

NADIA - Natural Deduction proof Assistant

Davi Romero de Vasconcelos¹, Robson Teixeira Paula¹, Maria Viviane Menezes¹

¹Universidade Federal do Ceará (UFC) – Campus de Quixadá
Av. José de Freitas Queiroz, 5003 – Cedro – 63902-580 – Quixadá – CE – Brazil

{daviromero, vivianemenezes}@ufc.br, robson.teixeira2000@gmail.com

Abstract. *Logic in Computer Science course is part of most Information and Communication Technology courses. The Natural Deduction system is widely used for teaching proofs and appears in many Logic textbooks. This work presents a proof assistant, NADIA (Natural Deduction Proof Assistant), for Natural Deduction system, in the Fitch (boxes) style, in order to assist in the teaching-learning of undergraduate and graduate students. NADIA allows students to write their proofs as closely as possible to the proofs they take on paper. NADIA automatically checks whether the proof is correct and, if not, displays any errors found. To assess students' experience in using NADIA, we performed an evaluation of the tool in five courses that were offered in 2021.1 and 2021.2.*

Resumo. *A disciplina de Lógica para Computação faz parte da maioria dos cursos de Tecnologia da Informação e Comunicação. O sistema de Dedução Natural é amplamente utilizado para o ensino de demonstrações e este conteúdo consta em muitos dos livros-texto de Lógica. Este trabalho apresenta um assistente de provas, NADIA (Natural Deduction Proof Assistant), para o sistema de Dedução Natural em Lógica Proposicional, no estilo de Fitch (caixas), com a finalidade de auxiliar no ensino-aprendizagem de estudantes de graduação e pós-graduação. NADIA permite que os estudantes escrevam suas demonstrações de forma mais próxima possível das provas que realizam no papel. NADIA verifica automaticamente se a demonstração está correta e, caso contrário, exibe os erros encontrados. Para avaliar a experiência dos estudantes no uso do NADIA realizamos uma avaliação da ferramenta com alunos de cinco turmas da disciplina de Lógica para Computação que foram ofertadas em 2021.1 e 2021.2.*

1. Introdução

No início do século passado, correntes filosóficas surgiram objetivando a fundamentação da matemática em bases sólidas utilizando a lógica para tal [da Costa 2008]. A escola formalista de Hilbert, por exemplo, buscava a prova da auto-consistência da matemática. Em particular, ela buscava solucionar este problema de uma forma automatizada através de um procedimento (mecânico) que fosse capaz de verificar a consistência das proposições matemáticas. Contudo, apenas com o desenvolvimento dos computadores e com a automatização do raciocínio em Inteligência Artificial na década de 60 é que métodos automatizáveis de prova mais eficientes foram desenvolvidos [Chang and Lee 1976].

Como a escola formalista buscava a prova da auto-consistência da matemática, provas em matemática passaram a ser objetos de estudo da própria matemática dando início a uma área conhecida como *Teoria da Prova*. Um dos conceitos centrais desta

teoria é o de *sistema dedutivo* que pode ser entendido como um mecanismo que permite a construção de argumentos formais, estabelecendo conclusões a partir de premissas.

O sistema dedutivo de Dedução Natural surgiu neste contexto [Gentzen 1969], buscando uma melhor análise da estrutura das provas lógicas, e é uma ferramenta essencial para a Teoria da Prova, pois permite um maior entendimento da noção de prova e do papel dedutivo das constantes lógicas, individualizadas pelas regras de introdução e eliminação. Nos sistemas de Dedução Natural, as inferências são quebradas em passos atômicos de tal forma que cada passo envolve uma constante lógica diferente. Tais passos são de dois tipos: aqueles que permitem a introdução dos conectivos lógicos; e, os que eliminam tais conectivos [Gentzen 1969]. As provas iniciam a partir de premissas que são assumidas ou de suposições que são necessariamente descartadas no decorrer da prova. As derivações são, em geral, apresentadas na forma de árvores, também conhecido como estilo de Gentzen, ou por meio de caixas, denominado de estilo de Fitch.

O sistema de Dedução Natural é amplamente utilizado para o ensino de demonstrações e consta em muitos dos livros-texto de Lógica [van Dalen 1994, Huth and Ryan 2004]. O artigo [Pelletier 1999] apresenta uma breve história de Dedução Natural, traçando o desenvolvimento da Dedução Natural do ponto de vista que esses fundadores a abraçaram à ampla aceitação do método na década de 1960. O referido artigo apresenta as diferentes escolhas feitas por escritores de livros-texto com o objetivo de determinar quais são as “características essenciais” da natureza da dedução.

Além de ser a base do entendimento de provas matemáticas, a lógica tem um papel de relevo na fundamentação da computação e possui um vasto espectro de aplicações desde a especificação e verificação de sistemas a aplicações em áreas como banco de dados, inteligência artificial, engenharia de software, dentre outras. No campus da Universidade Federal do Ceará em Quixadá, a disciplina Lógica para Computação faz parte dos currículos dos cursos de Sistemas de Informação, Engenharia de Software, Ciência da Computação e Engenharia de Computação como componentes obrigatórios. A disciplina possui um alto índice de reprovação. Para uma melhor assimilação dos conteúdos, é fundamental que os estudantes exercitem e que tenham *feedback* de suas demonstrações.

Nesse contexto, para auxiliar o aprendizado dos alunos em demonstração em Dedução Natural, *propomos* um assistente de provas, denominado *Natural Deduction Proof Assistant* (NADIA), que é uma ferramenta computacional que nos permite verificar se uma demonstração está correta ou não. Caso a demonstração não esteja correta, o assistente apresenta os erros encontrados na demonstração. O NADIA foi desenvolvido em Python, tem uma versão *Web* e pode ser integrada a plataforma Moodle.

Este trabalho tem como objetivo apresentar a ferramenta desenvolvida; o contexto em que ela foi utilizada como ferramenta auxiliar no processo de ensino-aprendizagem de Dedução Natural em Lógica Proposicional; e, os resultados obtidos com a avaliação do sistema realizada com 142 alunos matriculados em cinco turmas da disciplina de Lógica para Computação durante o ano letivo de 2021. A versão atual do NADIA permite a demonstração de provas em Lógica de Primeira-Ordem. Todavia, como as mesmas não foram utilizadas no período de avaliação da ferramenta, não iremos abordá-la neste artigo.

O restante desse trabalho está organizado como segue. A Seção 2 apresenta o Sistema de Dedução Natural. Na Seção 3, mostramos o Assistente de Provas. A Seção 4

traz os trabalhos relacionados. A Seção 5 aponta os resultados da avaliação da experiência do uso do NADIA. E, na Seção 6 apresentamos as conclusões e trabalhos futuros.

2. Fundamentação Teórica

O objetivo do estudo da lógica nos cursos de Tecnologia da Informação e Comunicação é desenvolver linguagens para modelar as situações do mundo e dos sistemas, de modo que possamos analisá-las formalmente, construindo argumentos sobre elas para serem apresentados e justificados rigorosamente. Nesta seção, apresentamos os conceitos da linguagem e do sistema de Dedução Natural da Lógica Proposicional.

2.1. A Linguagem da Lógica Proposicional

A Lógica Proposicional se baseia em *proposições* ou *frases declarativas* que se pode argumentar sobre sua veracidade ou falsidade. As frases declarativas são consideradas como *atômicas* (e.g., “comprei um bilhete”, “ganhei na loteria”) de certa forma que podemos atribuir símbolos distintos A, B, C, \dots a cada uma destas frases (e.g., A : “comprei um bilhete”, B : “ganhei na loteria”).

Para codificar frases mais complexas usamos os conectivos lógicos: \neg (negação), \wedge (conjunção), \vee (disjunção) e \rightarrow (implicação) [Huth and Ryan 2004]. Por exemplo: $A \wedge B$ codifica a frase “comprei um bilhete e ganhei na loteria”; $A \vee B$ codifica a frase “comprei um bilhete ou ganhei na loteria.”; $B \rightarrow A$ representa “se ganhei na loteria então comprei um bilhete.” e; $\neg B$: “não ganhei na loteria”.

O conjunto dos átomos proposicionais, juntamente com os conectivos e os símbolos ‘(, ,)’ formam o alfabeto da linguagem da Lógica Proposicional. A Definição 1 descreve quais as combinações de símbolos do alfabeto são consideradas fórmulas.

Definição 1 A linguagem da Lógica Proposicional pode ser definida por uma gramática na forma de Backus Naur (BNF - Backus Naur Form) como a seguir:

$$\varphi ::= \perp \mid A \mid (\neg\varphi) \mid (\varphi \wedge \varphi) \mid (\varphi \vee \varphi) \mid (\varphi \rightarrow \varphi)$$

em que \perp e A representam, respectivamente, a contradição e qualquer proposição atômica e cada ocorrência φ a direita de direita de ‘ $::=$ ’ representa qualquer fórmula já construída.

2.2. Dedução Natural em Lógica Proposicional

O sistema de Dedução Natural é um mecanismo que permite a construção de uma prova formal, estabelecendo uma conclusão φ a partir de um conjunto de premissas $\Gamma = \{\varphi_1, \varphi_2, \dots, \varphi_n\}$, denotado por $\Gamma \vdash \varphi$, aplicando-se sucessivamente *regras de demonstração*. O artigo [Pelletier 2000] compara o estilo de demonstrações de Dedução Natural de 33 livros-texto, sendo o estilo *Fitch* (caixas) o adotado na maioria deles.

Neste estilo, as demonstrações são apresentadas de forma linear e sequencial, na qual cada uma das linhas da prova é numerada, tem uma afirmação e uma justificativa. As justificativas são definidas por serem *premissas* da prova ou pela aplicação de uma das *regras do sistema dedutivo*. As subprovas dentro de uma prova maior têm *caixas* ao redor e servem para delimitar o escopo de hipóteses temporárias. Provas podem ter caixas dentro de caixas, ou pode-se abrir outras caixas depois de fechar outras, obedecendo as

regras de demonstração. Uma fórmula só pode ser utilizada em uma prova em um determinado ponto se essa fórmula aconteceu anteriormente e se nenhuma caixa que contenha essa ocorrência da fórmula tenha sido fechada.

Uma demonstração em Dedução Natural é construída a partir da enumeração das premissas e da aplicação das *regras de Dedução Natural*. Em geral, cada conectivo possui uma regra para adicionar e outra regra para eliminar este conectivo. A construção de uma demonstração é um exercício criativo. Não é óbvio que regras aplicar e em qual ordem para, a partir das premissas, obter a conclusão desejada. A seguir, serão apresentadas as regras de Dedução Natural.

Premissas O primeiro passo em uma demonstração em Dedução Natural no estilo *Fitch* é enumerar as premissas da prova. A Figura 1 apresenta a estrutura geral da regra das premissas, na qual $\varphi_1, \varphi_2, \dots, \varphi_n$ são representadas em uma linha cada, seguindo uma numeração sequencial e como justificativa “premissa”.

1.	φ_1	premissa
2.	φ_2	premissa
⋮	⋮	⋮
n.	φ_n	premissa
⋮	⋮	⋮

Figura 1. Enumeração das premissas.

Conjunção Introdução ($\wedge i$) A regra da introdução da conjunção ($\wedge i$) é apresentada na Figura 2a (ou Figura 2b), na qual a fórmula $\varphi \wedge \psi$ pode ser concluída em uma linha p se φ e ψ foram demonstradas nas linhas m (ou n) e n (ou m), respectivamente, anteriores a linha p e que não foram descartadas. A Figura 2c exibe a aplicação $\wedge i$ da fórmula $A \wedge B$ na linha 3 a partir das fórmulas A e B , definidas nas linhas 1 e 2, respectivamente, que são anteriores a linha 3.

⋮	⋮	⋮	⋮	⋮	⋮	
$m.$	φ		$m.$	ψ		1. A premissa
⋮	⋮	⋮	⋮	⋮	⋮	2. B premissa
$n.$	ψ		$n.$	φ		3. $A \wedge B$ $\wedge i$ 1,2
⋮	⋮	⋮	⋮	⋮	⋮	
$p.$	$\varphi \wedge \psi$	$\wedge i$ m, n	$p.$	$\varphi \wedge \psi$	$\wedge i$ m, n	(c) $A, B \vdash A \wedge B$
	(a) Regra $\wedge i$			(b) Regra $\wedge i$		

Figura 2. Conjunção Introdução ($\wedge i$)

Conjunção Eliminação ($\wedge e$) A regra da eliminação da conjunção ($\wedge e$) é apresentada na Figura 3a (ou Figura 3b), na qual a fórmula φ (ou ψ) pode ser concluída na linha p a partir da eliminação à esquerda (ou à direita) da conjunção da fórmula $\varphi \wedge \psi$ da linha n (anterior a m e não foi descartada). A Figura 3c exibe uma aplicação da regra na qual A é obtida na linha 3 pela eliminação da conjunção à esquerda da fórmula $A \wedge B$ da linha 1.

$\begin{array}{c} \vdots \\ m. \quad \varphi \wedge \psi \\ \vdots \\ p. \quad \varphi \quad \wedge e \ m \end{array}$	$\begin{array}{c} \vdots \\ m. \quad \varphi \wedge \psi \\ \vdots \\ p. \quad \psi \quad \wedge e \ m \end{array}$	<ol style="list-style-type: none"> 1. $A \wedge B$ premissa 2. C premissa 3. A $\wedge e$ 1 4. $A \wedge C$ $\wedge i$ 3,2
(a) Regra $\wedge e$ à esquerda	(b) Regra $\wedge e$ à direita	(c) $A \wedge B, C \vdash A \wedge C$

Figura 3. Conjunção Eliminação ($\wedge e$)

Implicação Eliminação ($\rightarrow e$) A regra da eliminação da implicação ($\rightarrow e$), também conhecida como *Modus Ponens*, é apresentada na Figura 4a (ou Figura 4b), na qual a fórmula ψ pode ser concluída na linha p a partir da eliminação da implicação da fórmula $\varphi \rightarrow \psi$ da linha m (ou n) e φ da linha n (ou m), anteriores a p e não descartadas. A Figura 5b exibe uma aplicação da regra na qual a fórmula C é obtida na linha 4 pela eliminação da implicação das fórmulas $B \rightarrow C$ e B das linha 2 e 3, respectivamente.

$\begin{array}{c} \vdots \\ m. \quad \varphi \rightarrow \psi \\ \vdots \\ n. \quad \varphi \\ \vdots \\ p. \quad \psi \quad \rightarrow e \ m, n \end{array}$	$\begin{array}{c} \vdots \\ m. \quad \varphi \\ \vdots \\ n. \quad \varphi \rightarrow \psi \\ \vdots \\ p. \quad \psi \quad \rightarrow e \ m, n \end{array}$	<ol style="list-style-type: none"> 1. $A \wedge B$ premissa 2. $B \rightarrow C$ premissa 3. B $\wedge e$ 1 4. C $\rightarrow e$ 2,3
(a) Regra $\rightarrow e$	(b) Regra $\rightarrow e$	(c) $A \wedge B, B \rightarrow C \vdash C$

Figura 4. Implicação Eliminação ($\rightarrow e$)

Implicação Introdução ($\rightarrow i$) A regra da introdução da implicação ($\rightarrow i$) constrói condicionais que não ocorrem como premissas. Para construção de um condicional é necessário realizar *raciocínio hipotético*, isto é, supor *temporariamente* que uma dada fórmula é verdadeira. Chamamos esta fórmula de *hipótese*. Assim, utilizamos *caixas de demonstração*, que servem para delimitar o *escopo da hipótese temporária*. Observe na Figura 5a que para provar o condicional $\varphi \rightarrow \psi$ na linha $n + 1$, colocamos φ como *hipótese* no topo de uma caixa (linha m), aplicamos um número finito de regras de forma a obter ψ na linha n . Todo o raciocínio para obter ψ depende da veracidade de φ e, por isso, as fórmulas resultante deste raciocínio ficam delimitadas na caixa. Na linha seguinte ($n + 1$) podemos aplicar a regra $\rightarrow i$ para obter $\varphi \rightarrow \psi$, sendo que este condicional não mais depende da hipótese φ . Na justificativa da linha $n + 1$ utilizamos o nome da regra seguido das linhas inicial e final da caixa ($\rightarrow i$ m - n). A Figura 5b exibe uma aplicação da regra na qual a fórmula $A \rightarrow C$ é obtida na linha 6 a partir da caixa das linhas 3 a 5 em que A é a hipótese e C é a conclusão da caixa.

Observe que para a obtenção de ψ é possível utilizar quaisquer outras fórmulas como premissas e conclusões provisórias feitas até então. Demonstrações podem ter caixas dentro de caixas, ou pode-se abrir novas caixas depois de fechar outras. No entanto, existem regras sobre quais fórmulas podem ser utilizadas em que ponto na demonstração. Em geral, só podemos usar uma fórmula em um determinado ponto se esta fórmula ocorre

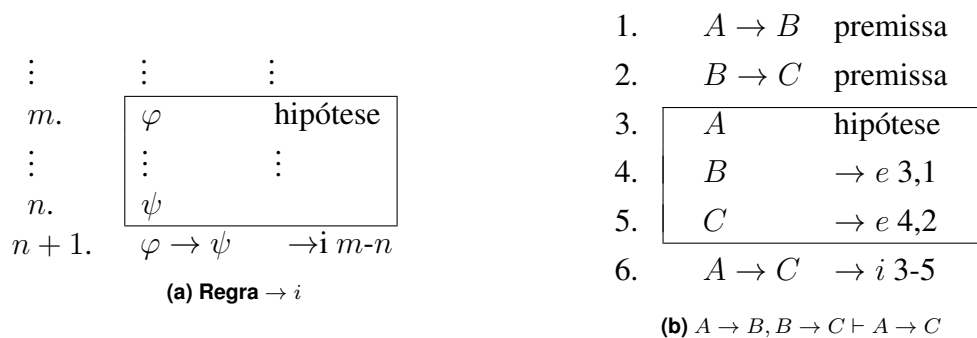


Figura 5. Implicação Introdução ($\rightarrow i$)

antes desse ponto e se nenhuma caixa que contenha a ocorrência desta fórmula tenha sido fechada [Huth and Ryan 2004].

Por limitações de espaço, as definições das demais regras não serão detalhas neste artigo, quais sejam: **Negação Introdução** ($\neg i$), Figura 6a; **Negação Eliminação** ($\neg e$), Figuras 6b e 6c; **Disjunção Introdução** ($\vee i$), Figuras 6d e 6e; **Disjunção Eliminação** ($\vee e$), Figura 6f; **Redução Ao Absurdo** (raa), Figura 6g; **Contradição Eliminação** ($\perp e$), Figura 6h; e **Copie**, Figura 6i. Para maiores detalhes, veja [Huth and Ryan 2004].

3. O Assistente de Provas NADIA

O assistente de provas NADIA, *Natural Deduction Proof Assistant*, é uma ferramenta que permite verificar automaticamente a correção de uma demonstração no sistema de Dedução Natural no estilo Fitch. NADIA foi desenvolvido na linguagem Python, pode ser utilizado em *desktop*, ou em uma plataforma *Web*¹. O assistente de provas NADIA pode ser integrado ao Moodle, permitindo assim uma melhor integração e prática com o restante do conteúdo da disciplina de Lógica.

A ideia é permitir que a demonstração seja escrita de maneira mais próxima possível da prova em papel, permitindo que estudantes possam escrever sua demonstração em texto simples e verificar a correção de sua demonstração. Caso a prova esteja errada, NADIA apresentará quais são as regras que foram utilizadas de forma errada.

A Figura 7 apresenta a correspondência entre os símbolos da lógica e os que são utilizados no NADIA. A partir de uma demonstração correta, é possível gerar código *Latex* nos estilos de demonstração: Fitch (caixas) utilizando o pacote <https://ctan.org/pkg/logicprooflogicproof> e Gentzen (árvores) utilizando o pacote `proof`.

A Figura 8 apresenta a visão geral da versão *Web* do assistente NADIA. A ferramenta tem uma área para edição da demonstração em texto simples, uma seção para o resultado da análise da demonstração e os seguintes *links*: *Check*, para verificar a correção da demonstração; *Manual*, para visualizar um documento com as regras e exemplos de demonstração; *Fitch*, para gerar o código *Latex* no estilo Fitch; e, *Gentzen*, para gerar o código *Latex* no estilo de Gentzen. Os *links Fitch* e *Gentzen* são exibidos apenas após a verificação da correção de uma demonstração.

A Figura 9a apresenta a demonstração de $A \rightarrow C, B \rightarrow C, A \vee B \vdash C$, no estilo de Fitch. A partir da verificação de que essa demonstração está correta, é possível gerar

¹Versão *Web* disponível em: <https://sistemas.quixada.ufc.br/nadia/>

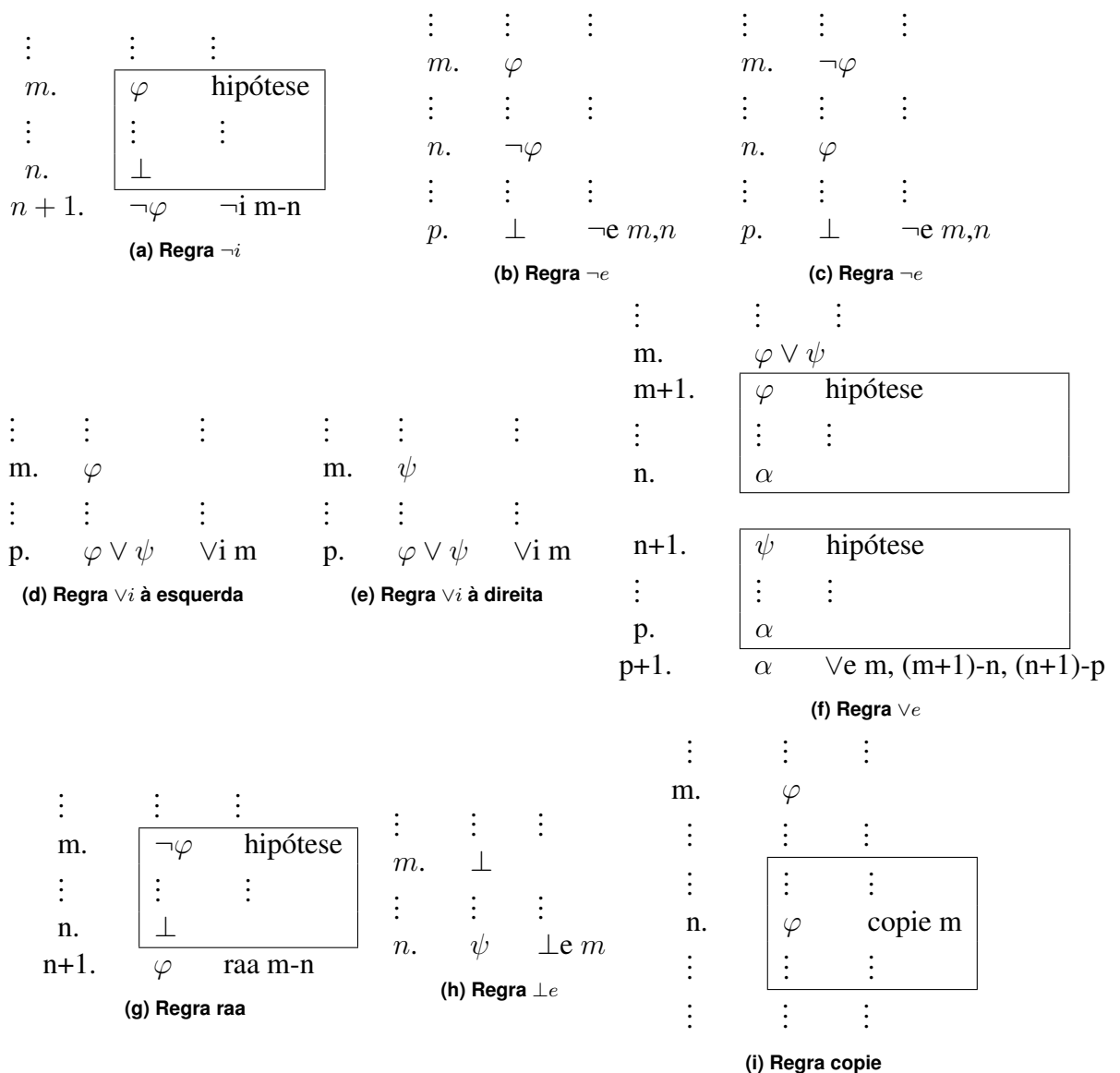


Figura 6. Demais Regras do Sistema de Dedução Natural

o código em Latex desta prova no estilo Fitch (Figura 9b) e no estilo de Gentzen (Figura 9c). A Figura 9d ilustra uma prova incorreta deste teorema. Note que na linha 7 o usuário usou a fórmula A da linha 4. Ocorre que essa fórmula estava no escopo de uma caixa (linhas 4 – 5) que já havia sido fechada. Portanto, a fórmula A já havia sido descartada e não poderia ser utilizada na linha 7.

4. Trabalhos Correlatos

Há algumas ferramentas para auxiliar o ensino e aprendizagem do conteúdo de Dedução Natural em Lógica Proposicional e Lógica de Predicados. A seguir, resumimos as funcionalidades de alguns assistentes de provas, bem como destacamos as semelhanças e diferenças entre estas ferramentas e a proposta neste trabalho.

- O JAPE² [Bornat and Sufrin 1996] é um assistente de provas *desktop* desenvol-

²O código-fonte está disponível em <https://github.com/RBornat/jape/>

Símbolo	\neg	\wedge	\vee	\rightarrow	\perp	premissa	hipótese	caixa	\vdash
Latex	<code>\lnot</code>	<code>\land</code>	<code>\lor</code>	<code>\rightarrow</code>	<code>\bot</code>	premissa	hipótese	caixa	<code>\vdash</code>
NADIA	<code>&</code>	<code>~</code>	<code> </code>	<code>-></code>	<code>@</code>	pre	hip	<code>{}</code>	<code> -</code>

Figura 7. Equivalência entre os símbolos da lógica, NADIA e Latex

NADIA - Natural Deduction proof Assistant

Menu para verificar a correção da demonstração, para o manual e para as demonstrações em Fitch e Gentzen.

Check »	Manual	Fitch	Gentzen
---------	--------	-------	---------

```

1. A                pre
2. A->C            pre
3. C                ->e 1,2

```

Esta área deve ser usada para escrever a demonstração a ser analisada.

A demonstração está correta.
A, A->C |- C

Esta área apresenta o resultado da análise da demonstração. Se estiver correta, é exibido o teorema demonstrado. Caso contrário, os erros encontrados na demonstração são exibidos.

Figura 8. Visão Geral da versão Web de NADIA

vido em Java para a construção de provas no estilo *Fitch*. As provas são realizadas através da inserção das regras de inferência por meio de sua *interface gráfica*.

- O ProofWeb³ [Maxim et al. 2010] é um assistente de provas *web* que pretende ser uma evolução do JAPE. Utiliza o Coq⁴, que é um assistente estado-da-arte para escrever provas matemáticas. O usuário deve seguir uma *sintaxe própria* ou usar a *interface gráfica* para adicionar as regras de inferência. O ProofWeb permite exibir as provas nos estilos *Fitch* e *Gentzen*.
- O PANDA⁵ [Gasquet et al. 2011] também é um assistente de provas *Web* desenvolvido em Java que se diferencia dos anteriores por permitir a elaboração de provas no estilo de *Gentzen* a partir de sua *interface gráfica*.

Para fins de ensino e aprendizagem de Dedução Natural em Lógica é muito importante que a ferramenta possibilite que o aluno escreva provas o mais parecido possível com o que é disponibilizado na literatura e com o que usualmente faria no papel. No lado esquerdo da Figura 10a podemos visualizar a escrita da prova de $A \wedge B \vdash A \wedge B \rightarrow A$ na ferramenta ProofWeb. Observe que há uma sintaxe própria da ferramenta que o aluno precisa aprender para construção da prova. Essa sintaxe difere muito do que o aluno escreveria no papel. Nesse sentido, o NADIA foi desenvolvido para que o estudante escreva provas da forma mais parecida possível ao que escreveria no papel. Por outro lado, as demonstrações no JAPE (Figura 10b) e no PANDA (Figura 10c) são realizadas por meio da aplicação de regras na interface gráfica.

5. Avaliação da Experiência do Estudante

A avaliação da experiência de uso do sistema foi realizada com alunos das turmas de Lógica para Computação dos semestres 2021.1 e 2021.2 do Campus da UFC em Quixadá.

³O ProofWeb não foi encontrado na url <http://prover.cs.ru.nl/> referenciada no artigo.

⁴Coq está disponível em: <https://coq.inria.fr/>

⁵PANDA não foi encontrado na <https://www.irit.fr/panda/> referenciada no artigo.

Check »	Manual	Fitch	Gentzen
1. A->C		pre	
2. B->C		pre	
3. A B		pre	
4. {	A	hip	
5.	C	->e 4,1 }	
6. {	B	hip	
7.	C	->e 6,2 }	
8. C		e 3, 4-5, 6-7	

A demonstração está correta.
 $A \rightarrow C, A|B, B \rightarrow C \vdash C$

(a) NADIA

1. $A \rightarrow C$ premissa
2. $B \rightarrow C$ premissa
3. $A \vee B$ premissa
4.

A	hipótese
C	$\rightarrow e$ 4,1
5.

B	hipótese
C	$\rightarrow e$ 6,2
6.

C	$\vee e$ 3, 4-5, 6-7
-----	----------------------

$$\frac{A \vee B \quad \frac{[A]^1 \quad A \rightarrow C}{C} \rightarrow e \quad \frac{[B]^2 \quad B \rightarrow C}{C} \rightarrow e}{C} \vee e, 1, 2$$

(c) Estilo Gentzen

(b) Estilo Fitch

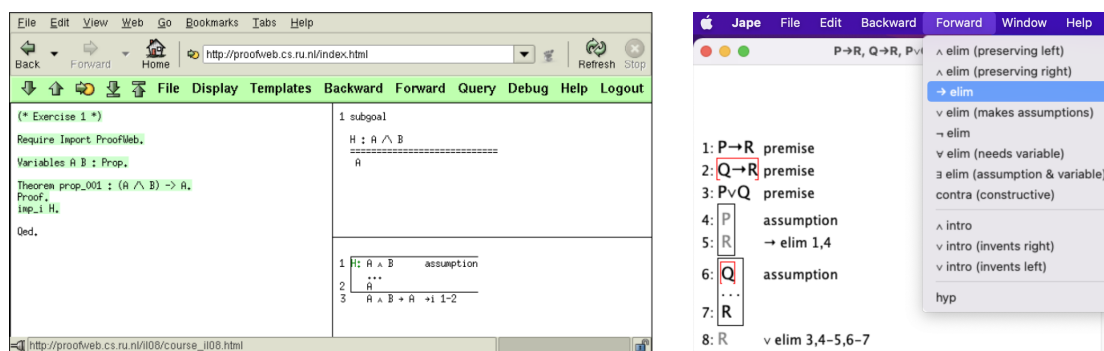
Check »	Manual
1. A->C	pre
2. B->C	pre
3. A B	pre
4. {	A hip
5.	C ->e 4,1 }
6. {	B hip
7.	C ->e 4,1 }
8. C	e 3, 4-5, 6-7

Os seguintes erros foram encontrados:

Erro de sintaxe na linha 7:
 7. C ->e 4,1 }
 ^, a referência a fórmula da linha 4 não pode ser utilizada, pois esta fórmula já foi descartada.

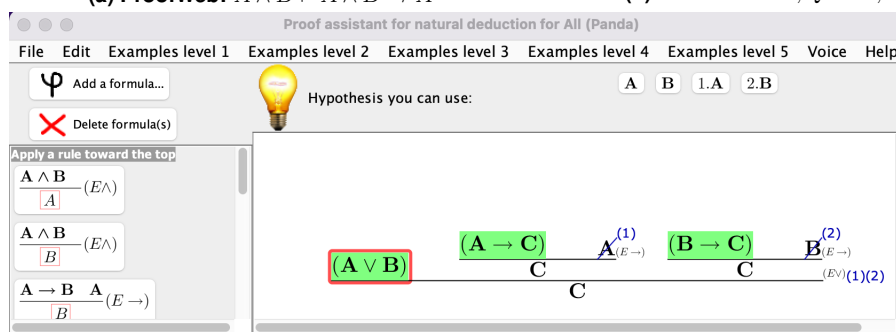
(d) Demonstração incorreta com mensagem de erro apresentada pelo NADIA.

Figura 9. NADIA - Demonstração de $A \rightarrow B, B \rightarrow C, A \vee B \vdash C$



(a) ProofWeb: $A \wedge B \vdash A \wedge B \rightarrow A$

(b) JAPE: $P \rightarrow R, Q \rightarrow R, P \vee Q \vdash R$



(c) PANDA: $A \rightarrow C, B \rightarrow C, A \vee B \vdash C$

Figura 10. Assistentes de Provas

Em 2021.1, havia 3 turmas da disciplina, contabilizando 122 alunos matriculados. Em 2021.2, havia 2 turmas, contabilizando 101 alunos matriculados.

O formulário de avaliação foi disponibilizado para que os estudantes respondessem a perguntas divididas em dois blocos: (i) perguntas sobre a utilização do sistema como ferramenta de apoio ao estudo de Dedução Natural em Lógica Proposicional e (ii) perguntas sobre a usabilidade do sistema. No semestre 2021.1, obtivemos 90 respostas ao formulário de avaliação (disponível aqui) e no 2021.2, 52 respostas (disponível aqui).

As perguntas do bloco “utilização do sistema como ferramenta de apoio ao estudo do conteúdo de Lógica Proposicional” foram: “Você utilizou o NADIA como ferramenta de estudo de Dedução Natural em Lógica Proposicional?”; “Caso tenha utilizado o NADIA, em quais momentos a ferramenta serviu como suporte às suas atividades?”; “Com que frequência você utilizou o NADIA no período em que o conteúdo de Dedução Natural em Lógica Proposicional estava sendo ministrado?”; “Você considera que o NADIA ajudou a exercitar o conteúdo de Dedução Natural em Lógica Proposicional?”.

No semestre 2021.1, 98.9% dos respondentes utilizaram o NADIA como ferramenta de estudo de Dedução Natural em Lógica Proposicional. A Figura 11(a) mostra que a maioria dos estudantes mencionaram utilizar a ferramenta duas ou mais vezes por semana durante o período que o conteúdo estava sendo ministrado. Neste semestre também 98.9% dos alunos responderam que a ferramenta ajudou a exercitar o conteúdo. Em relação à forma como os alunos utilizaram a ferramenta, 91.0% responderam que utilizaram para resolver exercícios das *atividades assíncronas* propostas; 75.3% responderam que utilizaram resolver exercícios das *atividades síncronas* e; 46.1% relataram que utilizaram para resolver outros exercícios além dos propostos pelos professores.

No semestre 2021.2, 98.1% dos respondentes utilizaram o NADIA como ferramenta de estudo do conteúdo de Dedução Natural em Lógica Proposicional. A Figura 11(b) mostra que, também no semestre 2021.2, a maioria dos estudantes mencionaram utilizar a ferramenta duas ou mais vezes por semana durante o período que o conteúdo estava sendo ministrado. Neste semestre ainda 96.2% dos alunos responderam que a ferramenta ajudou a exercitar o conteúdo. Em relação à forma como os alunos utilizaram a ferramenta, 96.1% responderam que utilizaram para resolver exercícios das *atividades assíncronas*; 80.4% responderam que utilizaram para resolver exercícios das *atividades síncronas* e; 47.1% relataram que utilizaram para resolver outros exercícios além dos propostos pelos professores. As perguntas sobre usabilidade foram em relação à facilidade de uso do sistema; compreensão das mensagens; organização da ferramenta e dificuldades na utilização. Além destas perguntas, incluímos um campo aberto para sugestões de melhorias.

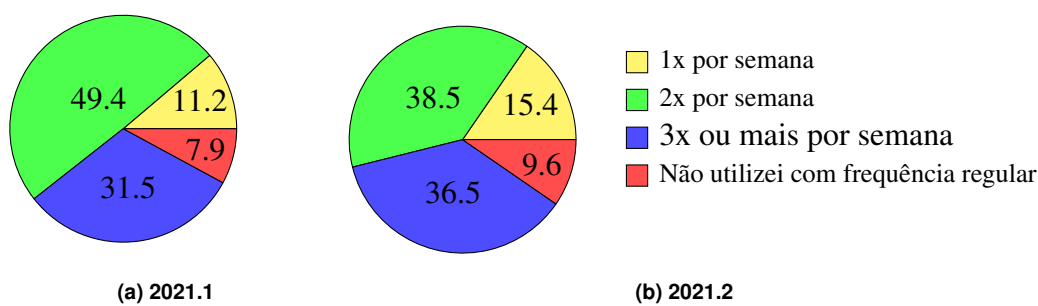


Figura 11. Frequência semanal de utilização do sistema durante o período que o conteúdo de dedução Natural estava sendo ministrado.

Em 2021.1, 34.8% avaliou como “muito bom” a facilidade de uso do sistema; 56.2% avaliou como “bom”; 7.9%, como regular e 1.1%, como ruim. Em relação à compreensão das mensagens reportadas, 17% avaliou como “muito bom”; 35.2% avaliou como bom; 45.5%, como “regular”; e 2.3%, como “ruim”. Em relação à organização da ferramenta: 40.4% avaliou como “muito bom”; 44.9% avaliou como “bom”; 13.5%, como regular e 1, 1% avaliou como ruim. Em relação à identificação e correção de erros, 11.2% dos respondentes avaliou como “muito bom”; 48.3% avaliou como “bom”; 37.1%, como regular e; 3.4% avaliou como “ruim”. Dentre as melhorias apontadas, em campo aberto, pelos alunos deste semestre podemos citar: construção de um manual interativo; mensagens de erro mais precisas e; inserção automática das numerações das linhas.

Em 2022.2, 34.6% avaliou como “muito bom” a facilidade de uso do sistema; 46.2% avaliou como “bom”; 19.2%, como regular. Em relação à compreensão das mensagens reportadas, 11.5% avaliou como “muito bom”; 36.5% avaliou como bom; 34.6%, como “regular”; e 17.3%, como “ruim”. Em relação à organização da ferramenta: 44.2% avaliou como “muito bom”; 42.3% avaliou como “bom” e 13.5%, como regular. Em relação à identificação e correção de erros, 15.4% dos respondentes avaliou como “muito bom”; 32.7% avaliou como “bom”; 34.6%, como regular e; 17.3%, como “ruim”. Dentre as melhorias apontadas, em campo aberto, pelos alunos deste semestre podemos citar: mostrar na página a correspondência entre os símbolos da lógica e os do NADIA; melhorar as mensagens de erro e; colorir as chaves que representam as caixas.

6. Conclusão e Trabalhos Futuros

Este trabalho apresentou um assistente de provas para auxiliar no ensino de Dedução Natural no estilo de Fitch (caixas). A ferramenta foi desenvolvida em Python, integrada ao Moodle e testada em cinco disciplinas de Lógica para Computação, nos semestres de 2021.1 e 2021.2, nos quais 223 alunos estavam matriculados neste período.

NADIA permite que o estudante escreva as provas de Dedução Natural em Lógica Proposicional no estilo Fitch de uma forma bem semelhante ao que se aprende na literatura e ao que usualmente se faz no papel. A ferramenta foi avaliada por 142 estudantes neste período e a grande maioria mencionou: utilizar a ferramenta duas ou mais vezes por semana durante o período que o conteúdo de Dedução Natural em Lógica Proposicional estava sendo ministrado; que a ferramenta ajudou a exercitar o conteúdo de Dedução Natural em Lógica Proposicional; utilizar a ferramenta para resolver exercícios das *atividades assíncronas* propostas; e, utilizar a ferramenta para resolver exercícios das *atividades síncronas* propostas. Em relação à usabilidade da ferramenta, o principal problema apontado pelos alunos foi em relação à compreensão das mensagens de erros reportadas.

Como trabalho futuro pretendemos aperfeiçoar a interface do NADIA, permitindo que as demonstrações sejam visualizadas na própria interface; melhorar a qualidade das repostas apresentadas aos erros e; realizar a avaliação com alunos matriculados no semestre 2022.1 para a implementação de Dedução Natural em Lógica de Predicados.

Agradecimentos

Este trabalho é parcialmente financiado pelo projeto 04772314/2020/FUNCAP.

Referências

- Bornat, R. and Sufrin, B. (1996). Jape's quiet interface.
- Chang, C.-L. and Lee, R. C.-T. (1976). *Symbolic logic and mechanical theorem proving*. Academic press.
- da Costa, N. (2008). *Introdução aos Fundamentos da Matemática*. Hucitec, 4a edition.
- Gasquet, O., Schwarzentruher, F., and Strecker, M. (2011). Panda: A proof assistant in natural deduction for all. a gentzen style proof assistant for undergraduate students. In Blackburn, P., van Ditmarsch, H., Manzano, M., and Soler-Toscano, F., editors, *Tools for Teaching Logic*, pages 85–92, Berlin, Heidelberg. Springer Berlin Heidelberg.
- Gentzen, G. (1969). *The collected papers*. North-Holland Publishing Company.
- Huth, M. and Ryan, M. D. (2004). *Logic in computer science - modelling and reasoning about systems (2. ed.)*. Cambridge University Press.
- Maxim, H., Kaliszyk, C., van Raamsdonk, F., and Wiedijk, F. (2010). Teaching logic using a state-of-art proof assistant. *Acta Didactica Napocensia*, 3.
- Pelletier, F. J. (1999). A brief history of natural deduction. *History and Philosophy of Logic*, 20(1):1–31.
- Pelletier, F. J. (2000). A history of natural deduction and elementary logic textbooks. *Logical consequence: Rival approaches*, 1:105–138.
- van Dalen, D. (1994). *Logic and structure (3. ed.)*. Universitext. Springer.