte das exigências do Programa de Graduação em Ciência da Computação, para a obtenção do título de Bacharel.

APROVADA em 05 de julho de 2023.

Prof. Dr. Ricardo Terra	UFLA
Prof. Dr. Maurício Ronny de Almeida Souza	UFLA
Prof. Dr. Luiz Henrique Andrade Correia	UFLA
Prof. Dr. Rafael Serapilha Durelli	UFLA

Prof. Dr. Ricardo Terra
Orientador

Prof. Dr. Maurício Ronny de Almeida Souza
Coorientador

**LAVRAS – MG
2023**

*Dedico este trabalho à minha mãe Marialva, fonte diária de força e inspiração, e
à minha avó Eliza que nunca me deixou desistir.*

AGRADECIMENTOS

Agradeço à minha mãe Marialva e à minha avó Eliza por sempre me apoiarem e nunca me deixarem desistir. Aos meus amigos da república Alevan, Murilo e Rafael, à Comp Júnior e às amizades verdadeiras lá construídas. Aos docentes e técnicos do Departamento de Ciência da Computação da UFLA e do Instituto de Matemática e Computação da UNIFEI, em especial ao professor Ricardo Terra por me acompanhar em grande parte da minha jornada acadêmica, ao professor Maurício Ronny por me coorientar, ao professor Roberto Costa por me incentivar nos momentos difíceis da minha vida e ao professor Teófilo por sempre me apoiar desde antes da graduação. Cheguei onde cheguei porque me apoiei no ombro de gigantes.

“É melhor acender uma vela do que praguejar contra a escuridão.”
(ADÁGIO)

RESUMO

A disciplina de Linguagens Formais e Autômatos (LFA) possui grande relevância na área da computação. Entretanto, a natureza matemática dos conteúdos torna difícil o processo de ensino aprendizagem. Aliado a isso, está a abordagem tradicional dentro das salas de aula, com pouca inovação, aulas centradas no professor e pouca participação dos alunos, o que reflete em altos índices de reprovação. Este estudo, portanto, propõe uma estratégia de uso de ferramentas centrado na ideia de que recursos educacionais digitais expandem o potencial e a eficácia do processo de ensino e aprendizagem tradicional. Inicialmente, levantou-se os conteúdos práticos mais recorrentes da disciplina nos livros de referência e nas ementas das cinco melhores universidades do Brasil. Em seguida, após uma pesquisa exaustiva das principais ferramentas, avaliou-se Automata, Automata Tutor, CMSimulator, JFLAP, LFApp e LFAweb em relação tanto as funcionalidades providas como também a cobertura dos conteúdos. Em suma, conclui-se que Automata Tutor, JFLAP e LFAweb são complementares e devem ser utilizadas em conjunto. Automata Tutor para aplicação de exercícios, JFLAP para algoritmos de simulação e LFAweb para algoritmos de transformação.

Palavras-chave: Linguagens Formais e Autômatos. Ferramentas. Estratégia de uso.

ABSTRACT

Formal Languages and Automata (FLA) is a highly relevant discipline in the field of computer science. However, teaching automata theory and formal languages can be challenging for instructors and learning the subject can be difficult for students due to its mathematical nature, leading to high failure rates. This study proposes a strategy that utilizes digital educational resources to enhance the potential and effectiveness of the traditional teaching and learning process. Initially, we identified the most frequently encountered practical content in the discipline by reviewing reference books and syllabi from the top five universities in Brazil. Next, after we conducted an extensive search of various tools, we evaluate Automata, Automata Tutor, CMSimulator, JFLAP, LFApp, and LFAweb w.r.t. their functionalities and content coverage. In summary, we concluded that Automata Tutor, JFLAP, and LFAweb are complementary and should be used together. Automata Tutor is recommended for exercises and applications, JFLAP for simulation algorithms, and LFAweb for transformation algorithms.

Keywords: Formal Languages and Automatas. Tools. Usage Strategy.

LISTA DE FIGURAS

Figura 2.1 – Hierarquia de Chomsky	23
Figura 3.1 – Processo para responder as QPs	28
Figura 4.1 – Biblioteca Automata	33
Figura 4.2 – Automata Tutor - Professor	34
Figura 4.3 – Automata Tutor - Aluno	34
Figura 4.4 – CMSimulator	36
Figura 4.5 – JFLAP	37
Figura 4.6 – LFApp	38
Figura 4.7 – LFAweb	38
Figura 4.8 – Simulações utilizando o JFLAP	42
Figura 4.9 – Transformações utilizando o LFAweb	43
Figura 4.10 – Diagrama de Venn ilustrando a distribuição da quantidade de conteúdos contemplados pelas ferramentas	47

LISTA DE TABELAS

Tabela 4.1 – Principais conteúdos práticos de LFA	32
Tabela 4.2 – Comparativo das ferramentas	41
Tabela 4.3 – Recomendação de uso da ferramenta por conteúdo	46

SUMÁRIO

1	INTRODUÇÃO	21
2	Linguagens Formais e Autômatos	23
3	DESENHO DA PESQUISA	27
3.1	Questões da pesquisa	27
3.2	Métodos	27
3.2.1	Etapa 1 - Levantar os principais conteúdos	28
3.2.2	Etapa 2 - Buscar ferramentas	29
3.2.3	Etapa 3 - Analisar as ferramentas	29
3.2.4	Etapa 4 - Propor uma estratégia de uso para as ferramentas	30
4	CONDUÇÃO DA PESQUISA	31
4.1	QP1: Quais conteúdos são mais recorrentes no contexto de ensino de LFA?	31
4.2	QP2: Quais as ferramentas disponíveis para o ensino e aprendizagem de LFA?	32
4.3	QP3: Como as ferramentas disponíveis apoiam os conteúdos de ensino de LFA?	39
4.4	Proposta de uma estratégia de uso para as ferramentas	40
5	CONCLUSÃO	49
	REFERÊNCIAS	51
A	APÊNDICE	53

1 INTRODUÇÃO

A disciplina de Linguagens Formais e Autômatos (LFA) possui grande relevância nos cursos de computação, dado que estuda conceitos importantes para a formação profissional na área (CHESNEVAR; GONZÁLEZ; MAGUITMAN, 2004) e dá suporte para disciplinas como Compiladores e Inteligência Artificial (DOGNINI; RAABE, 2003). Em contrapartida, é desafiador por parte dos professores ensinarem e difícil para alunos aprenderem a teoria dos autômatos e as linguagens formais devido a natureza matemática dos conteúdos (VIJAYALASKHMI; KARIBASAPPA, 2012), o que reflete em altos índices de reprovação, aproximadamente 50% na Universidade Federal de Lavras (TERRA, 2023).

É comum que os conteúdos programáticos e didáticos da disciplina tenham uma abordagem tradicional, centrada no professor, com aulas expositivas que oferecem pouca diferenciação e inovação, já que os principais resultados foram obtidos nas décadas de 1960 e 1970 (RAMOS, 2009). No entanto, essa abordagem pode limitar a exploração de novas perspectivas que poderiam contribuir significativamente para a qualidade do ensino e aprendizagem. Este estudo é centrado na ideia de que recursos educacionais digitais expandem o potencial e a eficácia do processo de ensino e aprendizagem tradicional, uma vez que a utilização correta desses recursos propicia o aprendizado lúdico do estudante (CASCINI; CAMPOS, 2015).

Além disso, essas ferramentas possuem as seguintes contribuições que são consideradas relevantes no ensino: (i) ferramentas interativas oferecem recursos visuais e gráficos que podem auxiliar a compreender conceitos abstratos; (ii) exemplos práticos aproximam a teoria com situações reais; (iii) a disponibilidade de ferramentas proporciona ao aluno estudo independente; e (iv) *feedbacks* mais rápidos e individualizados, os quais destacam os erros e facilitam aos alunos identificarem suas principais dificuldades.

Diante disso, o objetivo do trabalho é propor uma estratégia de uso de ferramentas que podem apoiar o processo de ensino e aprendizagem de LFA, com a intenção de otimizar esse processo durante o curso da disciplina. A elaboração da proposta consiste em: (i) analisar os conteúdos presentes nos livros didáticos e ementas de LFA das principais universidades do país; (ii) buscar ferramentas que auxiliam no ensino e aprendizagem de LFA e (iii) analisar como as ferramentas disponíveis apoiam os principais conteúdos.

Este trabalho está organizado da seguinte forma. O Capítulo 2 introduz LFA e os níveis da Hierarquia de Chomsky. O Capítulo 3 apresenta três questões de pesquisa que conduzem o trabalho, bem como a metodologia adotada. O Capítulo 4 apresenta os resultados parciais obtidos bem como a proposta da estratégia de uso das ferramentas. Por fim, o Capítulo 5 relata os resultados obtidos, principais contribuições e trabalhos futuros.

2 LINGUAGENS FORMAIS E AUTÔMATOS

Linguagens Formais e Autômatos (LFA) é uma importante área da computação que aborda modelos matemáticos que possibilitam a especificação e reconhecimento de linguagens, suas propriedades e características (SUDKAMP, 2007). Essa área é considerada clássica no contexto da computação, uma vez que seus principais resultados foram obtidos nas décadas de 1960 e 1970 (RAMOS, 2009).

Sua base de conhecimento é dividida em duas subáreas, a teoria dos autômatos e a teoria das linguagens formais, as quais eram consideradas independentes até a década de 60. Foi então a partir de pesquisas e resultados obtidos, principalmente pelo linguista Noam Chomsky em 1956, que essas duas subáreas foram unidas e se tornaram indissociáveis uma da outra (RAMOS, 2009). O trabalho de Chomsky foi de prover sistemas formais para geração de palavras sintaticamente corretas (SUDKAMP, 2007), conhecido como Hierarquia de Chomsky. A Figura 2.1 ilustra a hierarquia e a associação entre as duas subáreas. É importante salientar que cada categoria é um subconjunto próprio da categoria superior (SUDKAMP, 2007).

Tipo	Linguagem	Gramática	Máquina aceitadora
Tipo 0	Linguagem Recursivamente Enumerável (LRE)	Gramática irrestrita (GI)	Máquina de Turing (MT)
Tipo 1	Linguagem Recursiva (LREC)	Gramática Sensível ao Contexto (GSC)	Autômato Linearmente Limitado (ALL)
Tipo 2	Linguagem Livre de Contexto (LLC)	Gramática Livre de Contexto (GLC)	Autômato com Pilha (AP)
Tipo 3	Linguagem Regular (LR)	Gramática Regular (GR)	Autômato Finito (AF)



↑
 Maior poder computacional
 Menos restritiva

Fonte: Adaptado de Sudkamp (2007)

Figura 2.1 – Hierarquia de Chomsky

A Linguagem Regular (LR) é a linguagem mais simples que existe na hierarquia, porém muito importante em analisadores léxicos (SUDKAMP, 2007), comumente utilizados através de expressões regulares. Para identificar os padrões dessas linguagens, é implementado um Autômato Finito (AF) que sempre para em estado de aceitação se a palavra pertence a linguagem ou em estado de rejeição caso não pertença (SUDKAMP, 2007). O AF se apresenta de três formas, todas com o mesmo poder computacional: Autômato Finito Determinístico (AFD), Autômato Finito Não-Determinístico (AFND) e Autômato Finito Não-Determinístico com transições λ (AFND- λ). Palavras pertencentes a essa linguagem podem ser geradas através das Gramáticas Regulares (GR), que são definidas por uma quádrupla $GR = (V, \Sigma, P, S)$, onde V é o conjunto dos símbolos não terminais, Σ é o conjunto de símbolos terminais, P é o conjunto de regras de produção e S o símbolo inicial da gramática. Cada regra do conjunto P é do tipo $\mu \rightarrow v$ tal que $\mu \in V$ e $v \in \lambda \mid \Sigma \mid \Sigma V$ (SUDKAMP, 2007).

A Linguagem Livre de Contexto (LLC) tem um maior poder computacional que a LR e consegue ser bem adequada para a criação de sintaxes de linguagens de programação (SUDKAMP, 2007). Para reconhecer esse tipo de linguagem, existem os Autômatos com Pilha (AP) que, diferentemente dos AFs, operam com uma pilha de memória, o que faz terem um maior poder computacional que os AFs. O AP que tem o poder computacional de decidir se a palavra pertence a classe das LLC é o Autômato com Pilha Não-Determinístico (APND). Palavras pertencentes a essa linguagem podem ser geradas por meio de Gramáticas Livres de Contexto (GLC) que são definidas pela quádrupla $GLC = (V, \Sigma, P, S)$ previamente definida, porém cada regra do conjunto P é do tipo $\mu \rightarrow v$ tal que $\mu \in V$ e $v \in (V \cup \Sigma)^*$ (SUDKAMP, 2007)

A Linguagem Recursiva (LREC) tem um maior poder computacional comparada à LLC (e conseqüentemente à LR), assim como representado na imagem da hierarquia. São reconhecidas por um Autômato Linearmente Limi-

tado (ALL) que é similar a uma Máquina de Turing (MT), porém com memória limitada (HOPCROFT; ULLMAN, 1969). Para gerar palavras que pertençam a essa linguagem, é necessário uma Gramática Sensível ao Contexto (GSC), que é definida pela quádrupla $GSC = (V, \Sigma, P, S)$. Porém suas regras de produção P são menos restritivas pois são da forma $\mu \rightarrow v$ onde $\mu \in (V \cup \Sigma)^+$ e $v \in (V \cup \Sigma)^+$ tal que $|\mu| \leq |v|$ (SUDKAMP, 2007), ou seja, a única restrição é que as regras sejam não contráteis, isto é, as regras do lado esquerdo tem tamanho menor ou igual as do lado direito.

A Linguagem Recursivamente Enumerável (LRE) é a linguagem com maior poder computacional na hierarquia. É reconhecida por uma Máquina de Turing (MT), modelo abstrato para os computadores modernos (SUDKAMP, 2007) e modelo teórico com o maior poder computacional. Se a MT para para todas as palavras pertencentes a linguagem mas não para para palavras que não pertencem a linguagem, então é uma MT reconhecedora e a linguagem é uma LRE (SUDKAMP, 2007). A MT também possui suas variantes, como MT com fita bidirecional, MT multifita e MT não determinística, todas com poder computacional equivalente ao de uma MT determinística. Para gerar palavras que pertençam à LRE, é necessária uma Gramática Irrestrita (GI), que é definida pela mesma quádrupla $GI = (V, \Sigma, P, S)$, entretanto suas regras são quase totalmente irrestritas, isto é, são do tipo $\mu \rightarrow v$, onde $\mu \in (V \cup \Sigma)^+$ e $v \in (V \cup \Sigma)^*$ tal que basicamente a única restrição é $|\mu| > 0$ (SUDKAMP, 2007).

Este capítulo apresentou os conceitos e definições gerais referentes a disciplina de LFA. Esse conhecimento é essencial para o entendimento da pesquisa, cuja metodologia é introduzida já no próximo capítulo.

3 DESENHO DA PESQUISA

Neste capítulo, são pontuados o objetivo geral e as questões de pesquisa, que podem ser vistas como objetivos secundários. O objetivo geral consiste em propor uma estratégia de uso das ferramentas, com a intenção de otimizar o ensino e aprendizagem dos conteúdos trabalhados durante a disciplina de LFA.

A Seção 3.1 apresenta três questões de pesquisa que norteiam o desenvolvimento e a Seção 3.2 apresenta os métodos para responder cada uma das questões.

3.1 Questões da pesquisa

As seguintes questões de pesquisa auxiliam a alcançar o objetivo geral do trabalho:

- **QP1.** *Quais conteúdos são mais recorrentes no contexto de ensino de LFA?* O objetivo é encontrar nas ementas das universidades e em livros didáticos quais são os conteúdos de ensino e como são trabalhados em LFA.
- **QP2.** *Quais as ferramentas disponíveis para o ensino e aprendizagem de LFA?* O objetivo é encontrar ferramentas que auxiliem o ensino e aprendizagem de LFA.
- **QP3.** *Como as ferramentas disponíveis apoiam os conteúdos de ensino de LFA?* O objetivo é mapear como as características e funcionalidades de cada ferramenta auxiliam nos conteúdos de LFA.

3.2 Métodos

Esta seção detalha a estratégia adotada para responder cada uma das questões de pesquisa (Seção 3.1) e, para isso, é definido um processo composto por quatro etapas lineares conforme ilustrado na Figura 3.1.

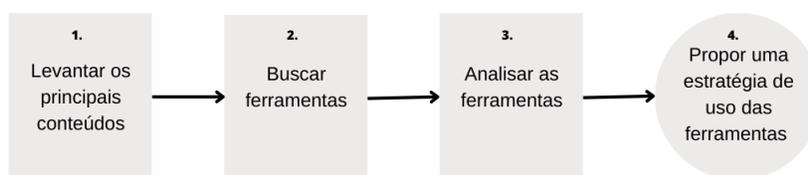


Figura 3.1 – Processo para responder as QPs

A Etapa 1 responde a QP1 levantando os principais conteúdos trabalhados no ensino de LFA nos livros didáticos e nas cinco melhores universidades brasileiras, nos cursos de Ciência da Computação e Sistemas de Informação (SYMONDS, 2022). A Etapa 2 responde a QP2 buscando ferramentas que auxiliam o ensino e aprendizagem de LFA. A Etapa 3 responde a QP3 mapeando de qual maneira as ferramentas obtidas na Etapa 2 contemplam os conteúdos obtidos na Etapa 1. Por fim, a Etapa 4 apresenta uma proposta de estratégia de uso das ferramentas obtidas na Etapa 2 no intuito de otimizar o ensino e aprendizagem dos conteúdos levantados na Etapa 1.

3.2.1 Etapa 1 - Levantar os principais conteúdos

A primeira etapa consiste em encontrar os conteúdos práticos mais recorrentes no ensino de LFA. Para isso é feita uma busca nas ementas das cinco melhores universidades do Brasil nos cursos de Ciência da Computação e Sistemas de Informação. Caso a universidade não contenha a disciplina com o nome de “Linguagens Formais e Autômatos”, é escolhida uma ementa que mais se aproxime, se existir. A seleção dessas universidades baseia-se no *QS World University Rankings* (SYMONDS, 2022), pois é um *ranking* já consolidado com existência desde 2004 e faz a classificação analisando quatro tipos de indicadores: Reputação do corpo docente, empregabilidade dos graduandos, citações em pesquisa e índice H que mede o impacto das publicações científicas. Este estudo desconsidera o indicador de empregabilidade dos graduandos e chegou às seguintes universidades organiza-

das em ordem decrescente de classificação: USP, UFMG, UNICAMP, UNESP e UFRGS.

Subsequente às ementas, são consultados nos principais livros de apoio à disciplina – especificamente Sudkamp (2007) e Hopcroft e Ullman (1969) – os capítulos que abordam esses conteúdos. Por fim, espera-se obter uma lista com os principais conteúdos no ensino de LFA.

3.2.2 Etapa 2 - Buscar ferramentas

A segunda etapa consiste em encontrar ferramentas para auxiliar o ensino e aprendizagem de LFA. Para isso, são consultados na literatura os artigos que as apresentam. A base de dados utilizada é o *Google Scholar* e são utilizadas as seguintes palavras chave: “*formal languages and automatas tools*”, “*formal languages tools*”, “*automata tools*” e “*linguagens formais e autômatos ferramentas*”. Os artigos que são levados em consideração são os que estão em língua portuguesa ou inglesa, tenham sido publicados entre os anos de 2015 e 2022, contemplem pelo menos 25% dos conteúdos obtidos na Etapa 1 e que proveem ferramentas para serem testadas. Por fim, espera-se obter uma lista com ferramentas que apoiam o ensino e aprendizagem na disciplina de LFA.

3.2.3 Etapa 3 - Analisar as ferramentas

A terceira etapa consiste em analisar as ferramentas obtidas na Etapa 2 e avaliar de que maneira apoiam os conteúdos levantados na Etapa 1. Para isso, primeiramente uma versão executável da ferramenta é obtida, caso não seja publicamente acessível pela Internet. Posteriormente, cada conteúdo é testado em cada uma das ferramentas. Para uma análise justa, os testes são sempre os mesmos. A análise busca entender qual a cobertura da ferramenta em relação aos conteúdos levantados.

Após isso, são analisadas as seguintes características de cada ferramenta: *fins didáticos*, i.e., ferramentas que possuem respostas passo a passo para melhor entender os algoritmos aplicados; *plataforma de uso*, i.e., em quais dispositivos a ferramenta pode ser utilizada: *mobile*, *desktop* ou *web*; *última atualização*, i.e., ano da última atualização; *exercícios disponíveis*, i.e., ferramentas que possuem exercícios disponíveis para os usuários. Por fim, espera-se obter um mapa de quais conteúdos cada ferramenta cobre.

3.2.4 Etapa 4 - Propor uma estratégia de uso para as ferramentas

A quarta e última etapa consiste em propor uma estratégia de uso das ferramentas a fim de otimizar o ensino e aprendizagem de LFA. Para isso, primeiramente é consultada a resposta obtida na Etapa 3 e sintetizada em três principais indicadores: *Cobertura de conteúdos*, separada em níveis da hierarquia; *Respostas passo a passo*; e *Exercícios disponíveis*. Esses indicadores são escolhidos de modo a focar em ferramentas completas e/ou que promovem um maior ambiente educacional.

Em seguida, as ferramentas são vistas em conjunto, a fim de analisar como cada ferramenta se complementa. Para isso, a análise é feita utilizando um diagrama de Venn com todas as ferramentas obtidas, onde cada conjunto é construído a partir dos conteúdos que cada ferramenta cobre. Com base nessas análises, é proposta uma estratégia de uso das ferramentas levando em consideração as necessidades específicas do público-alvo: professores e alunos.

Este capítulo organizou a pesquisa em quatro questões de pesquisa bem definidas as quais são devidamente abordadas e respondidas no próximo capítulo.

4 CONDUÇÃO DA PESQUISA

Este capítulo responde as questões de pesquisa apresentadas na Seção 3.1 utilizando os respectivos métodos descritos na Seção 3.2.

4.1 QP1: Quais conteúdos são mais recorrentes no contexto de ensino de LFA?

A Tabela 4.1 sintetiza os principais conteúdos trabalhados no ensino de LFA. Parte dos conteúdos foram extraídos das ementas (Apêndice A) e outra parte foram extraídos dos capítulos dos livros.

A Hierarquia de Chomsky é utilizada para facilitar a organização da tabela, pois as ementas mostram uma forte tendência de as universidades terem o conteúdo programático em ordem decrescente dos níveis hierárquicos. A coluna ID mapeia seguindo a expressão regular: $T[0-3][MIG][0-9]\{1,2\}$. Iniciada por T seguida do número 0, 1, 2 ou 3 representando o nível hierárquico, seguida de “M” ou “G” representando respectivamente Máquina ou Gramática e, por fim, um número identificador. As colunas seguintes apresentam o nome do conteúdo, a descrição e a marcação de ● caso a ementa da universidade aborde de maneira explícita ou ○ caso apenas dê indícios.

É possível observar que todas as universidades levadas em consideração no estudo abordam os conteúdos de LFA, porém a maior parte dos conteúdos não são tratados de forma explícita nas ementas.

Por fim, é possível observar que o maior foco de atividades práticas estão nos níveis 3 e 2 da hierarquia que correspondem até o meio do curso. Ainda, é possível notar que, no nível 3, o principal foco está relacionado com a construção de máquinas e o nível 2 mais voltado para construção de gramáticas.

Tabela 4.1 – Principais conteúdos práticos de LFA

ID	Nome	Descrição	USP	UFMG	UNICAMP	UNESP	UFRGS
T3M1	Operadores regulares	Executar operadores regulares em AFs	●	●	●	●	●
T3M2	AF \rightarrow ER	Converter AF para expressão regular	●	●	●	●	●
T3M3	ER \rightarrow AF	Converter expressão regular para AF	●	●	●	●	●
T3M4	AFND- $\lambda \rightarrow$ AFD	Converter AFND- λ para AFD	●	●	●	●	●
T3M5	AFs	Simular AFs	●	●	●	●	●
T3M6	Máquina de Mealy	Simular máquina de Mealy	●	●	●	●	●
T3M7	Máquina de Moore	Simular máquina de Moore	●	●	●	●	●
T3M8	Completar	Tornar total a função de transição do AF	●	●	●	●	●
T3M9	Minimização	Tornar o número de estados do AF mínimo	●	●	●	●	●
T3G1	Verificador de palavra	Verificar se a palavra pertence à gramática	●	●	●	●	●
T3G2	Identificador gramática	Identificar qual tipo de gramática	●	●	●	●	●
T2M1	AP	Simular APs	●	●	●	●	●
T2G1	Árvore de derivação	Exibir árvore de derivação de uma palavra	●	●	●	●	●
T2G2	Derivação mais à esquerda	Exibir derivação mais à esquerda de uma palavra	●	●	●	●	●
T2G3	Símbolo inicial não recursivo	Remover a recursividade do símbolo inicial	●	●	●	●	●
T2G4	Remoção de produções vazias	Remover produções vazias (λ) da gramática	●	●	●	●	●
T2G5	Remoção de regras de cadeia	Remover regras de cadeia do tipo $A \rightarrow B$	●	●	●	●	●
T2G6	Remoção de símbolos inúteis	Remover produções que não são alcançáveis ou que não geram símbolos terminais	●	●	●	●	●
T2G7	Remoção de recursão direta à esquerda	Remover recursão direta à esquerda	●	●	●	●	●
T2G8	Remoção de recursão direta e indireta à esquerda	Remover recursão direta e indireta à esquerda	●	●	●	●	●
T2G9	GLC \rightarrow AP	Converter GLC para AP	●	●	●	●	●
T2G10	FNC	Converter GLC para Forma Normal de Chomsky	●	●	●	●	●
T2G11	FNG	Converter GLC para Forma Normal de Greibach	●	●	●	●	●
T2G12	CYK	Executar o algoritmo de CYK	●	●	●	●	●
T2G13	Ambiguidade	Verificar se a gramática é ambígua	●	●	●	●	●
T1M1	ALL	Simular ALL	●	●	●	●	●
T0M1	MT	Simular MT	●	●	●	●	●
T0M2	Variantes MT	Simular variantes da MT	●	●	●	●	●

4.2 QP2: Quais as ferramentas disponíveis para o ensino e aprendizagem de LFA?

Por meio da metodologia descrita na Seção 3.2, esta seção descreve as ferramentas disponíveis para o ensino e aprendizagem de LFA.

1) Automata¹: Biblioteca Python que tem como intuito prover estruturas e algoritmos sobre AFs, APs e MTs. A última versão é a 7.1.0, lançada em 08/01/2023, disponível para ser instalada no *desktop* via gerenciador de pacotes do Python.

A ferramenta não possui interface gráfica, com isso, diferentemente das outras ferramentas descritas a seguir, todas as funcionalidades são executadas via

¹ <<https://github.com/caleb531/automata>>

código. O público alvo não aparenta ser estudantes de LFA e sim desenvolvedores que utilizam os algoritmos para outras finalidades. No entanto, possui uma documentação que apresenta todas as funções de forma clara e objetiva, facilitando o uso. A Figura 4.1 exemplifica a instalação via *Google Colab* e execução de um AF.

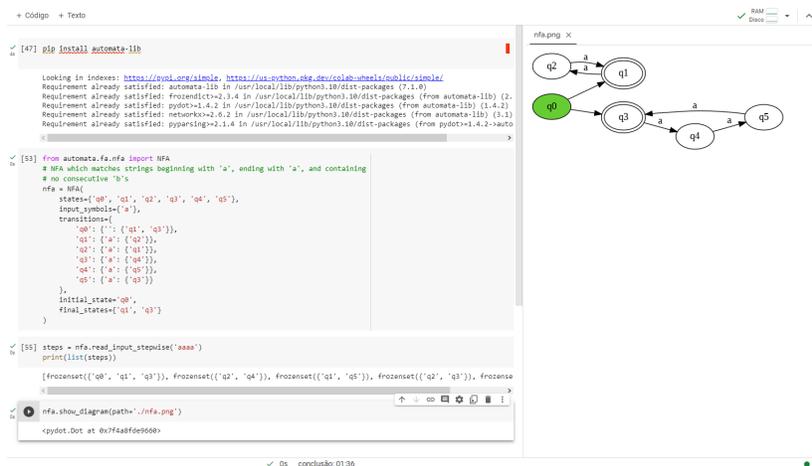


Figura 4.1 – Automata

2) Automata Tutor²: Ferramenta educacional voltada para auxiliar estudantes a aprenderem conceitos básicos de LFA e professores a proporem exercícios e acompanhar o desenvolvimento do aluno (D'ANTONI et al., 2020). A última versão é a 3.0, lançada em 29/05/2020, disponível para *web* e somente em inglês.

A ferramenta possui duas interfaces distintas: professor e aluno, ambas contam com um menu lateral com todos os conteúdos contemplados. A interface do professor consiste em criar salas e selecionar estudantes de sua preferência, gerar exercícios, acessar relatórios de nota de cada estudante e fazer download de exercícios. Já a interface do aluno consiste em resolver exercícios, que podem ser gerados pela própria ferramenta ou pelo professor. Além disso, a ferramenta apresenta um diferencial de *feedback* instantâneo para cada exercício resolvido. A Figura 4.2 exemplifica o ambiente em que o professor cria um exercício respecti-

² <<https://automata-tutor.model.in.tum.de>>

vamente de máquina e gramática e a Figura 4.3 exemplifica a resolução do aluno respectivamente de máquina e gramática em conjunto com o *feedback* instantâneo.

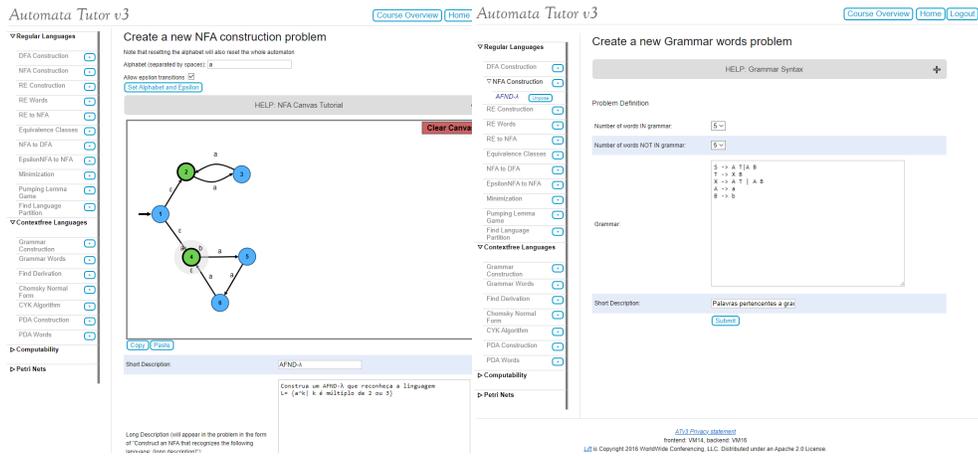


Figura 4.2 – Automata Tutor - Professor

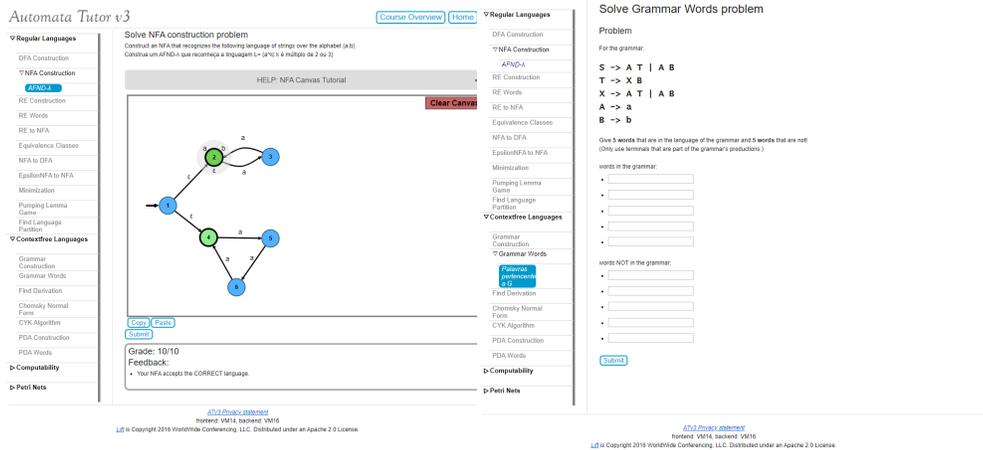


Figura 4.3 – Automata Tutor - Aluno

3) CMSimulator³: Ferramenta educacional com intuito de auxiliar alunos e professores no processo de ensino e aprendizagem de LFA (CHUDA; TRIZNA;

³ <<https://play.google.com/store/apps/details?id=fiitstu.gulis.cmsimulator>>

KRATKY, 2015). A última versão é a 5.2.0, lançada em 27/06/2020, disponível para dispositivos móveis somente em inglês.

CMSimulator é uma ferramenta com foco maior em testes. Apresenta duas telas distintas para manipulação de máquinas e gramáticas. Para cada uma das telas conta com uma funcionalidade de realização de múltiplos testes, sendo possível a entrada ser feita uma a uma ou por expressão regular. Além disso, também oferece suporte de salvar e carregar arquivos no formato XML e simular resposta passo a passo. O menu principal também conta com informações sobre exercícios, configuração e ajuda para o uso. Os exercícios só estão disponíveis para usuários logados. É importante mencionar que durante a realização da pesquisa, não foi possível ter acesso aos exercícios devido a um erro apresentado pela ferramenta durante as tentativas de login. Foi enviado um *e-mail* para a desenvolvedora, porém não houve retorno e, portanto, não foi possível avaliar a cobertura, complexidade e completude dos exercícios. A Figura 4.4 apresenta as três principais telas da ferramenta. O item (a) apresenta o menu principal, o item (b) apresenta a tela de manipulação de máquinas e o item (c) a tela de manipulação de gramática.

4) JFLAP⁴: Ferramenta desenvolvida na Universidade de Duke com intuito de auxiliar o ensino e aprendizagem de LFA (RODGER; FINLEY, 2006). A última versão é a 7.1, lançada em 27/08/2018, disponível para *desktop* somente em inglês.

JFLAP é a ferramenta mais completa entre os trabalhos analisados. Apresenta duas telas distintas para manipulação de máquinas e gramáticas. Para cada uma delas é possível salvar e carregar arquivos no formato XML, realizar múltiplos testes, simular respostas passo a passo e salvar imagem da tela. Além disso, a tela de máquinas possui dois recursos que não foram encontrados nas demais ferramentas: desfazer (*undo*) e refazer (*redo*), além de poder aplicar

⁴ <<https://www.jflap.org>>

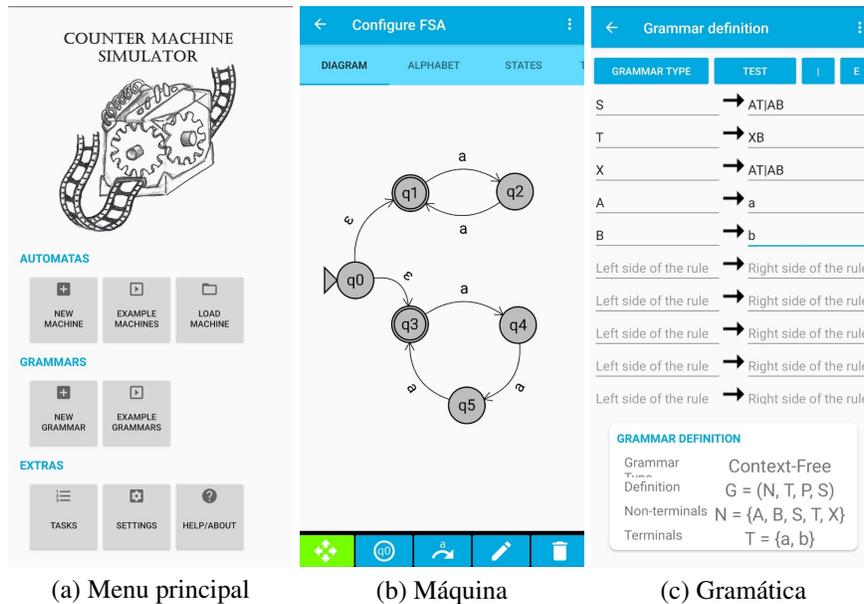


Figura 4.4 – CMSimulator

layouts pré-definidos para melhorar a visualização. Para auxiliar no uso, a ferramenta ainda conta com uma página *web* com tutoriais de uso, propostas de exercícios, artigos, história da ferramenta e dados estatísticos de uso. A Figura 4.5 apresenta as três principais telas da ferramenta. O item (a) apresenta o menu principal, o item (b) apresenta a tela de manipulação de máquinas e o item (c) a tela de manipulação de gramática.

5) LFApp⁵: Ferramenta voltada para estudantes praticarem e aprenderem conceitos de LFA (PEREIRA; TERRA, 2018). A última versão é a 1.11, lançada em 14/03/2018, disponível para dispositivos móveis, com suporte para português e inglês.

A ferramenta apresenta duas telas distintas para manipulação de máquinas e gramáticas, ambas contam com o suporte de exibir resposta passo a passo e mostrar o histórico de uso. Além disso possui o diferencial de exibir os

⁵ <<https://play.google.com/store/apps/details?id=org.ufla.lfapp>>

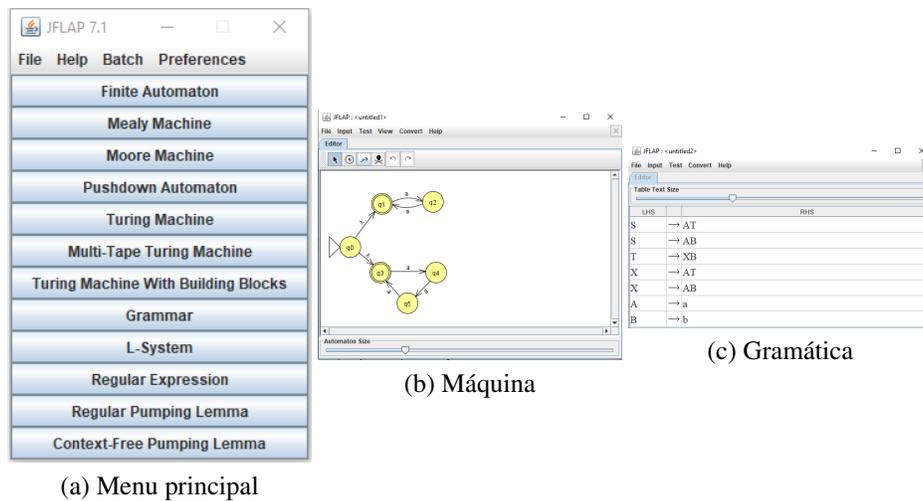


Figura 4.5 – JFLAP

algoritmos que foram aplicados a cada resolução, o que pode ser relevante no meio acadêmico. A Figura 4.6 apresenta as três principais telas da ferramenta. O item (a) apresenta o menu principal, o item (b) apresenta a tela de manipulação de máquinas e o item (c) a tela de manipulação de gramática.

6) LFAWeb⁶: Ferramenta educacional voltada para o ensino e aprendizagem de LFA. É uma evolução da ferramenta LFAApp porém voltada ao ambiente *web*. LFAweb tem o intuito de fornecer as mesmas funcionalidades da ferramenta predecessora porém em um ambiente mais abrangente. A última versão é a 2.4.0, lançada em 18/08/2022, com suporte para português e inglês.

LFAweb conta com um menu lateral disponível em todas as telas, que permite um fácil acesso aos recursos. A ferramenta apresenta duas telas distintas para manipulação de máquinas e gramáticas, as quais contam com o suporte de salvar e carregar arquivos (formato DOT para máquinas e TXT para gramáticas), exibir histórico de uso, simular respostas passo a passo e exibir o algoritmo aplicado a cada resolução. Além disso, na tela de máquinas, a ferramenta conta

⁶ <lfaweb.dcc.ufla.br/>

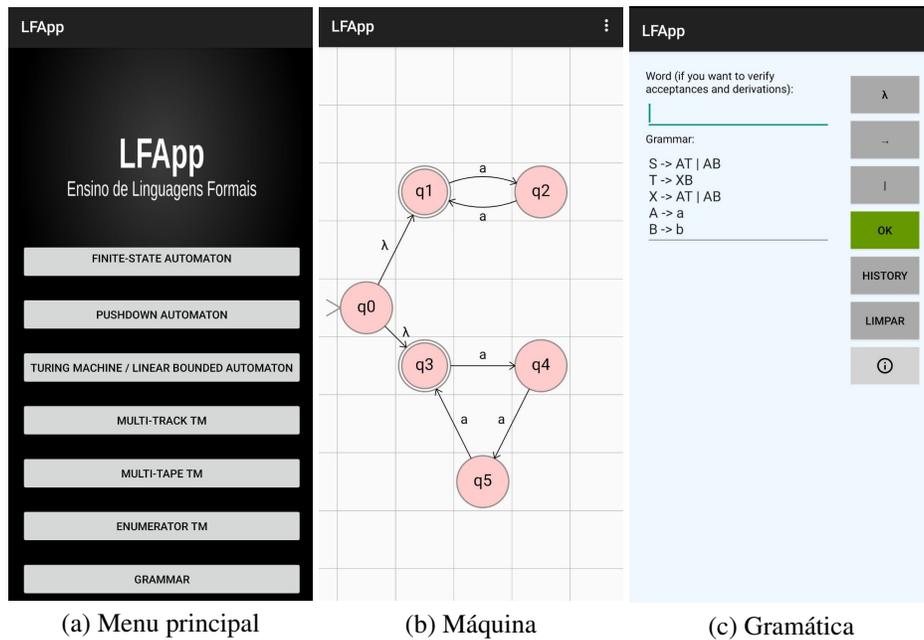


Figura 4.6 – LFAApp

com o recurso de aplicar *layout* pré definido para melhorar a visualização. A Figura 4.7 exemplifica as telas de manipulação.

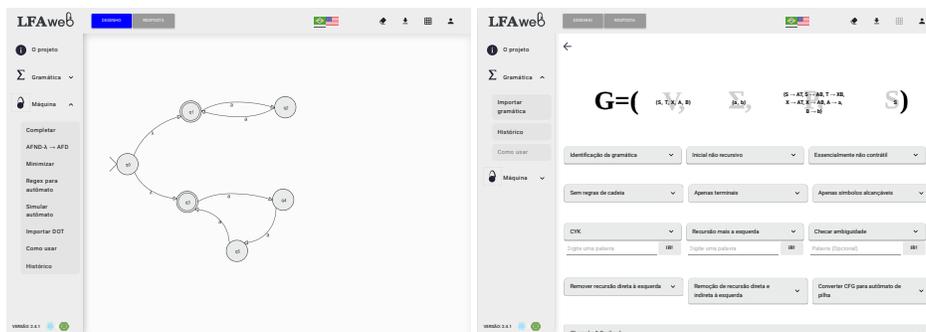


Figura 4.7 – LFAweb

Com base no exposto, as ferramentas Automata, LFAweb, Automata Tutor e JFLAP apresentam indícios de ainda receberem manutenção, enquanto CM-Simulator e LFAApp não apresentam. Automata possui uma recente atualização.

LFAweb é uma evolução da ferramenta LFApp e possui uma recente atualização. Automata Tutor, por sua vez, apresenta avisos de manutenção em seu *link* de uso. JFLAP, apesar de sua última atualização ser de 2018, vem recebendo manutenções desde 1996, o que demonstra sua robustez e continuidade no desenvolvimento. Por outro lado, CMSimulator apresenta indícios de não estar sendo mantida, já que não houve resposta para o e-mail enviado. LFApp, por sua vez, foi descontinuada.

Além disso, todas possuem suporte acadêmico. Quanto ao suporte a criação de exercícios, a ferramenta Automata Tutor fornece ao professor funcionalidades para criação, aplicação e correção de exercícios. As ferramentas CMSimulator, JFLAP, LFApp e LFAweb podem ser usadas pelo professor como ambiente para realização de atividades práticas, embora não tenham funcionalidades específicas para gerenciar a aplicação de exercícios. A página *web* do JFLAP, por exemplo, fornece sugestões de como estruturar exercícios com apoio da ferramenta. Quanto ao suporte de testes, CMSimulator e JFLAP se destacam por apresentarem funções que executam testes em múltiplas entradas ao mesmo tempo. Por fim, LFApp e LFAweb se destacam por exibir os algoritmos que foram executados a cada resolução, o que acredita-se ser relevante para o aprendizado acadêmico.

4.3 QP3: Como as ferramentas disponíveis apoiam os conteúdos de ensino de LFA?

A Tabela 4.2 sintetiza quais conteúdos são contemplados pelas ferramentas encontradas. A marcação ● representa que a ferramenta cobre totalmente o conteúdo e a ◐ que cobre parcialmente.

No nível 3 da hierarquia, a ênfase é dada principalmente aos conteúdos voltados para máquinas, e as ferramentas oferecem uma ampla cobertura, sendo que construção de AFs e transformação AFND- λ para AFD apresentam 100% de cobertura. Além disso, conteúdos como ER para AFD, Minimização, Verificador de palavra e Identificador de gramática estão disponíveis em 5 das 6 ferramentas.

No nível 2 da hierarquia, a ênfase é dada principalmente aos conteúdos voltados para gramáticas e LFApp e LFAweb se destacam por abranger completamente os conceitos de gramática enquanto JFLAP deixa de cobrir apenas o conteúdo de transformação para Forma Normal de Greibach. Por outro lado, Automata, Automata Tutor e CMSimulator apresentam uma cobertura limitada, sugerindo que essas ferramentas são mais voltadas para o estudo de máquinas.

No nível 1 da hierarquia, apenas um conteúdo é analisado e a ferramenta LFAweb é a única que não oferece cobertura desse conteúdo. No nível 0 da hierarquia, a ênfase é em MTs e suas variantes e Automata, JFLAP, LFApp se destacam oferecendo cobertura total.

Dado o exposto, é possível observar que JFLAP, LFApp e LFAweb são ferramentas completas que provêm funcionalidades tanto para máquinas quanto para gramáticas. Por um lado, JFLAP é a mais completa com cobertura de 96%. Por outro lado, as ferramentas Automata, Automata Tutor e CMSimulator mostram ter maior foco em máquinas, simulações e conteúdos básicos de gramáticas.

4.4 Proposta de uma estratégia de uso para as ferramentas

Com base nos resultados obtidos nas seções anteriores, esta seção propõe uma estratégia de uso das ferramentas a fim de otimizar o ensino e aprendizagem de LFA.

A proposta não tem como objetivo minimizar o número de ferramentas aplicadas no ensino, mas maximizar o apoio no ensino e aprendizagem da disciplina. As ferramentas CMSimulator e LFApp não são recomendadas devido à falta de atualizações e possíveis *bugs* que podem afetar o seu uso. Automata é uma ferramenta com foco em desenvolvedores, desse modo não é recomendada para o ensino aprendizagem, mas pode ser utilizada em outras áreas do conhecimento. Por outro lado, as ferramentas Automata Tutor, JFLAP e LFAweb compõem um

Tabela 4.2 – Comparativo das ferramentas

ID	Nome	Automata	Automata Tutor	CMSimulator	JFLAP	LFApp	LFAweb
T3M1	Operadores regulares	●			◐	◐	◐
T3M2	AF → ER	●			●		
T3M3	ER → AF	●	●		●	●	●
T3M4	AFND- λ → AFD	●	●	●	●	●	●
T3M5	AFs	●	●	●	●	●	●
T3M6	Máquina de Mealy				●		
T3M7	Máquina de Moore				●		
T3M8	Completar				●	●	●
T3M9	Minimização	●	●		●	●	●
T3G1	Verificador de palavra		●	●	●	●	●
T3G2	Identificador gramática		●	●	●	●	●
T2M1	AP	●	●	●	●	●	
T2G1	Árvore de derivação		●	●	●	●	●
T2G2	Derivação mais à esquerda		●		●	●	●
T2G3	Símbolo inicial não recursivo				●	●	●
T2G4	Remoção de produções vazias				●	●	●
T2G5	Remoção de regras de cadeia				●	●	●
T2G6	Remoção de símbolos inúteis				●	●	●
T2G7	Remoção de recursão direta à esquerda				●	●	●
T2G8	Remoção de recursão direta e indireta à esquerda				●	●	●
T2G9	GLC → AP				●	●	●
T2G10	FNC		●		●	●	●
T2G11	FNG					●	●
T2G12	CYK		●		●	●	●
T2G13	Ambiguidade				●	●	●
T1M1	ALL	●		●	●	●	
T0M1	MT	●		◐	●	●	
T0M2	Variantes MT	●			●	●	

grupo que abrange todos os conteúdos da disciplina e oferecem funcionalidades que podem ser adequadas para diferentes perfis de alunos e professores.

Para professores que ministram aulas com conteúdos práticos, as ferramentas JFLAP e LFAweb são as melhores escolhas. Apesar de necessitar a instalação, JFLAP possui um conjunto de funcionalidades que contemplam de maneira didática os conteúdos relacionados a simulações de AFs, APs, MTs e gramáticas.

Em relação às máquinas, além de exibir passo a passo o processamento da palavra, a ferramenta também exibe a árvore de execução. Em relação à gramática, a ferramenta exibe a derivação da palavra via árvore ou tabela. A Figura 4.8 apresenta as telas de simulação de máquinas e gramáticas. O item (a) apresenta a simulação de um AFND- λ , o item (b) exibe a árvore de derivação de uma gramática e o item (c) apresenta a tabela de derivação. No entanto, no que diz respeito a transformações de máquinas e gramáticas, JFLAP apenas reporta o resultado final.

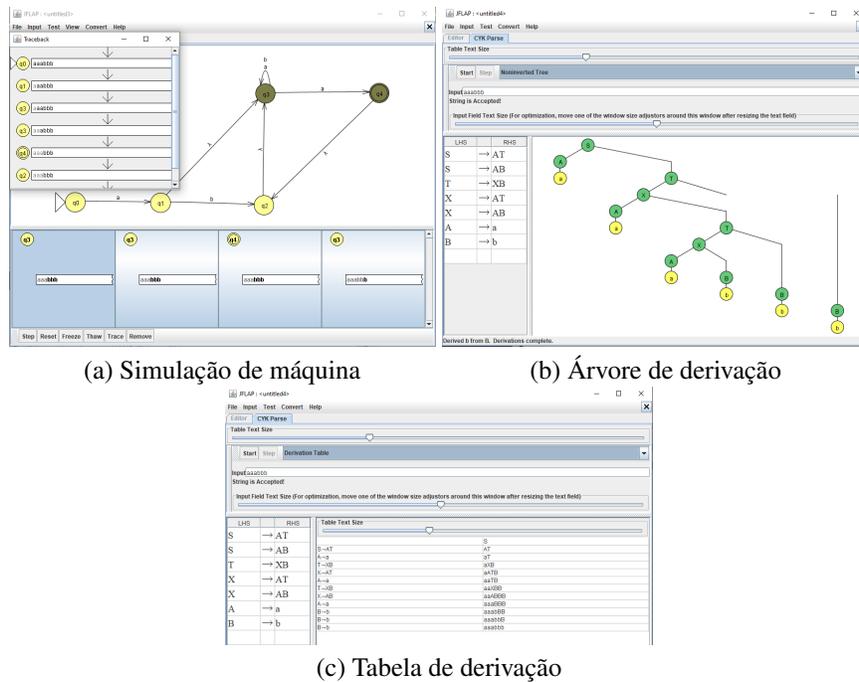


Figura 4.8 – Simulações utilizando o JFLAP

A ferramenta LFAweb, por outro lado, não se destaca pelas simulações de máquinas e gramáticas, mas sim, pelos algoritmos de transformação. Em relação às máquinas, os algoritmos de transformação exibem a solução passo a passo com o auxílio de uma tabela. LFAweb fornece uma tela com algoritmos de transformação gramatical, onde as soluções são feitas passo a passo. A Figura 4.9 apresenta três transformações. O item (a) exemplifica a minimização de um AF, o item (b)

a transformação AFND- λ para AFD e o item (c) os algoritmos de transformação para a Forma Normal de Chomsky.

(a) Minimização

Índice	Q1Q2	Q3Q4	Ítem(s)
(est. q0)	✓ → x	(1)	opop (est. q1q2)
(est. q1q0)	✓ → x	(est. q0)	a [est. q2]q0q1
(est. q2q0)	✓ → x	(1)	a [est. q2]q0q1
(est. q3)	✓ → x	(1)	a [est. q2]q0q1
(est. q1q0)	✓	(1)	
(est. q2q0)	✓	(1)	
(est. q3)	✓	(1)	
(est. q2q0)	✓	(1)	

(b) AFND- $\lambda \rightarrow$ AFD

AFND- λ	a	b	λ	Final- λ
q0	q0q1	(1)	(1)	q0q1
q1	(1)	q2q0	q0q1	q1, q0q1
q2	(1)	(1)	q0q1	q2, q0q1
q3	q0q1	q0q1	(1)	q0q1
q4	(1)	(1)	q2q0	q4, q1, q0q1

(c) FNC

Passo 1: $S \rightarrow S$
 $S \rightarrow aS \mid b \mid A$
 $A \rightarrow aA \mid A \mid C$
 $B \rightarrow a \mid b$
 $C \rightarrow C \mid B$
 $D \rightarrow C \mid B$

Passo 2: $S \rightarrow S$
 $S \rightarrow aS \mid b \mid A$
 $A \rightarrow aA \mid A \mid C$
 $B \rightarrow a \mid b$
 $C \rightarrow C \mid B$
 $D \rightarrow C \mid B$

Passo 3: $S \rightarrow aS \mid b \mid aA \mid A \mid C$
 $S \rightarrow aS \mid b \mid aA \mid A \mid C$
 $A \rightarrow aA \mid A \mid C \mid B$
 $B \rightarrow a \mid b$
 $C \rightarrow C \mid B$
 $D \rightarrow C \mid B$

Passo 4: $S \rightarrow aS \mid b \mid aA \mid A \mid C$
 $S \rightarrow aS \mid b \mid aA \mid A \mid C$
 $A \rightarrow aA \mid A \mid C \mid B$
 $B \rightarrow a \mid b$
 $C \rightarrow C \mid B$
 $D \rightarrow C \mid B$

Passo 5: $S \rightarrow aS \mid b \mid aA \mid A \mid C$
 $S \rightarrow aS \mid b \mid aA \mid A \mid C$
 $A \rightarrow aA \mid A \mid C \mid B$
 $B \rightarrow a \mid b$
 $C \rightarrow C \mid B$
 $D \rightarrow C \mid B$

Passo 6: $S \rightarrow aS \mid b \mid aA \mid A \mid C$
 $S \rightarrow aS \mid b \mid aA \mid A \mid C$
 $A \rightarrow aA \mid A \mid C \mid B$
 $B \rightarrow a \mid b$
 $C \rightarrow C \mid B$
 $D \rightarrow C \mid B$

Figura 4.9 – Transformações utilizando o LFAweb

Para professores que aplicam exercícios, Automata Tutor é a melhor ferramenta apesar de não contemplar todos os conteúdos. A ferramenta oferece suporte ao professor para gerenciar completamente a aplicação de exercícios. Apesar disso, JFLAP também pode ser usada para essa finalidade, já que conta com uma página *web* com sugestões de como estruturar exercícios com apoio da ferramenta. Além do mais, com relação às máquinas, é possível exportar e importar no formato XML e conta com a funcionalidade de equivalência de máquinas, que pode ser útil aos professores na correção dos exercícios.

O diagrama de Venn, ilustrado pela Figura 4.10, revela a importância da ferramenta na disciplina, uma vez que apenas o JFLAP contempla os conteúdos relacionados às máquinas de Mealy e Moore. Além disso, o diagrama também reporta que a união de JFLAP e LFAweb cobrem completamente 27 dos 28 con-

teúdos, excluindo apenas operadores regulares que é coberto parcialmente. É importante ressaltar que dos 27 conteúdos 19 são cobertos pelas duas ferramentas, 6 são cobertos exclusivamente pelo JFLAP, 1 exclusivamente pelo LFAweb e 1 coberto pelo JFLAP e Automata Tutor.

Diante desse cenário, este trabalho propõe a utilização contínua das ferramentas JFLAP e LFAweb ao longo do curso, oferecendo flexibilidade para utilizá-las de forma independente ou combinada. Além disso, para complementar essa abordagem, sugere-se o uso da ferramenta Automata Tutor para a aplicação e gerenciamento de exercícios.

Para auxiliar a visualização, a Tabela 4.3 indica quais as ferramentas que são recomendadas para abordar cada um dos conteúdos. É importante ressaltar que existem conteúdos em que as três ferramentas são recomendadas. Isso ocorre devido ao fato de que as ferramentas apresentam características complementares na abordagem de um mesmo conteúdo. Por exemplo, a ferramenta Automata Tutor é recomendada para todos os conteúdos que oferece cobertura, pois provê o gerenciamento completo para a aplicação de exercícios.

Embora não seja necessário indicar o raciocínio que guiou a recomendação de ferramentas para os conteúdos que são abordados de maneira semelhante por JFLAP e LFAweb ou são abordados de forma exclusiva por uma ferramenta, segue o raciocínio para cada um dos demais conteúdos:

T3M2: JFLAP é recomendada, pois apresenta a resposta passo a passo, além de permitir ao usuário interagir com a construção da resposta.

T3M3: Apesar de LFAweb contemplar o conteúdo, é recomendada a escolha de JFLAP, dado que o usuário pode interagir com a construção da resposta.

T3M4: LFAweb é recomendada, pois apresenta a resposta passo a passo e são utilizadas tabelas que auxiliam na compreensão da resposta.

T3M5: JFLAP é recomendada, pois apresenta recursos de exibir *stack trace* dos estados, equivalência de máquinas, resposta passo a passo e múltiplos testes.

T3M9: LFAweb é recomendada, pois apresenta a resposta passo a passo e é utilizada uma tabela que auxilia na compreensão da resposta.

T2G[3-6]: São recursos que compõem os passos da FNC. LFAweb é recomendada, pois exibe a resposta passo a passo e o algoritmo aplicado, enquanto JFLAP não é possível executar as funções separadamente.

T2G[7-8]: LFAweb é recomendada, pois exibe a resposta passo a passo e algoritmo aplicado.

T2G9: JFLAP é recomendada, pois exibe a resposta passo a passo.

T2G10: JFLAP e LFAweb são recomendadas, JFLAP aborda de uma maneira com que o usuário tem a possibilidade de interagir com a construção da resposta, enquanto LFAweb exibe a resposta passo a passo e exibe o algoritmo que foi aplicado.

T2G12: LFAweb é recomendada, pois exibe a resposta passo a passo utilizando a construção de uma tabela para auxiliar.

T2G13: LFAweb é recomendada, pois a resposta é direta e exibe as árvores de derivação que geram ambiguidade.

Considerando as recomendações, é importante que as ferramentas sejam apresentadas de forma atrativa, destacando as funcionalidades, fornecendo suporte e exemplos práticos de sua aplicação na disciplina. Além disso, faz-se relevante integrar as ferramentas ao currículo, desenvolvendo atividades e projetos que exijam sua utilização. Essa combinação estratégica de recursos pode oferecer um ambiente de aprendizado abrangente que pode potencializar o progresso dos estudantes ao longo do curso da disciplina.

Tabela 4.3 – Recomendação de uso da ferramenta por conteúdo

ID	Nome	Automata Tutor	JFLAP	LFAweb
T3M1	Operadores regulares		✓	✓
T3M2	AF \rightarrow ER		✓	
T3M3	ER \rightarrow AF	✓	✓	
T3M4	AFND- λ \rightarrow AFD	✓		✓
T3M5	AFs	✓	✓	
T3M6	Máquina de Mealy		✓	
T3M7	Máquina de Moore		✓	
T3M8	Completar		✓	✓
T3M9	Minimização	✓		✓
T3G1	Verificador de palavra	✓	✓	✓
T3G2	Identificador gramática	✓	✓	✓
T2M1	AP	✓	✓	
T2G1	Árvore de derivação	✓	✓	
T2G2	Derivação mais a esquerda	✓		✓
T2G3	Símbolo inicial não recursivo			✓
T2G4	Remoção de produções vazias			✓
T2G5	Remoção de regras de cadeia			✓
T2G6	Remoção de produções inúteis			✓
T2G7	Remoção de recursão direta à esquerda			✓
T2G8	Remoção de recursão direta e indireta à esquerda			✓
T2G9	GLC \rightarrow AP		✓	
T2G10	FNC	✓	✓	✓
T2G11	FNG			✓
T2G12	CYK	✓	✓	✓
T2G13	Ambiguidade			✓
T1M1	ALL		✓	
T0M1	MT		✓	
T0M2	Variantes MT		✓	

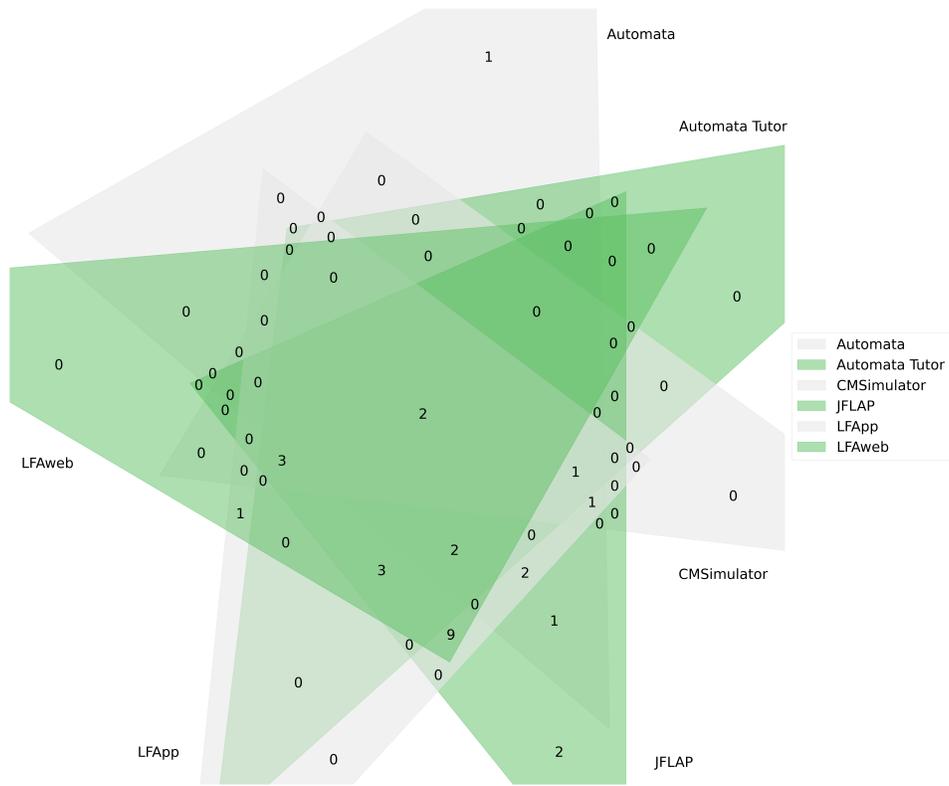


Figura 4.10 – Diagrama de Venn ilustrando a distribuição da quantidade de conteúdos contemplados pelas ferramentas

5 CONCLUSÃO

A disciplina de LFA desempenha um papel fundamental na formação acadêmica de estudantes de cursos superiores da área de computação. No entanto, os conteúdos são considerados clássicos e desafiadores, o que reduz o interesse e a motivação dos alunos. Nesse sentido, ferramentas educacionais podem ser utilizadas de forma a aprimorar o processo de ensino e aprendizagem em LFA. Diante desse cenário, este trabalho de conclusão de curso propõe uma estratégia de uso das ferramentas com o objetivo de otimizar o ensino e aprendizagem de LFA.

A condução desta pesquisa se baseia em três questões de pesquisa (QPs). A QP1 levantou 28 conteúdos nas universidades USP, UFMG, UNICAMP, UNESP e UFRGS, que foram organizados nos níveis da Hierarquia de Chomsky, sendo que os níveis 3 (LR) e 2 (LLC) concentram a maior parte dos conteúdos práticos. O nível 3 possui um enfoque em máquinas, enquanto o nível 2 foca principalmente gramáticas.

A QP2 obteve seis ferramentas que apoiam LFA: Automata, Automata Tutor, CMSimulator, JFLAP, LFAApp e LFAweb. É importante mencionar que LFAApp foi descontinuado e CMSimulator apresentou indícios de não receber atualizações. Mais importante, foi possível constatar que todas as ferramentas possuem suporte acadêmico, salvo Automata.

A QP3 evidenciou que apenas os conteúdos de construção de AFs e transformação AFND- λ para AFD possuem cobertura de todas as ferramentas. JFLAP e LFAweb se mostram como ferramentas completas. JFLAP apresenta a maior cobertura de conteúdos, cobrindo 27 em 28 e LFAweb 23 em 28.

Por fim, este estudo sugere que a integração das ferramentas Automata Tutor, JFLAP e LFAweb pode ser uma abordagem promissora para otimizar o processo de ensino e aprendizagem da disciplina. Automata Tutor é recomendada para aplicação de exercícios, especialmente para LR. O uso de JFLAP e LFAweb

é recomendado durante todo o curso da disciplina, mais precisamente, JFLAP no contexto de simulações e LFAweb em transformações.

Considerando os resultados parciais obtidos, sugere-se que o desenvolvimento de uma nova ferramenta siga os seguintes três requisitos: (i) tenha a amplitude de cobertura de conteúdos como JFLAP; (ii) tenha um ambiente de gerenciamento de exercícios como Automata Tutor; e (iii) seja uma plataforma *web* responsiva, garantindo portabilidade, assim como Automata Tutor e LFAweb. Esses requisitos são considerados elementos chave para assegurar a robustez da ferramenta no contexto educacional de LFA.

As principais contribuições deste trabalho são: (i) proposta de uma estratégia de uso de ferramentas com o objetivo de otimizar o ensino e aprendizagem de LFA e (ii) uma análise de conteúdos e ferramentas relacionadas a LFA, que pode servir de roteiro para a construção de novas ferramentas ou atualização de ferramentas já existentes.

Trabalhos futuros incluem: (i) analisar a usabilidade das ferramentas encontradas; (ii) avaliar o quanto as ferramentas contribuem para a aprendizagem sob a ótica educacional; (iii) propor um projeto de *frontend* utilizando a biblioteca Automata; (iv) estudar os motivos que causam a ausência de ferramentas atualizadas para dispositivos móveis e (v) replicar o estudo, porém com base em ementas de universidades de fora do Brasil.

REFERÊNCIAS

- CASCINI, D.; CAMPOS, S. Avaliação de jogos educacionais multiusuários: Uma revisão sistemática da literatura. In: **XXVI Simpósio Brasileiro de Informática na Educação (SBIE)**. [S.l.: s.n.], 2015. p. 519.
- CHESNEVAR, C. I.; GONZÁLEZ, M. P.; MAGUITMAN, A. G. Didactic strategies for promoting significant learning in formal languages and automata theory. **ACM SIGCSE Bulletin**, v. 36, n. 3, p. 7–11, 2004.
- CHUDA, D.; TRIZNA, J.; KRATKY, P. Android automata simulator. In: **9th International Conference on e-Learning (ICEEL)**. [S.l.: s.n.], 2015. p. 80–84.
- DOGNINI, M. J.; RAABE, A. L. A. Eduling-software educacional para linguagens regulares. In: **XIV Simpósio Brasileiro de Informática na Educação (SBIE)**. [S.l.: s.n.], 2003. p. 216–225.
- D'ANTONI, L. et al. Automata tutor v3. In: **32nd International Conference on Computer-Aided Verification (CAV)**. [S.l.: s.n.], 2020. p. 3–14.
- HOPCROFT, J. E.; ULLMAN, J. D. **Formal languages and their relation to automata**. [S.l.]: Addison-Wesley Longman Publishing, 1969.
- PEREIRA, C. H.; TERRA, R. A mobile app for teaching formal languages and automata. **Computer Applications in Engineering Education**, v. 26, n. 5, p. 1742–1752, 2018.
- RAMOS, M. V. M. Ensino de linguagens formais e autômatos em cursos superiores de computação. **Revista de Computação e Tecnologia (ReCeT)**, v. 1, n. 1, p. 22–34, 2009.
- RODGER, S. H.; FINLEY, T. W. **JFLAP: an interactive formal languages and automata package**. [S.l.]: Jones & Bartlett Learning, 2006.
- SUDKAMP, T. A. **Languages And Machines: An Introduction to the theory of Computer Science**. 3. ed. [S.l.]: Pearson Education, 2007.
- SYMONDS, Q. Q. **QS World University Rankings by Subject 2022: Computer Science and Information Systems**. 2022. Disponível em: <<https://www.topuniversities.com/university-rankings/university-subject-rankings/2022/computer-science-information-systems>>. Acesso em: 09 jan. 2023.
- TERRA, R. **Dados estatísticos da disciplina de Linguagens Formais e Autômatos. Technical report, Universidade Federal de Lavras (UFLA)**. 2023.
- UFMG. **Ementa - Fundamentos da Teoria da Computação**. 2023. Disponível em: <<https://ufmg.br/cursos/graduacao/2377/91205/60493>>. Acesso em: 15 fev. 2023.

UFRGS. **Grade curricular Ciência da Computação**. 2023. Disponível em: <http://www.ufrgs.br/ufrgs/ensino/graduacao/cursos/exibeCurso?cod_curso=305>. Acesso em: 15 fev. 2023.

UNESP. **BCC - Estrutura Curricular**. 2023. Disponível em: <<https://www.fc.unesp.br/#!/departamentos/computacao/cursos-de-graduao/bacharelado-em-ciencia-da-computacao/estrutura-curricular/>>. Acesso em: 15 fev. 2023.

UNICAMP. **Catálogo dos Cursos de Graduação**. 2023. Disponível em: <<https://www.dac.unicamp.br/sistemas/catalogos/grad/catalogo2021/disciplinas/mc.html#disc-mc322>>. Acesso em: 15 fev. 2023.

USP. **Ementa - Teoria da Computação e Linguagens Formais**. 2023. Disponível em: <<https://uspdigital.usp.br/jupiterweb/obterDisciplina?sgldis=SCC0205&verdis=3>>. Acesso em: 15 fev. 2023.

VIJAYALASKHMI, M.; KARIBASAPPA, K. Activity based teaching learning in formal languages and automata theory-an experience. In: **2nd International Conference on Engineering Education: Innovative Practices and Future Trends (AICERA)**. [S.l.: s.n.], 2012. p. 1–5.

A APÊNDICE

Este apêndice apresenta as ementas da disciplina de LFA ou disciplinas similares oferecidas pelas universidades brasileiras. Os conteúdos das ementas foram retiradas nos sites oficiais das instituições.

USP: A ementa selecionada é “*Teoria da Computação e Linguagens Formais*”, como a seguir:

“Linguagens Regulares: Autômatos finitos determinísticos e não determinísticos; expressões regulares; técnicas para identificar e descrever linguagens regulares; técnicas para mostrar que uma linguagem não é regular; propriedades de tais linguagens. Linguagens Livres de Contexto: Gramáticas Livres de Contexto; derivações; árvores de derivação; ambiguidade; autômatos de pilha; propriedades de tais linguagens; técnicas para mostrar que uma linguagem não é livre de contexto. Linguagens Sensíveis ao Contexto e Linguagens Recursivamente Enumeráveis: Máquinas de Turing; definições básicas e sua relação com a noção de um algoritmo/programa. Poder das Máquinas de Turing e Tese de Church-Turing. Indecidibilidade: Máquinas de Turing Universais; Limitações sobre a nossa habilidade de computar; problemas indecidíveis. Teoria de Complexidade: Complexidade de Tempo, Complexidade de Espaço, Intratabilidade”. (USP, 2023)

UFMG: A ementa selecionada é “*Fundamentos da Teoria da Computação*”, como a seguir:

“Conjuntos e linguagens formais. Linguagens regulares: autômatos finitos determinísticos, autômatos finitos não-determinísticos, expressões regulares, e sua equivalência. Linguagens livres-do-contexto: autômatos com pilha, gramáticas livres-do-contexto, e sua relação. Máquinas de Turing determinísticas e não-determinísticas como modelo universal de computação. Hierarquia de Chomsky. Com-

putabilidade. Teorema de Rice. Exemplos de funções indecidíveis. Implicações de indecidibilidade". (UFMG, 2023).

UNICAMP: A ementa selecionada é "*Linguagens Formais e Autômatos*", como a seguir:

"Revisão de conceitos básicos; alfabetos e linguagens; linguagens regulares; linguagens livres de contexto; linguagens recursivas e linguagens recursivamente enumeráveis; computabilidade e decidibilidade". (UNICAMP, 2023)

UNESP: A ementa selecionada é "*Teoria da Computação e Linguagens Formais*", como a seguir:

"Introdução e conceitos básicos. Autômatos finitos. Linguagens e gramáticas regulares. Linguagem livre de contexto (LLC). Autômatos a Pilha. Máquinas de Turing (MT). Hierarquia das Linguagens Formais. Limites da computação algorítmica". (UNESP, 2023)

UFRGS: A ementa selecionada é "*Linguagens Formais e Autômatos*", como a seguir:

"Alfabetos e Linguagens. Linguagens, gramáticas e expressões regulares, autômatos finitos. Linguagens e gramáticas livres de contexto e autômatos de pilha. Linguagens sensíveis ao contexto". (UFRGS, 2023)