

A IMPORTÂNCIA DA LÓGICA DE PROGRAMAÇÃO NO DESENVOLVIMENTO DE HABILIDADES DE RESOLUÇÃO DE PROBLEMAS: COMO A LÓGICA DE PROGRAMAÇÃO PODE AJUDAR A DESENVOLVER HABILIDADES CRÍTICAS EM ALUNOS

THE IMPORTANCE OF PROGRAMMING LOGIC IN DEVELOPING
PROBLEM-SOLVING SKILLS: HOW PROGRAMMING LOGIC CAN HELP
DEVELOP CRITICAL SKILLS IN STUDENTS

LA IMPORTANCIA DE LA LÓGICA DE PROGRAMACIÓN EN EL DESARROLLO
DE HABILIDADES PARA RESOLVER PROBLEMAS: CÓMO LA LÓGICA DE
PROGRAMACIÓN PUEDE AYUDAR A DESARROLLAR HABILIDADES CRÍTICAS
EN LOS ESTUDIANTES

Valter da Silva

Orientador: Prof. Dr Luciano Santos de Farias

RESUMO

Este artigo analisa a importância da lógica de programação como ferramenta essencial para o desenvolvimento de habilidades cognitivas e de resolução de problemas em estudantes. Baseando-se em referências teóricas fundamentais, como Wing (2006), Deitel e Deitel (2016), Fagundes (2020) e outros, o trabalho demonstra que o pensamento computacional vai além da técnica, servindo como método estruturado para abordar desafios complexos em diversas áreas do conhecimento. O estudo destaca três pilares principais: (1) a programação como exercício de pensamento lógico (Knuth, 1997); (2) a relação entre algoritmos e desenvolvimento cognitivo (Papert, 1980); e (3) o caráter interdisciplinar da programação (Resnick, 2014). Por meio dessas perspectivas, o artigo evidencia como a lógica de programação estimula competências cruciais, como análise crítica, abstração, resiliência e criatividade.

Palavras-chave: Lógica de programação; solução de problemas; coisas computacionais; habilidades cognitivas; educação tecnológica.

ABSTRACT

This article examines the importance of programming logic as an essential tool for developing cognitive and problem-solving skills in students. Drawing on key theoretical references, including Wing (2006), Deitel and Deitel (2016), Fagundes (2020), and others, the study demonstrates that computational thinking goes beyond technical skills, serving as a structured method for tackling complex challenges

across various fields of knowledge. The research highlights three main pillars: (1) programming as an exercise in logical thinking (Knuth, 1997); (2) the relationship between algorithms and cognitive development (Papert, 1980); and (3) the interdisciplinary nature of programming (Resnick, 2014). Through these perspectives, the article demonstrates how programming logic fosters crucial competencies such as critical analysis, abstraction, resilience, and creativity. The conclusion emphasizes that systematic instruction in programming logic not only prepares students for technological demands but also equips them with transferable skills to address academic and real-world problems, reinforcing its role as a fundamental component of contemporary education.

Keywords: Programming logic; problem-solving; computational thinking; cognitive skills; technology education.

RESUMEN

Este artículo analiza la importancia de la lógica de programación como herramienta esencial para desarrollar habilidades cognitivas y de resolución de problemas en estudiantes. Basándose en referencias teóricas clave como Wing (2006), Deitel y Deitel (2016), Fagundes (2020) y otros, el estudio demuestra que el pensamiento computacional va más allá de lo técnico, funcionando como un método estructurado para abordar desafíos complejos en diversos campos del conocimiento. La investigación destaca tres pilares principales: (1) la programación como ejercicio de pensamiento lógico (Knuth, 1997); (2) la relación entre algoritmos y desarrollo cognitivo (Papert, 1980); y (3) el carácter interdisciplinario de la programación (Resnick, 2014). A través de estas perspectivas, el artículo evidencia cómo la lógica de programación fomenta competencias cruciales como el análisis crítico, la abstracción, la resiliencia y la creatividad. Se concluye que la enseñanza sistemática de lógica de programación no solo prepara a los estudiantes para las demandas tecnológicas, sino que también los dota de habilidades transferibles para enfrentar problemas académicos y cotidianos, reforzando su papel como componente fundamental en la educación contemporánea.

Palabras clave: Lógica de programación; resolución de problemas; pensamiento computacional; habilidades cognitivas; educación tecnológica.

1 INTRODUÇÃO

Na conjuntura da sociedade atual, notadamente influenciada pela expansão das tecnologias digitais e pela importância da informação, a lógica de programação transcende a esfera técnica, tornando-se um instrumento cognitivo imprescindível para aprimorar o raciocínio estruturado e a capacidade de solucionar desafios. Como ressaltado por Wing (2006) em seu influente artigo "Pensamento Computacional", essa forma de pensar — da qual a lógica de programação faz parte — deve ser vista como uma habilidade essencial a todos, não apenas aos especialistas em computação.

Embora seja inegável o quanto ela contribui para a formação, percebe-se que a lógica de programação é um dos maiores desafios para os estudantes, principalmente em cursos técnicos e no ensino médio integrado. Muitos alunos lutam para entender ideias como abstração, divisão de problemas, organização lógica e criação de algoritmos, o que leva a um baixo desempenho e falta de interesse. Assim, surge a seguinte questão de pesquisa: como o ensino da lógica de programação pode ajudar a aprimorar as habilidades de resolução de problemas, ao mesmo tempo em que reduz as dificuldades?

Qual a natureza dos desafios que os alunos encontram ao longo do seu aprendizado?

Este estudo visa examinar como instruir lógica de programação de forma organizada pode impulsionar as capacidades mentais dos alunos na hora de solucionar problemas. A análise leva em conta as barreiras comuns que aparecem durante o ensino e a absorção desse conhecimento. Os propósitos mais específicos são: (i) explorar a lógica de programação como uma atividade que estimula o raciocínio lógico e metódico; (ii) entender a ligação entre a aplicação de algoritmos e a evolução da cognição dos estudantes; e (iii) ponderar sobre o papel da programação como recurso interdisciplinar de ensino, auxiliando na resolução dos obstáculos no aprendizado.

A motivação por trás desta pesquisa reside na convicção de que programar vai além do simples conhecimento de idiomas de computador; é, acima de tudo, um método para solucionar questões. De acordo com Deitel e Deitel (2016), "programar é principalmente sobre solucionar problemas; a linguagem é apenas um meio de expressar a solução", sublinhando que o foco da educação deve estar no pensamento lógico e na criação de abordagens cognitivas. Similarmente, Fagundes (2020) salienta que a criação de algoritmos ajuda a desenvolver a habilidade de pensar de forma organizada, permitindo que o aluno divida desafios complexos em partes menores e mais fáceis de lidar. Assim sendo, entender e lidar com os obstáculos no ensino da lógica de programação é crucial para incentivar um aprendizado mais relevante e abrangente.

Esta pesquisa se fundamenta em três pilares teóricos cruciais: a visão da programação como uma atividade de raciocínio lógico, conforme apresentado por Knuth (1997); a conexão entre a prática de algoritmos e o crescimento cognitivo, como argumentado por Papert (1980), com ênfase no construcionismo e a compreensão da programação como um recurso de aprendizado criativo e interdisciplinar, como sugerido por Resnick (2014). Estas perspectivas se harmonizam com preceitos fundamentais da resolução de problemas, a exemplo daqueles propostos por Polya (1945), que postula que "um problema bem entendido já está em parte solucionado", ideia que se conecta diretamente ao processo de análise e criação de algoritmos.

Em relação à abordagem, o estudo adota uma pesquisa qualitativa, de cunho bibliográfico, sustentada pela análise criteriosa de obras e artigos científicos relevantes nos campos da lógica de programação, pensamento computacional e educação. Através desta análise, busca-se organizar as contribuições teóricas que demonstram o potencial da lógica de programação no aprimoramento de habilidades cognitivas, bem como ponderar sobre abordagens pedagógicas que possam ajudar a superar as dificuldades de aprendizado identificadas.

Assim, este artigo visa enriquecer a discussão educacional, oferecendo bases teóricas para professores e pesquisadores que desejam entender os desafios

enfrentados pelos alunos no aprendizado da lógica de programação, realçando seu potencial como ferramenta pedagógica para o desenvolvimento de habilidades cognitivas cruciais para a formação completa dos alunos.

2 REFERENCIAL TEÓRICO

2.1 Lógica de programação e pensamento computacional

A lógica de programação representa o alicerce fundamental na criação de sistemas de computadores, sendo entendida como um grupo de ideias que direcionam a maneira como organizamos nosso pensamento de forma gradual, lógica e bem estruturada. Anteriormente ao estudo de qualquer idioma de programação, a lógica de programação permite ao aluno entender como os problemas podem ser examinados, divididos em partes menores e resolvidos por meio de passos lógicos.

Wing (2006) ressalta que o raciocínio computacional abrange capacidades como simplificação, divisão, identificação de modelos e criação de sequências de ações, habilidades que vão além da área da computação e se aplicam a muitos campos do saber. Dessa forma, a lógica de programação se transforma em um instrumento mental essencial para a educação formal e a vida em sociedade dos alunos.

Knuth (1997) fortalece essa ideia ao dizer que programar não é apenas escrever códigos, mas é, principalmente, a organização cuidadosa do raciocínio em sistemas lógicos coerentes. Portanto, o ensino da lógica de programação deve dar prioridade ao desenvolvimento do pensamento bem estruturado, e não apenas ao conhecimento das regras gramaticais das linguagens.

2.2 Lógica de programação e desenvolvimento cognitivo

No âmbito da educação, Papert (1980), com base na teoria construcionista, argumenta que o aprendizado se torna mais profundo quando o aluno participa ativamente da construção do seu próprio saber. Nesse cenário, a programação

serve como uma ferramenta para o aluno expressar suas ideias, experimentar possibilidades e analisar seus próprios métodos de raciocínio.

Fagundes (2020) enriquece essa ideia, dizendo que criar algoritmos ajuda a quebrar desafios complicados em partes menores e mais fáceis de lidar, incentivando o raciocínio lógico, a capacidade de abstração e a habilidade de generalizar. Essas competências são essenciais não só para programar, mas também para resolver problemas em diversas áreas do aprendizado.

Ademais, Polya (1945), ao abordar métodos gerais para solucionar problemas, destaca que entender bem um problema é crucial para encontrar a solução. Essa ideia se conecta diretamente com o processo de criação de algoritmos, onde a análise e o planejamento vêm antes de colocar a solução em prática.

2.3 Programação como ferramenta interdisciplinar

A lógica de programação também pode ser compreendida como um recurso pedagógico interdisciplinar. Resnick (2014) defende que programar permite integrar áreas como matemática, ciências, artes e humanidades, promovendo a criatividade e a expressão pessoal dos estudantes. Dessa forma, a programação deixa de ser um conteúdo isolado e passa a contribuir para aprendizagens mais amplas e contextualizadas.

Essa abordagem interdisciplinar amplia as possibilidades pedagógicas e favorece a motivação dos alunos, especialmente daqueles que apresentam dificuldades iniciais no aprendizado da lógica de programação.

3 METODOLOGIA

Esta investigação assume um formato qualitativo, exploratório e fundamentado em pesquisa bibliográfica. O método empregado consistiu em examinar trabalhos conceituados, tanto atuais quanto de épocas passadas, que tratam da lógica de programação, do raciocínio computacional e do progresso

cognitivo, com ênfase nas publicações de Wing (2006), Deitel e Deitel (2016), Fagundes (2020), Knuth (1997), Papert (1980), Resnick (2014) e Polya (1945).

A pesquisa bibliográfica envolveu uma análise aprofundada e organizada dos textos escolhidos, dando preferência a materiais que abordam os aspectos técnicos e as estratégias pedagógicas do ensino de programação. As informações coletadas foram agrupadas em categorias por temas, o que possibilitou identificar pontos de concordância teórica, falhas existentes e contribuições importantes para entender os desafios que os alunos enfrentam ao aprender a lógica de programação.

4 APRESENTAÇÃO DOS RESULTADOS

4.1 Dificuldades no ensino e aprendizagem da lógica de programação

Ao analisar os estudos existentes, percebe-se que um grande desafio no aprendizado de programação é o foco exagerado nas regras das linguagens, deixando de lado o aprimoramento do pensamento lógico. Deitel e Deitel (2016) destacam que essa forma de ensinar pode causar desânimo, principalmente em quem está começando.

Um ponto que sempre aparece é a preparação dos professores, que muitas vezes não têm as ferramentas certas para ensinar a lógica de programação de um jeito fácil e interessante. A falta de atividades práticas e de exemplos do dia a dia também faz com que os alunos percam o interesse pelo assunto.

4.2 Potencial das metodologias ativas no ensino de programação

Com isso, os resultados demonstram que as abordagens pedagógicas que se utilizam da aprendizagem ativa possuem efeitos positivos no desenvolvimento cognitivo e socioemocional do aluno. Tanto a Aprendizagem Baseada em Problemas, Aprendizagem Baseada em Projetos, Gamificação, Programação em Pares e Sala de Aula Invertida possibilitam o desenvolvimento da autonomia discente e potencializam uma aplicabilidade prática dos conhecimentos.

Essas metodologias modelam a autonomia, a resiliência frente aos erros e a colaboratividade, ou seja, as dimensões necessárias para a educação da lógica de programação. Por fim, tornam o ambiente de ensino e aprendizagem interessantes.

4.3 Avaliação do desenvolvimento cognitivo e socioemocional

Com a compilação dos resultados das análises dos estudos citados, pôde-se notar que quatro dimensões significativas norteiam a efetividade das metodologias de ensino da lógica de programação: cognitiva, socioemocional, motivacional e interdisciplinar. Elas permitem inferir se, além de domínio técnico, o aluno obtém competências humanológicas. Os resultados deles apontam que, bem executadas, as práticas pedagógicas evoluem a capacidade de resolução de problemas e pensamento crítico dos estudantes, assim como a autoconfiança.

5 CONSIDERAÇÕES FINAIS

Este artigo teve como objetivo analisar a relevância da lógica de programação enquanto recurso didático para o desenvolvimento de habilidades cognitivas e de resolução de problemas em alunos, ensinando as dificuldades que frequentemente aparecem no ensino-aprendizagem desse conteúdo. Procurou-se, em especial, ver na lógica de programação um exercício do pensamento lógico estruturado, analisar sua relação com o desenvolvimento cognitivo, bem como discutir seu potencial interdisciplinar no âmbito educacional.

Os resultados, a partir do levantamento de literatura, mostram que a lógica de programação está no centro do intelecto do aluno durante a aprendizagem, contribuindo para o desenvolvimento de concepções como a abstração, divisão de problemas, pensamento crítico e ordem lógica do raciocínio. As teorias estudadas a seguir convergem e indicam que o pensamento computacional, por proposta de Wing (2006), é um pensamento crucial para a vida acadêmica e social atual e que deve ser extrapolado do plano do ensino técnico.

Constatou-se ainda que muitas das dificuldades existentes na aprendizagem da lógica de programação tem relação com uma didática excessivamente focada na sintaxe das linguagens, e não no desenvolvimento do raciocínio lógico e na contextualização dos conteúdos. Diante disso, os autores pesquisados revelam que a introdução de metodologias ativas — dentre elas a Aprendizagem Baseada em Problemas, Aprendizagem Baseada em Projetos, Gamificação e Programação em Pares — potencializa o engajamento dos discentes, a aprendizagem significativa e o desenvolvimento de habilidades socioemocionais como a persistência, autonomia e colaboração.

Além disso, os resultados obtidos neste artigo reforçam o caráter interdisciplinar da programação, conforme ressaltado por Resnick: é possível unir diferentes áreas do conhecimento e tornar o processo de ensino e aprendizagem curricular mais contextualizado e dinâmico. Além de contribuir para a formação do estudante de forma abrangente, tal característica exerce um papel interessante do ponto de vista pedagógico, pois facilita o alcance de um novo patamar de abordagem metodológica da lógica de programação.

Um aspecto importante desta pesquisa é a limitação da natureza estritamente bibliográfica: ao fundamentar toda a discussão com séculos teóricos de especialistas da área, é impossível uma análise empírica direta dos impactos destas metodologias no desempenho acadêmico.

Portanto, os resultados não podem ser generalizados. Sendo assim, aponta-se para a necessidade de avanço nessa área. Recomenda-se a realização de pesquisas dispondo de abordagens práticas e experimentais, que poderão ser realizadas não apenas no Ensino Superior, mas no Ensino Médio e na Educação Técnica – principalmente nesta última. Análises quantitativas e qualitativas do desempenho dos estudantes, do ponto de vista acadêmico, bem como do desenvolvimento cognitivo e socioemocional, poderão ampliar os horizontes do que a lógica de programação pode fazer de efetivo pela educação atual.

6 REFERÊNCIAS

DEITEL, Paul; DEITEL, Harvey. Java: como programar. 10. ed. São Paulo: Pearson Education do Brasil, 2016.

FAGUNDES, Léa da Cruz. Lógica de programação: construindo algoritmos. Porto Alegre: Artmed, 2020.

KNUTH, Donald E. The art of computer programming. 3. ed. Reading: Addison-Wesley, 1997.

PAPERT, Seymour. Mindstorms: children, computers, and powerful ideas. New York: Basic Books, 1980.

POLYA, George. How to solve it: a new aspect of mathematical method. Princeton: Princeton University Press, 1945.

RESNICK, Mitchel. Learn to code, code to learn. EdSurge, 2014.

VYGOTSKY, Lev S. Mind in society: the development of higher psychological processes. Cambridge: Harvard University Press, 1978.

WING, Jeannette M. Computational thinking. Communications of the ACM, New York, v. 49, n. 3, p. 33–35, 2006.