

As inteligências múltiplas e as ferramentas educacionais desenvolvidas para auxiliar os processos de ensino-aprendizagem de Lógica de Programação

Deborah Godoy Martins Corrêa, Unifesp, deborah.correa@unifesp.br,
<https://orcid.org/0000-0002-5533-0280>

Tiago de Oliveira, Unifesp, tiago.oliveira@unifesp.br,
<https://orcid.org/0000-0002-3676-5967>

Resumo: Este estudo busca identificar as ferramentas e os softwares educacionais direcionados à programação, desenvolvidos entre os anos de 1981 e 2019, visando analisá-los em relação à teoria das inteligências múltiplas. O mapeamento dessas ferramentas e softwares baseou-se na correlação entre o conteúdo didático que apresentam e as inteligências múltiplas mais aderentes a elas. Para isso, foi realizada uma pesquisa em bases de dados para identificar as ferramentas existentes e, a partir de parâmetros pré-definidos, buscou-se associar o estímulo da habilidade a cada inteligência. Como principais resultados, verificou-se que entre as ferramentas há um déficit principalmente em relação a contemplar estudantes que apresentam de maneira mais afluída as inteligências naturalista, interpessoal e intrapessoal. A análise dos dados permite, além de auxiliar os professores na escolha de ferramentas educacionais mais adequadas a cada perfil de estudantes, repensar o desenvolvimento de ferramentas ou softwares mais personalizados, que efetivamente desenvolvam competências e habilidades, e que possuam diversidade na apresentação de conteúdos o suficiente para atender aos diversos tipos de inteligências.

Palavras-chave: *Teorias da aprendizagem, ferramentas educacionais, inteligências múltiplas, Lógica de Programação, softwares educacionais.*

Multiple intelligences and educational tools developed to support the teaching-learning processes of Programming Logic

Abstract: Based on Gardner's theory this article focuses at seeking programming educational tools and softwares developed from 1981 to 2019, in order to analyze them in relation to multiple intelligences. The mapping process of these tools and softwares was based on the correlation between the didactic content they present and the most adherent multiple intelligences. To this end, a search was carried out to identify the existing tools. In the mapping process, some parameters were set to associate these tools and softwares to the stimulus of the skill related to each intelligence. As main results, it was found a deficit of tools regarding students whose characteristic is a remarkable naturalistic, interpersonal and intrapersonal intelligences. The data analysis, besides helping teachers to choose the educational tools that are most suitable for each student profile, allows them to rethink the development of more individualized tools or softwares, in order to effectively develop skills and abilities with sufficient diversity in the presentation of content, contemplating different types of intelligences.

Keywords: *Learning theories, educational tools, multiple intelligences, Programming Logic, educational software.*

1. Introdução

Na tentativa de suprir essas dificuldades, muitas ferramentas têm sido desenvolvidas em prol do ensino de computação, baseadas em diversas linguagens, em *games*, em robótica, entre outras abordagens didático-pedagógicas. No entanto, acreditamos que

um dos grandes desafios está em como utilizá-las de forma a alavancar os processos de ensino-aprendizagem, levando em consideração o perfil de cada estudante, seus conhecimentos prévios e buscando partir de contextualizações e conteúdos didáticos-pedagógicos mais familiares, indo em direção, portanto, a uma aprendizagem significativa.

Segundo Raabe e Silva (2005), aprender a programar não é uma tarefa fácil e, independentemente da linguagem escolhida, exige dos estudantes muito foco, determinação e abstração, principalmente quando nunca tiveram contato com a programação. Neste caso, além da compreensão dos novos conceitos há de se considerar que cada um assimila os conteúdos em tempos distintos, e aqueles que apresentam aptidão para determinadas capacidades, podem não apresentar para outras capacidades (GARDNER, 1994). Dentro desse contexto, Gardner considera que cada indivíduo apresenta diversas “inteligências”, relativamente independentes entre si e com diferentes medidas, e que toda inteligência deve ter o potencial de identificar problemas relevantes, assim como as habilidades necessárias para resolvê-los.

Com a finalidade de reconhecer se as ferramentas utilizadas para auxiliar os processos de ensino-aprendizagem de Lógica de Programação (LP) contemplam a diversidade das múltiplas inteligências, foi realizada inicialmente uma revisão da literatura para levantar as principais ferramentas disponibilizadas entre 1981 e 2019. Posteriormente, buscou-se mapear as ferramentas encontradas em relação às inteligências múltiplas contempladas ao se identificar as abordagens de apresentação de conteúdo didático-pedagógico disponíveis nessas ferramentas, correlacionado, com isso, cada ferramenta com os tipos de inteligências correspondentes.

Além desta introdução, este artigo apresenta também a fundamentação teórica na seção 2, a metodologia adotada na realização do estudo na seção 3, a discussão e análise dos resultados na seção 4 e as considerações finais na seção 5.

2. Fundamentação Teórica

Muitas metodologias e ferramentas têm sido propostas para facilitar o ensino-aprendizagem de LP, tendo em vista as dificuldades e necessidades dos estudantes para compreender e empregar os conceitos abstratos de programação e a aplicação desses conceitos teóricos quando do desenvolvimento de algoritmos e programas para serem executados em sistemas computacionais. Por conta dessa complexidade de aprendizado, tem-se relatado na literatura científica um alto índice de retenção e evasão de estudantes em Lógica de Programação (RAPKIEWICZ *et al.*, 2007).

As principais causas apontadas na literatura (PRIETCH; PAZETO, 2010) para explicar esses altos índices de retenção e evasão estão centradas nos estudantes, que acessam o ensino superior sem a base matemática necessária, apresentam baixa capacidade de abstração e falta de habilidade para entender e resolver problemas, assim como dificuldade em descrever os processos envolvidos (JUNIOR; RAPKIEWICZ, 2004). Aliados a essas causas, o tamanho das turmas e os diferentes níveis de conhecimento, estilos e ritmos de aprendizado que os estudantes apresentam são fatores agravantes, que levam à desmotivação ao estudo de lógica (GOMES *et al.*, 2008).

Apesar de existirem pesquisas e trabalhos sobre ensino e aprendizagem de programação desde a década de 80, o tema continua atual, pois as altas taxas de reprovação persistem. Nesse contexto é que se insere este trabalho, ao realizar o mapeamento das ferramentas desenvolvidas e contextualizá-las com as inteligências múltiplas, sistematizando estes dados de forma a auxiliar os pesquisadores a

desenvolver ferramentas mais personalizadas e auxiliar os professores na identificação e filtragem das ferramentas mais adequadas para cada tipo de estudante.

A próxima seção apresenta as dificuldades dos processos de ensino-aprendizagem da disciplina LP sob as óticas didática, cognitiva e afetiva.

2.1 Dificuldades no Ensino e Aprendizado de Programação

A evolução do ensino de programação no Brasil foi tema de pesquisa realizada por Junior e Rapkiewicz (2004) que, ao analisarem as propostas para a melhoria do aprendizado no período de 1999 a 2003, verificaram que 50% dos trabalhos apresentavam ferramentas computacionais visando facilitar os processos de ensino e aprendizagem; enquanto 25% focavam em estratégias de ensino e/ou de avaliação de competências e 25% discutiam estratégias suportadas por ferramentas computacionais.

Silva Ribeiro *et al.* (2012) publicaram sobre as principais linhas utilizadas no desenvolvimento de software baseados na programação visual para o ensino de algoritmos e programação entre 2004 e 2011. Aureliano e Tedesco (2012) apresentaram uma revisão sistemática da literatura sobre as publicações nacionais relativas à programação para iniciantes nos anais do Simpósio Brasileiro de Informática na Educação (SBIE) e dos *Workshops* de Informática na Escola (WIE) no período de 2002 a 2011. Por sua vez, Corrêa *et al.* (2018) publicaram outra revisão sistemática nacional compreendendo os anos de 2012 a 2017 sobre as estratégias didática-pedagógicas relacionadas ao ensino e aprendizado de Lógica de Programação para o ensino fundamental, ensino médio e ensino superior.

As dificuldades encontradas nos processos de ensino-aprendizagem de programação têm sido o foco de pesquisadores há décadas, e Gomes e Mendes (2007) e Faria (2013) consideram algumas possíveis causas (Tabela 1).

Tabela 1 - Problemas na aprendizagem de lógica de programação.

Natureza didática	Grande número de alunos; Dificuldade de o professor compreender as habilidades do aluno e traçar estratégias pedagógicas apropriadas; Falta de criatividade na proposição de técnicas alternativas para resolução de algoritmos; Diferença de experiência e ritmo de aprendizagem entre os alunos; Dificuldade de acompanhamento individualizado do aluno; Ambiente de realização das provas, tempo limitado, pressão e <i>stress</i> ; Pouco uso dos monitores da disciplina; Ensino conjunto de algoritmo e linguagem de programação; Ausência de bons materiais.
Natureza cognitiva	Alunos sem perfil para solução de problemas; Deficiência no raciocínio operatório formal e em matemática; Diferença de experiência e no ritmo de aprendizagem; Dificuldade em avaliar os problemas como um todo; Conhecimentos e estilos de aprendizagem diversos; Alto grau de abstração e formalização; Dificuldade em compreender e interpretar enunciados.
Natureza afetiva	Ocasionais/espórâdicos; Constantes: baixa autoestima, pouca motivação, aversão ao conteúdo ou ao professor; insegurança;

Falta de persistência e empenho;
 Problemas de ordem pessoal;
 Recursos financeiros escassos para aprendizagem
 extraclasse;
 Alunos desorientados na escolha do curso ou uma visão
 equivocada sobre ele.

Fonte: Gomes e Mendes (2007); Faria (2013). Elaborado pela autora.

A próxima seção traz algumas considerações sobre a Teoria das Inteligências Múltiplas de Gardner: seus tipos, descrições e habilidades.

2.2 Inteligências Múltiplas

Para Gardner (1995), toda inteligência deve ter o potencial de identificar problemas relevantes, assim como as habilidades necessárias para resolvê-los, e considera que cada ser humano apresenta diversas “inteligências”, relativamente independentes entre si e com diferentes medidas: verbo-linguística, lógico-matemática, rítmico-musical, visual-espacial, corporal-cinestésica, interpessoal, intrapessoal, naturalista, e que são capazes de desenvolver competências e aprender através delas.

A Teoria das Inteligências Múltiplas foi baseada em evidências neurocientíficas e Gardner (1995) associou cada inteligência a diferentes estruturas cerebrais. Shearer e Karanian (2017) analisaram mais de trezentos estudos e concluíram que há forte correlação entre as regiões cerebrais descritas por Gardner e a literatura sobre neurociências cognitivas que se acumulou nos últimos anos, e que há sólidas evidências de que cada uma das oito inteligências possui sua própria arquitetura neural.

Antunes (1988) descreve cada tipo de inteligência, e as habilidades inerentes a cada uma delas (Tabela 2).

Tabela 2 - Descrição das Inteligências e suas habilidades

Inteligência	Descrição	Habilidades
Verbo Linguística	Capacidade de processar rapidamente mensagens linguísticas, de ordenar palavras e de dar sentido lúcido às mensagens.	Descrever, narrar, observar, comparar, relatar, avaliar, concluir, sintetizar, compreender, explicar, memorizar.
Lógico Matemática	Facilidade para o cálculo e percepção da geometria espacial. Prazer específico em resolver problemas com palavras cruzadas, charadas ou lógica, como tangram, jogos de gamão e xadrez.	Enumerar, seriar, deduzir, medir, comparar, concluir, provar.
Visual Espacial	Capacidade de perceber formas e objetos mesmo em ângulos não usuais com precisão, de imaginar movimento ou deslocamento interno entre as partes de uma configuração, de recriar aspectos da experiência visual e de perceber as direções no espaço concreto e abstrato.	Localizar no espaço, localizar no tempo, comparar, observar, deduzir, relatar, combinar, transferir.
Rítmico Musical	Facilidade para identificar sons diferentes, perceber nuances em sua intensidade e direção. Reconhece sons naturais e, na música, percebe a distinção entre tom, melodia, ritmo, timbre e frequência e isola sons em agrupamentos musicais.	Observar, identificar, relatar, reproduzir, conceituar, combinar.

Corporal Cinestésica	Capacidade de trabalhar habilmente com objetos, tanto os que envolvem a motricidade dos dedos quanto os que exploram o uso integral do corpo.	Comparar, medir, relatar, transferir, demonstrar, interagir, sintetizar, interpretar, classificar.
Naturalista	Atração pelo mundo natural e sensibilidade em relação a ele, capacidade de identificação da linguagem natural a capacidade de êxtase diante da paisagem humanizada, ou não.	Relatar, demonstrar, selecionar, levantar hipóteses, classificar, revisar.
Interpessoal	Capacidade de perceber e compreender outras pessoas, descobrir as forças que as motivam e sentir grande empatia pelo outro indistinto.	Interagir, perceber, relacionar-se com empatia.
Intrapessoal	Capacidade de autoestima, automotivação, de formação de um modelo coerente e verídico de si mesmo e do uso desse modelo para operacionalizar a construção da felicidade pessoal e social.	Apresentar autoestima e autoconhecimento, ser ético.

Fonte: Antunes(1988). Elaborado pela autora.

A Sociedade Brasileira de Computação (SBC) estabelece nos Referenciais de Formação na área de Computação (2017), que os projetos desenvolvidos devem promover a integração curricular e a formação da visão do todo na formação do estudante, a partir da “resolução de problemas, habilidades de tomada de decisão e senso crítico, capacidade de planejamento, capacidade de relação interpessoal relacionadas com o trabalho em grupo, a relação de conflitos e liderança”, entre outros.

Nesse sentido, considerando a competência de cada um na resolução de problemas em face aos estímulos apresentados pelo ambiente, Gardner (1995) exalta que a Teoria das Inteligências Múltiplas faz sentido à educação ao respeitar as diversas multiplicidades: as diferenças entre as pessoas, os diferentes modos de aprender e as diversas formas como podem ser avaliadas.

A próxima seção tem como objetivo apresentar o método e os materiais utilizados no estudo.

3. Material e Método

A literatura mostra que inúmeras ferramentas têm sido desenvolvidas para auxiliar o aprendizado teórico, ou seja, focadas nos comandos e estruturas utilizados para resolução de problemas (RIGO, 2012).

Para Gomes *et al.* (2008) os aplicativos contribuem para a construção de algoritmos e recorrem a diversas técnicas, sendo as principais: a. representações visuais ou animações de algoritmos; b. linguagens de programação baseadas em ícones; c. linguagens de programação baseadas em *design languages*; d. sistemas de tutores inteligentes; e. micromundos.

Rigo (2012) também cita aplicativos que favorecem o aprendizado dos discentes quanto à construção de algoritmos, a criação de sistemas computacionais, e o mapeamento e classificação de perfis dos estudantes, enquanto Souza *et al.* (2013) cita os repositórios de cursos online. Além destes, vários sites de competições e suporte à correção de código são citados por Borowski (2016). Todas as ferramentas citadas têm contribuído para a melhoria da qualidade do ensino de programação, porém as dificuldades na aprendizagem persistem.

Os projetos que utilizam ferramentas, jogos ou programas, podem ser muito positivos nos ambientes de ensino, permitindo o aprendizado a partir da exploração

quase sem limites do assunto, pois a interdisciplinaridade é primordial dentro dos cursos da área da computação. Desta forma, os projetos tornam-se estratégias pedagógicas, e as ferramentas ao considerar as inteligências múltiplas podem reconhecer diferentes habilidades e talentos dos estudantes, e viabilizar rotas de aprendizagem personalizadas (CATALDI e LAGE, 2010).

Nesse contexto, o estudo desenvolvido utiliza uma abordagem mista (quali-quantitativa), de caráter descritivo, e a revisão de literatura utilizou dados das plataformas IEEE Explorer, Google Scholar, Capes, BDTD (Biblioteca Digital Brasileira de Teses e Dissertações), além do acesso aos acervos das bibliotecas das universidades interligadas em rede (www.usp.br/sibi) para a pesquisa das ferramentas educacionais relacionadas ao aprendizado de Lógica de Programação no período de 1981 até 2019. Em relação às análises de dados, foram utilizados métodos de estatística descritiva.

As questões que nortearam a pesquisa são: 1) “Quais ferramentas ou softwares computacionais foram desenvolvidos com o objetivo de auxiliar o ensino-aprendizagem de Lógica de Programação?” e 2) “Quais inteligências são mais estimuladas por essas ferramentas?”.

A partir das características e habilidades propostas por Antunes (1988), buscou-se verificar quais inteligências poderiam ser mais estimuladas através da utilização de cada ferramenta, a partir dos parâmetros inerentes a cada inteligência (Tabela 3).

Tabela 3 - Parâmetros que associam o estímulo da habilidade à inteligência

Inteligência	Parâmetros
Verbo - Linguística	Estimula a habilidade para a comunicação e expressão de ideias?
Lógico - Matemática	Estimula a habilidade para o raciocínio dedutivo na solução do problema?
Visual - Espacial	Estimula a habilidade de criar ou alterar imagens mentalmente?
Rítmico - Musical	Estimula a habilidade no reconhecimento de tons, sons, ritmos frequência?
Corporal - Cinestésica	Estimula a habilidade de movimentos com equilíbrio e agilidade?
Naturalista	Estimula a habilidade de valorizar o ambiente?
Interpessoal	Estimula a habilidade de se relacionar e interagir de forma positiva?
Intrapessoal	Estimula a habilidade de autocompreensão, potencialidades e limitações?

Fonte: Antunes (1988). Elaborado pela autora.

4. Resultados encontrados

A seguir são apresentadas as respostas para as questões de pesquisa.

1) “*Quais ferramentas computacionais foram desenvolvidas com o objetivo de auxiliar o ensino-aprendizagem de Lógica de Programação?*” Ao realizar a revisão bibliográfica foram encontradas 64 ferramentas ou softwares que respondem à questão de pesquisa, no período compreendido entre 1981 e 2019 (Tabela 4). As ferramentas estudadas, seus autores e referências bibliográficas, descrições e relação das inteligências que preferencialmente estimulam estão disponíveis em: https://docs.google.com/document/d/1cg8gLahypfy4F_0eeD4PQBKNCsuhZNV7JEJdWCIYrLo/edit?usp=sharing.

Tabela 4 - Ferramentas que contribuem para o aprendizado de LP

ANO	FERRAMENTAS (1981 - 2000)	ANO	FERRAMENTAS (2001 - 2019)
1981	KAREL THE ROBOT	2001	SICAS, JKAREL ROBOT, LEGO MINDSTORMS
1985	LISP-TUTOR	2002	BIP
1986	PROUST	2003	JELIOT, BLUEJ, OOP-ANIM, TEL C
1988	BALSA II, VIP	2004	TRAKLA2
1989	POLKA	2005	JHAVÉ , LUDWIG, ALGOL C
1990	GAIGS, TANGO	2007	PROGUIDE, SCRATCH
1991	TPM	2009	VISUALG
1992	ZEUS, XTANGO, ADAPT	2010	TUTORICC
1993	CEILIDH, TORTOISE	2011	HOOPALOO, INCOM, THE HUXLEY
1994	DRUIDS, FLAIR, TURTLEGRAPH, BACCII	2012	SINBAD
1995	POLKA-RC	2013	COMPUTINO, MIT APPINVENTOR, LIGHT BOT, NEWPROG
1996	LOOP TUTOR	2014	HALYEN
1997	SEE, BACCII++, MRUDS, C-TUTOR	2015	NOBUG'S SNACK BAR
1998	COACH	2016	LORD OF CODE, DEFENSE OF THE ANCIENTS 2
2000	ALICE	2017	TRI-LOGIC, PLAY CODE DOG
		2018	ADA, URI ONLINE JUDGE BLOCKS, ALGORITHM.H, TUPY ONLINE, HELPBLOCK, SOLOLEARN
		2019	THE HUXLEY, SUPER MARIO LOGIC

Elaborado pela autora.

A Figura 1 mostra a quantidade de ferramentas por ano em que foram pesquisadas.

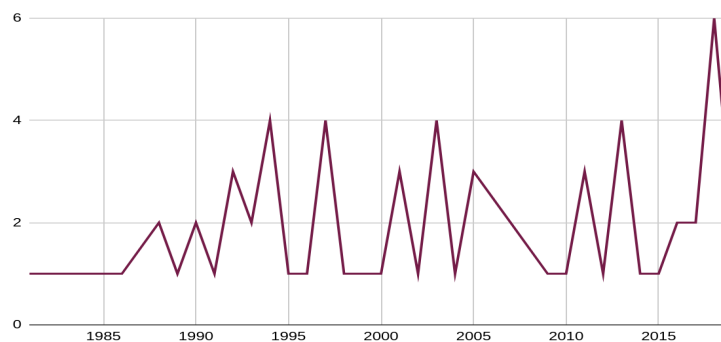


Figura 1 - Número de ferramentas pesquisadas por ano. Elaborado pela autora.

É possível observar no gráfico que sempre há o desenvolvimento de novas ferramentas, sendo constante ao longo dos anos. Esse fato reforça a preocupação dos pesquisadores, e a necessidade de endereçar os problemas relatados anteriormente para o ensino de Lógica de Programação.

2) Em relação à segunda pergunta de pesquisa: “*Quais inteligências são mais estimuladas por essas ferramentas?*” De acordo com o mapeamento realizado tendo como guia os parâmetros apresentados na Tabela 3, foi possível construir a relação descrita no link apresentado anteriormente. Analisando o mapeamento inferido, têm-se que: todas as 64 ferramentas estimulam preferencialmente a inteligência Lógico-Matemática; 63 ferramentas, a inteligência Visual-Espacial; 51 ferramentas, a Verbo-Linguística; 38 ferramentas, a Rítmico-Musical; 34 ferramentas, a Corporal-Cinestésico; 31 ferramentas a Intrapessoal; e 26 ferramentas, a Interpessoal. Tendo em vista que o foco dessas ferramentas são o aprendizado de Lógica de Programação, por definição, todas elas estimulam a inteligência Lógico-Matemática. Por sua vez, outra inteligência que aparece com bastante frequência nas ferramentas analisadas é a Visual-Espacial, como pode ser observado nas ferramentas SEE, ALICE, SCRATCH, LEGO MINDSTORMS e SUPER MARIO LOGIC. Isso ocorre tendo em vista que essas ferramentas estimulam essa inteligência por meio de representações que permitem perceber o mundo visual e espacial de forma tridimensional, e de criar e modificar imagens mentais. Em contrapartida, não foi verificada, por exemplo, ferramenta que utilize fenômenos da natureza, questões ambientais ou de sustentabilidade para apoiar o ensino de programação dos estudantes que têm a inteligência naturalista mais aflorada.

A Figura 2 mostra, por meio de um gráfico do tipo radar, que as ferramentas desenvolvidas estimulam prioritariamente as inteligências lógico-matemática e visual-espacial, seguidas da verbo-linguística e rítmico-musical. Poucas estimulam as inteligências intra e interpessoais, e nenhuma a naturalista.

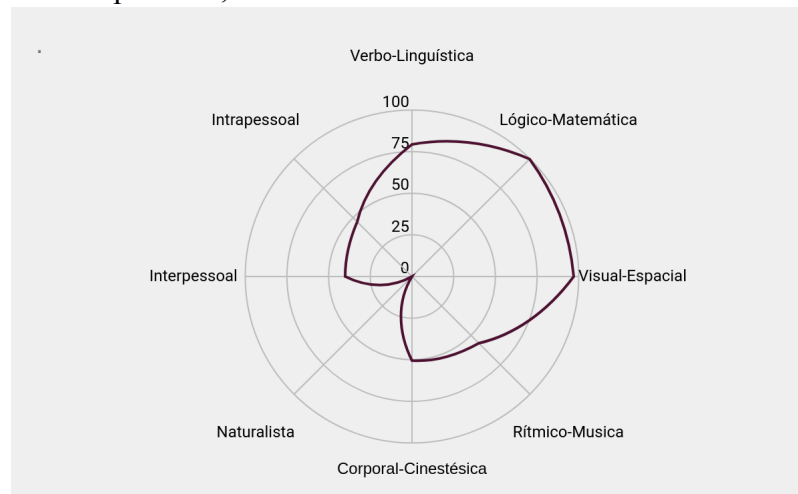


Figura 2. Inteligências mais estimuladas pelas ferramentas educacionais (%)

Aqui também vale comentar que cada indivíduo pode apresentar todas as inteligências, com intensidades distintas, mas em geral se levarmos em consideração o uso das ferramentas encontradas nesse estudo, provavelmente os que tiverem as inteligências Lógico-Matemática e Visual-Espacial mais estimuladas podem ter uma maior facilidade de aprendizado dos conceitos de Lógica de Programação, enquanto os que tiverem, por exemplo, as inteligências naturalista e interpessoal mais afloradas, poderão encontrar uma maior dificuldade de aprendizado tendo em vista que a maior

parte dessas ferramentas podem não ser as mais adequadas para o perfil desses estudantes.

É interessante comentar que essa análise baseada na correlação realizada entre as inteligências múltiplas e as ferramentas encontradas que apontaram para uma maior frequência das inteligências Lógico-Matemática e Visual-Espacial também pode ser relativamente ponderada e contextualizada com o estudo apresentado em Sisen - Sistema Computacional baseado na Teoria das Inteligências Múltiplas e nos Estilos de Aprendizagem para Auxiliar os Processos de Ensino-Aprendizagem (CÔRREA *et al.*, 2021). Nesse estudo, ao se analisar as turmas de estudantes em Lógica de Programação de uma determinada universidade nos anos de 2018, 2019 e 2020, essas duas inteligências também se destacaram nos perfis dos estudantes observados.

5. Considerações Finais

Esta pesquisa apresentou um levantamento das ferramentas computacionais utilizadas no ensino-aprendizagem de Lógica de Programação, e a correlação com as inteligências mais estimuladas.

O planejamento do uso didático das ferramentas educacionais requer a identificação de requisitos para atingir os aspectos pedagógicos e, considerando contextos adequados a cada inteligência, pode permitir um olhar mais consciente e personalizado, favorecendo as diversas formas de aprender.

Ao respeitar as múltiplas inteligências dos estudantes, e trazer conteúdos educacionais relacionados e específicos, as ferramentas podem contribuir no reconhecimento de que a diversidade de turmas e estudantes implica na escolha da metodologia de ensino-aprendizagem, e na multiplicidade das estratégias que atendam todos os estudantes envolvidos.

Dentro desse contexto, o estudo apresentado neste artigo pode contribuir para auxiliar os professores na escolha de ferramentas educacionais mais adequadas a cada perfil de estudante, indo em direção a uma aprendizagem mais significativa. Em conjunto com uma metodologia de ensino apropriada, esse estudo pode, portanto, ajudar a endereçar os problemas reportados na Tabela 1 sobre a aprendizagem de Lógica de Programação tanto de natureza didática, cognitiva como também afetiva, como por exemplo: de ordem didática - "traçar estratégias pedagógicas apropriadas" e "falta de criatividade na proposição de técnicas alternativas para resolução de algoritmos"; de ordem cognitiva - "estilos de aprendizagem diversos" e "dificuldade em compreender e interpretar enunciados"; e de ordem afetiva - "pouca motivação", "aversão ao conteúdo" e "insegurança".

Estudos futuros poderiam incluir a aplicação de diferentes métodos de análise no estudo, assim como incluir grupos de estudantes da área da computação para ratificarem seus resultados. Além disso, de acordo com os resultados apresentados, o desenvolvimento de novas ferramentas educacionais para o ensino de Lógica de Programação visando suprir ou atender determinadas inteligências também é um caminho importante a se seguir. Como exemplo, podemos citar o desenvolvimento de ferramentas educacionais de ensino de Lógica de Programação que possuam características colaborativas e focadas na realização de atividades/tarefas em grupo, o que poderia contribuir para o aprendizado de estudantes com inteligência interpessoal mais afluada, além de ajudar no cumprimento dos Referenciais de Formação na área de Computação da SBC comentados na subseção 2.2. Para auxiliar o aprendizado de estudantes com inteligência naturalista mais afluada, o desenvolvimento de ferramentas educacionais que utilizem os preceitos de metaverso (tipo de mundo virtual que tenta

replicar a realidade através de dispositivos digitais) visando a resolução de problemas de Lógica de Programação por meio da emulação de cenários e ambientes do mundo real poderá ser uma opção.

Referências Bibliográficas

- ANTUNES, C. As Inteligências Múltiplas e seus Estímulos. Campinas (SP): Papirus, 1998.
- AURELIANO, V. C. O.; TEDESCO, P. C. D. A. R. Ensino-aprendizagem de programação para iniciantes: uma revisão sistemática da literatura focada no SBIE e WIE. In: Anais do Simpósio Brasileiro de Informática na Educação, v. 23, 2012.
- BOROWSKI, D. The 10 Most Popular Coding Challenge Websites. Disponível em: <https://www.freecodecamp.org/news/the-10-most-popular-coding-challenge-websites-of-2016-fb8a5672d22f/>
- CATALDI, Z.; LAGE, F. J. Modelado del estudiante en sistemas tutores inteligentes. TE & ET, 2010.
- CORRÊA, D. G. M. *et al.* Sisen - Sistema computacional baseado na teoria das inteligências múltiplas e nos estilos de aprendizagem para auxiliar os processos de ensino-aprendizagem. Revista Novas Tecnologias na Educação, v. 19, n. 1, p. 533-542, 2021.
- CORRÊA, D.G.M. *et al.* Strategies Focused on the Teaching of Programming Logic: A Systematic Review of Brazilian Literature. In: 2018 XIII Latin American Conference on Learning Technologies (LACLO). IEEE, 2018. p. 292-298.
- FARIA, E. M. A Contribuição da Teoria Histórico-Cultural de Vygotsky para o Ensino e Aprendizagem de Algoritmo. Goiânia: PUC/GO. 116p. 2013. Tese de Doutorado.
- GARDNER, H. Estruturas da mente – A Teoria das Inteligências Múltiplas, Porto Alegre, Artmed,, 1994.
- _____. Inteligências Múltiplas: a teoria na prática. Tradução Maria Adriana Veríssimo Veronese. Porto Alegre: Artmed, 1995.
- GOMES, A.; MENDES, A. J. Problem solving in programming. Proceedings of the 19th Annual Workshop of the Psychology of Programming Interest Group. Joensuu, Finlandia, 2007.
- GOMES, A. *et al.* Aprendizagem de programação de computadores: dificuldades e ferramentas de suporte. In: Revista Portuguesa de Pedagogia, p. 161-179, 2008.
- JÚNIOR, J. C. R. P.; RAPKIEWICZ, C. E. O Processo de Ensino e Aprendizagem de Algoritmos e Programação: Uma Visão Crítica da Literatura. In: III Workshop de Educação em Computação e Informática do Estado de Minas Gerais, Belo Horizonte, MG, 2004.
- PRIETCH, S. S.; PAZETO, T. A. Estudo sobre a Evasão em um Curso de Licenciatura em Informática e Considerações para Melhorias. In: Anais do VIII Workshop de Educação e Informática, Maceió, AL, 2010.
- RAABE, A. L. A.; SILVA, J. M. C. Um Ambiente para Atendimento das Dificuldades de Aprendizagem de Algoritmos. In: Anais do XIII Workshop de Educação em Computação, São Leopoldo, RS, 2005.
- RAPKIEWICZ, C. E. *et al.* Estratégias pedagógicas no ensino de algoritmos e programação associadas ao uso de jogos educacionais. In: Revista Novas Tecnologias na Educação. Porto Alegre, RS, 2007.
- RIGO, R. SInBAD - Sistema Inteligente Bayesiano de Apoio ao Discente: orientação no estudo de programação de computadores. Goiânia: IF/UFG, 2012. 80 p. Dissertação de Mestrado.
- SHEARER, C. B.; KARANIAN, J. M. The neuroscience of intelligence: Empirical support for the theory of multiple intelligences? In: Trends in neuroscience and education, v. 6, p. 211-223, 2017.
- DA SILVA RIBEIRO, R.; BRANDÃO, L. de O.; BRANDÃO, A.. Uma visão do cenário Nacional do Ensino de Algoritmos e Programação: uma proposta baseada no Paradigma de Programação Visual. In: Simpósio Brasileiro de Informática na Educação-SBIE. 2012.
- SOUZA, M. N. A.; PAES, D. B.; TABOSA, H. R. Reflexões sobre o acesso aberto à informação científica. Revista Anisando em Ciência da Informação (RACIn), v. 1, p. 50-66, 2013.