

Desafios na aprendizagem de lógica de programação entre estudantes de tecnologia: uma
revisão sistemática da literatura

Challenges in learning programming logic among technology students: a systematic literature review

*Desafíos en el aprendizaje de la lógica de programación entre los estudiantes de tecnología: una
revisión sistemática de la literatura*

Marcos Winícios Pereira Martins¹
Regina Paiva Melo Marin²
Laura Beatriz Alves³

Resumo: Programar é uma atividade que beneficia qualquer indivíduo ao estimular o pensamento lógico e a resolução de problemas. No entanto, o ensino de programação é geralmente voltado para futuros profissionais da computação, especialmente no desenvolvimento de software, e é predominante em cursos técnicos e de graduação em informática. A aprendizagem de programação nesses cursos tem se mostrado complexa, o que resulta em dificuldades significativas para os alunos na compreensão de lógica de programação, leva a altos índices de reprovação e desistência em disciplinas introdutórias. Este trabalho visa identificar os principais desafios enfrentados por alunos durante a aprendizagem de programação de computadores, ao focar nas disciplinas introdutórias de lógica de programação. Para isso, foi realizada uma revisão sistemática abrangente em bases de dados como Google Acadêmico, IEEE, Periódicos da Capes, ScienceDirect e Scielo. O estudo revelou que as dificuldades mais comuns são relacionadas ao raciocínio lógico e à capacidade de abstração, fundamentais para a resolução de problemas em programação. Este trabalho pode auxiliar na elaboração de estratégias e métodos de ensino que melhoram o processo de ensino e aprendizagem de programação, ao promover melhores resultados acadêmicos e reduzindo a taxa de desistência e reprovação.

Palavras-chave: Aprendizagem. Dificuldades da programação. Lógica de programação. Programação de computadores.

Abstract: Programming is an activity that benefits any individual by stimulating logical thinking and problem-solving skills. However, programming education is generally aimed at future computing professionals, especially in software development, and is predominant in technical and undergraduate computing courses. Learning programming in these courses has proven to be complex, resulting in significant difficulties for students in understanding programming logic, leading to high failure and dropout rates in introductory courses. This study aims to identify the main challenges faced by students during the learning process of computer programming, focusing on introductory programming logic courses. To achieve this, a systematic literature review was conducted using databases such as Google Scholar, IEEE, Capes Journals, ScienceDirect, and Scielo. The study revealed that the most common difficulties are related to logical reasoning and abstraction capabilities, which are fundamental for problem-solving in programming. This work can assist in the development of teaching strategies and methods that improve the teaching and learning process of programming, promoting better academic results, and reducing dropout and failure rates.

Keywords: Computer programming. Learning. Programming difficulties. Programming logic.

1 Bacharel em Sistemas de Informação, Analista de Desenvolvimento de Sistemas na Stefanini (Indaiatuba/SP), pmarcoswinicios@gmail.com.

2 Pós-doutorado em Ciência da Computação, Professora do Instituto Federal de Educação, Ciência e Tecnologia Catarinense (IFC), rpmmarin@gmail.com.

3 Doutora em Tecnologia de Sistemas de Informação e Comunicação, Professora do Instituto Federal Goiano (IFG), laurab.alves@gmail.com.

Resumen: Programar es una actividad que beneficia a cualquier individuo al estimular el pensamiento lógico y las habilidades para resolver problemas. La programación enseña a futuros profesionales de la computación, especialmente en el desarrollo de software, y predomina en cursos técnicos y de grado en informática. Aprender a programar en estos cursos ha demostrado ser complejo, lo que resulta en dificultades significativas para los estudiantes en la comprensión de la lógica de programación, conduciendo a altos índices de reprobación y abandono en los cursos introductorios. El estudio busca identificar los principales desafíos de los estudiantes en el aprendizaje de la programación de computadoras, centrándose en los cursos introductorios de lógica de programación. Para ello, se realizó una revisión sistemática utilizando bases de datos como Google Académico, IEEE, Periódicos de Capes, ScienceDirect y Scielo. El estudio reveló que las dificultades más comunes están relacionadas con el razonamiento lógico y la capacidad de abstracción, fundamentales para la resolución de problemas en la programación. Este trabajo puede ayudar en el desarrollo de estrategias y métodos de enseñanza que mejoren el proceso de enseñanza y aprendizaje de la programación, promoviendo mejores resultados académicos y reduciendo la tasa de abandono y reprobación.

Palabras-chave: Aprendizaje. Dificultades de la programación. Lógica de programación. Programación de computadoras.

INTRODUÇÃO

O setor de tecnologia é dividido em hardware e software. O hardware envolve a produção e manutenção de componentes físicos como desktops e servidores, enquanto o software lida com a criação e manutenção de programas de computador, como sistemas operacionais e aplicativos móveis (Ieger; Bridi, 2014). Com o setor de software crescendo no Brasil (Bridi, 2015), a demanda por profissionais também aumenta.

Cursos técnicos e de graduação, como Técnico em Informática, Análise e Desenvolvimento de Sistemas, Ciência da Computação, Engenharia da Computação e Sistemas de Informação, formam programadores, analistas de sistemas e engenheiros de software. Estes cursos geralmente incluem disciplinas de programação, sendo a introdução à lógica de programação um componente essencial (Mendonça Neto, 2013). No entanto, essa disciplina é um obstáculo significativo para os alunos, resultando em altos índices de reprovação e desistência devido à complexidade do conteúdo.

O objetivo deste trabalho é identificar as dificuldades enfrentadas pelos alunos nos cursos de tecnologia ao aprenderem programação. Este estudo pode contribuir para o desenvolvimento de metodologias de ensino mais eficazes, focadas nas dificuldades identificadas. A pesquisa foi realizada por meio de uma revisão bibliográfica de artigos científicos publicados entre 2008 e 2023.

2 REFERENCIAL TEÓRICO

Para discutir sobre o tema da programação, é essencial compreender o conceito de programar e a relevância do ensino dessa habilidade. Programar envolve resolver problemas seguindo um processo lógico passo a passo, para alcançar uma solução precisa (AMBRÓSIO *et al.*, 2011).

Embora seja um conhecimento essencialmente ligado à tecnologia da informação, o ensino de lógica de programação está presente em uma variedade de cursos de graduação, abrangendo áreas das ciências exatas, humanas e biológicas (Law, Lee, Yu, 2010; Bosse, Gerosa, 2020).

O ensino de lógica de programação desempenha um papel significativo nas escolas, contribuindo para o desenvolvimento das competências necessárias para resolver problemas, o que se mostra benéfico na aquisição de conhecimentos e habilidades aplicáveis em diversos contextos (Zanetti, Borges, Ricarte, 2016).

Entretanto, apesar dos resultados positivos observados, essa prática é pouco comum nas escolas brasileiras (OLIVEIRA *et al.*, 2020). Ainda assim, é inegável que o conhecimento em programação pode trazer benefícios substanciais a qualquer indivíduo, mesmo que não tenha intenção de seguir uma carreira profissional em tecnologia.

Embora o ensino de programação esteja integrado em uma variedade de cursos de diferentes áreas, este trabalho se concen-

tra principalmente nos alunos de cursos de tecnologia e engloba cursos de graduação e cursos técnicos. Uma análise das disciplinas presentes na grade curricular desses cursos revela a presença de uma disciplina dedicada ao ensino dos fundamentos da lógica de programação, comumente denominada como Lógica de Programação no contexto brasileiro. Essa disciplina é frequentemente requisito para outras disciplinas voltadas ao desenvolvimento de software (Mendonça Neto, 2013).

Tudo no computador é controlado de forma lógica, logo, para entender o funcionamento de um computador, é fundamental que o aluno desenvolva a capacidade de pensar de forma lógica. O raciocínio lógico é uma habilidade utilizada pelas pessoas para chegar em determinadas soluções eficazes de forma que seja possível explicar o caminho percorrido até chegar no resultado (Sukanto, Megasari, 2018).

Santiago *et al.*, (2017) afirma que muitos alunos não conseguem desenvolver habilidades para construir programas básicos mesmo após dois semestres participando de matérias de programação, sendo um fator que traz desmotivação para os alunos universitários.

A complexidade do aprendizado de programação no início dos cursos de tecnologia se dá pelo fato de ser um novo assunto totalmente diferente do que já teria visto até o momento, o que exige uma nova forma de pensar e a analisar os problemas (Priesnitz Filho *et al.*, 2012) (Oliveira *et al.*, 2020).

Portanto, ao desenvolver um plano ou método de ensino, torna-se crucial considerar estratégias destinadas a abordar as dificuldades frequentemente enfrentadas pelos estudantes. Este estudo tem como objetivo revisar a literatura existente sobre o tema e identificar soluções práticas que possam ser aplicadas no contexto educacional.

3 METODOLOGIA

Este estudo é uma revisão sistemática de literatura com o objetivo de analisar trabalhos que abordam as dificuldades e desafios en-

frentados por alunos de cursos de tecnologia durante a aprendizagem de lógica de programação.

Para orientar o desenvolvimento do estudo, foi formulada uma Questão de Pesquisa (QP): “Quais as dificuldades enfrentadas por alunos de tecnologia durante o processo de aprendizagem de programação?”. Durante a investigação, surgiu a necessidade de definir questões de pesquisa mais específicas, denominadas Questões de Apoio (QA) (Morais, Mendes Neto, Osório, 2020). Entre estas, destaca-se uma questão relacionada ao conhecimento matemático, visto que programação e matemática exigem habilidades similares e estão intimamente ligadas às ciências exatas.

QA 1- O que é lógica de programação?

QA 2 - É necessário ter conhecimento elevado em matemática para aprender a programar?

QA 3 - Quais fatores contribuem para que alunos enfrentam dificuldades no processo de aprendizagem de lógica de programação.

Para realizar esta pesquisa, foram selecionados trabalhos publicados entre 2008 e 2023. A busca pelos artigos foi conduzida nas bases de dados Periódicos da Capes, Scielo, Google Acadêmico, ScienceDirect e IEEE. Para obter resultados relacionados ao tema, utilizou-se o conectivo “OR” em todas as bases de dados para agrupar as seguintes palavras-chave: programação de computadores, lógica de programação, ensino de lógica de programação, ensino de programação de computadores, aprendizagem de lógica de programação, aprendizagem de programação de computadores, dificuldades na aprendizagem de programação, dificuldades no ensino de programação. Adicionalmente, com o uso do operador “AND”, foi incluído o termo “metodologias ativas”, para identificar estudos que abordassem metodologias de ensino aplicadas em cursos de programação.

Também foi conduzida uma segunda pesquisa utilizando palavras-chave em inglês. Em consonância com esse propósito, não houve restrição quanto às datas de publicação dos estudos nem às regiões geográficas onde foram

realizados. A seleção dos artigos a serem utilizados foi realizada por meio de uma análise que considerou critérios de inclusão (CI) e critérios de exclusão (CE), os quais estão detalhados a seguir:

CI: Artigos com abordagem voltada ao tema proposto, com foco nas dificuldades dos alunos, disponíveis gratuitamente na íntegra e publicados entre 2008 e 2020.

CE: Artigos que estejam focados somente em alguma ferramenta específica, duplicados, que não estejam em português ou inglês, com pesquisas realizadas com pessoas que não sejam alunos de cursos de informática, com exceção do ensino fundamental e médio ou focados somente em assuntos mais avançados de programação.

Os artigos foram baixados e organizados no software Mendeley (2024), utilizado para facilitar a gestão de estudos e referências. Na etapa seguinte, foi realizada a leitura completa dos artigos, selecionando aqueles que atendiam aos critérios de inclusão e exclusão previamente estabelecidos. Todo esse processo foi executado duas vezes: inicialmente utilizando palavras-chave em português e posteriormente em inglês. É importante ressaltar que nas bases de dados IEEE e Science Direct foram utilizadas exclusivamente palavras-chave em inglês.

4 RESULTADOS E DISCUSSÃO

Após realizar esta filtragem, foram selecionados 1 artigo do ScienceDirect, 2 artigos do Scielo, 3 artigos do IEEE, 9 artigos do Portal de Periódicos da Capes e 15 artigos do Google Acadêmico, totalizando 30 artigos. Dentre estes, 19 artigos foram 6 publicados em português e 11 artigos em inglês.

A habilidade de programar é reconhecida como uma atividade de alta complexidade, conforme destacado por autores brasileiros como Gomes (2008) e Ferreira, Gonzaga e Santos (2010), e por seis autores estrangeiros, incluindo Ullah *et al.* (2019), Law, Lee e Yu (2010), Holvikivi (2010), Grover e Basu (2017), Cheah (2020) e Sukamto e Megasari (2018).

Diante dessas análises, fica claro que as dificuldades enfrentadas pelos alunos durante o processo de aprendizagem de programação não são exclusivas do Brasil. Segundo Cheah (2020), este é um desafio global. Portanto, alcançar bons resultados requer considerável esforço, paciência e a adoção de métodos de estudo bem estruturados (GOMES *et al.*, 2008).

Ao revisar esses estudos, identificaram-se diversos problemas associados às dificuldades na programação. Esses incluem reprovação na disciplina, desistência da matéria e até mesmo abandono do curso, conforme demonstrado no Quadro 1, a seguir.

Quadro 1 - Consequência das dificuldades em programação

Problemas	Autores/trabalhos	Quantidade de ocorrências
Reprovação na disciplina	Bosse e Gerosa (2020), Cheah (2020), Coutinho, Bonates e Moreira (2018), França e Amaral (2013), Holvikivi (2010), Oliveira <i>et al.</i> , (2017), Ferreira <i>et al.</i> , (2010), Lima e Leal (2013) Morais, Mendes Neto e Osório (2020), Souza e Moreira (2015), Priesnitz Filho <i>et al.</i> , (2012) Mendonça Neto (2013), Ullah <i>et al.</i> , (2019)	13

Desistência do curso	Ambrósio <i>et al.</i> , (2011), Bosse e Gerosa (2020), Cheah (2020), Oliveira <i>et al.</i> , (2017), Ferreira <i>et al.</i> , (2010), Lima e Leal (2013), Maceda, Villícana e Barrera (2016), Morais, Mendes Neto e Osório (2020) Souza e Moreira (2015), Santiago e Kronbauer (2017)	10
Desistência da disciplina	Ambrósio <i>et al.</i> , (2011), Bosse e Gerosa (2020), Mendonça Neto (2013), Morais, Mendes Neto e Osório (2020)	4

Fonte: Elaborado pelos autores (2024).

A partir dos dados da Quadro 1, nota-se uma preocupante dificuldade na aprendizagem da programação, refletida pelo alto índice de reprovação em disciplinas introdutórias, conforme apontado por 13 autores. Mesmo que alguns não tenham explicitamente mencionado a complexidade da programação, os dados da Tabela 5 sugerem essa conclusão.

Em síntese, programar envolve a habilidade de utilizar linguagens de programação

para criar algoritmos que resolvam problemas. Para realizar essa atividade, é crucial desenvolver diversas habilidades fundamentais.

Após revisar os estudos, identificou-se um conjunto dessas habilidades, com destaque para duas frequentemente mencionadas pelos autores: raciocínio lógico, citado por 14 autores, e capacidade de abstração, mencionada por 11 autores. Essas e outras habilidades estão detalhadas no Quadro 2.

Quadro 2- Habilidades essenciais para programar

Habilidades	Autores/trabalhos	Quantidade de ocorrências
Raciocínio Lógico	Ambrósio <i>et al.</i> , (2011), Batistela e Teixeira (2018), Bosse e Gerosa (2020), Coutinho, Bonates e Moreira (2018), Ferreira <i>et al.</i> , (2010), Oliveira <i>et al.</i> , (2017), Lima e Leal (2013), Morais, Mendes Neto e Osório (2020), Pontes (2013), Santiago e Kronbauer (2017), Scherer, Siddiq e Sánchez Viveros (2020), Souza e Moreira (2015), Souza, Silveira e Parreira (2018), Sukanto e Megasari (2018)	14

Capacidade de Abstração	Ambrósio <i>et al.</i> , (2011), Bosse e Gerosa (2020), Coutinho, Bonates e Moreira (2018), Priesnitz Filho <i>et al.</i> , (2012) Ferreira <i>et al.</i> , (2010), Holvikivi (2010), Lima e Leal (2013), Morais, Mendes Neto e Osório (2020), Pontes (2013), Santiago e Kronbauer (2017), Oliveira <i>et al.</i> , (2017)	11
Compreensão e resolução de problemas	Ambrósio <i>et al.</i> , (2011), Bosse e Gerosa (2020), Gomes <i>et al.</i> , (2008), Holvikivi (2010), Maceda, Villícana e Barrera (2016), Mendonça Neto (2013), Oliveira <i>et al.</i> , (2017), Scherer, Siddiq e Sánchez Viveros (2020)	8
Matemática	Ambrósio <i>et al.</i> , (2011), Coutinho, Bonates e Moreira (2018), Gomes <i>et al.</i> , (2008), Maceda, Villícana e Barrera (2016), Morais, Mendes Neto e Osório (2020), Mendes Neto e Osório (2020), Silva Filho e Coutinho (2021), Zeki, Nagai e Seraphim (2022)	7
Pensamento criativo	Ambrósio <i>et al.</i> , (2011), Batistela e Teixeira (2018), Scherer, Siddiq e Sánchez Viveros (2020)	3
Análise	Holvikivi (2010), Oliveira <i>et al.</i> , (2017)	2
Sequenciação lógica	Oliveira <i>et al.</i> , (2017)	1
Memorização	Oliveira <i>et al.</i> , (2017)	1

Fonte: Elaborado pelos autores (2024).

Essas habilidades iniciais são essenciais para uma aprendizagem eficaz de lógica de programação e, conseqüentemente, para o êxito na disciplina introdutória de programação de computadores, ou mesmo em disciplinas subsequentes (Maceda, Villícana, Barrera, 2016).

É importante ressaltar que, embora todas essas habilidades sejam necessárias,

é pouco comum que alguém as domine completamente antes de iniciar o aprendizado da programação. Elas são adquiridas e desenvolvidas progressivamente à medida que o aluno pratica consistentemente.

É exatamente durante a prática inicial que os alunos enfrentam desafios e dificuldades que podem resultar nas conseqüências descritas anteriormente na Tabela 1. A maio-

ria dessas dificuldades surge devido à falta de algumas das habilidades listadas na Tabela 2.

A análise crítica dos estudos utilizados neste trabalho permitiu identificar essas dificulda-

des de forma a facilitar o reconhecimento dos principais desafios enfrentados por alunos de programação. Esses dados estão detalhados no Quadro 3.

Quadro 3- Tabela das dificuldades enfrentadas por alunos

Dificuldade	Autores/Trabalhos	Quantidade de ocorrências
Falta de conhecimento em conceitos técnicos de programação	Ambrósio <i>et al.</i> , (2011), Bosse e Gerosa (2020), Cheah (2020), França e Amaral (2013), Gomes <i>et al.</i> , (2008), Gonçalves <i>et al.</i> , (2017), Grover e Basu (2017), Marimuthu e Govender, 2018, Mendonça Neto (2013), Santiago e Kronbauer (2017) Souza (2009), Souza, Silveira e Parreira (2018), Pontes (2013),	13
Dificuldade ao necessitar utilizar raciocínio lógico	Ambrósio <i>et al.</i> , (2011), Bosse e Gerosa (2020), Cheah (2020), Gomes <i>et al.</i> , (2008), Gonçalves <i>et al.</i> , (2017), Grover e Basu (2017), Lima e Leal (2013), Mendonça Neto (2013), Pontes (2013), Souza e Moreira (2015), Souza, Silveira e Parreira (2018), Castro e Santos (2023)	12
Não conseguir pensar de forma abstrata	Ambrósio <i>et al.</i> , (2011), Cheah (2020), Gomes <i>et al.</i> , (2008) Gonçalves <i>et al.</i> , (2017), Morais, Mendes Neto e Osório (2020), Pontes (2013), Priesnitz Filho, Abegg e Simonetto (2012), Santiago e Kronbauer (2017), Souza, Silveira e Parreira (2018), Castro e Santos (2023)	10

Não conseguir resolver problemas	Ambrósio <i>et al.</i> , (2011), Cheah (2020), Gomes <i>et al.</i> , (2008), Marimuthu e Govender (2018), Mendonça Neto (2013), Pontes (2013), Prokopyev <i>et al.</i> , (2020), Souza (2009)	8
Dificuldades ao tentar utilizar linguagens de programação, sintaxe	Ambrósio <i>et al.</i> , (2011) Gomes <i>et al.</i> (2008), Grover e Basu (2017), Marimuthu e Govender (2018), Priesnitz Filho, Abegg e Simonetto (2012), Pontes (2013), Souza, Silveira e Parreira (2018)	7
Dificuldades para compreender e desenvolver algoritmos	Ambrósio <i>et al.</i> , (2011), Cheah (2020), Dantas <i>et al.</i> , (2020), Gomes <i>et al.</i> , (2008) Gonçalves <i>et al.</i> , (2017), Mendonça Neto (2013)	6
Falta de conhecimento matemático	Gomes <i>et al.</i> , (2008), Grover e Basu (2017), Mendonça Neto (2013), Morais, Mendes Neto e Osório (2020), Silva Filho e Coutinho (2021), Zeki, Nagai e Seraphim (2022)	6

Fonte: Elaborado pelos autores (2024).

As dificuldades dos alunos estão apresentadas no Quadro 3, organizadas pela frequência de ocorrência nos estudos utilizados neste trabalho. A compreensão dos conceitos técnicos da programação pode ser considerada um desafio complexo. Esta é a dificuldade mais comum entre os estudantes, mencionada por 13 autores. Em seguida, destacam-se duas dificuldades cruciais para esta discussão. A primeira delas é a capacidade de aplicar o raciocínio lógico, citada por 12 autores.

A segunda dificuldade importante é a habilidade de utilizar a abstração, mencionada por 10 autores. Abstração refere-se à capacidade de imaginar como as coisas funcionam na realidade de maneira concreta (Ambrósio

et al., 2011). Um exemplo de abstração ocorre quando o aluno compreende um problema do mundo real e desenvolve uma solução através de algoritmos utilizando uma linguagem de programação.

É relevante destacar essas duas dificuldades porque estão intimamente relacionadas, o que influencia diretamente a capacidade do indivíduo de resolver problemas. Gomes *et al.* (2008) e Ambrósio *et al.* (2011) indicam que a dificuldade em resolver problemas é a mais significativa entre todas. Os autores observam que os alunos com dificuldades frequentemente iniciam a codificação do algoritmo sem compreender completamente o problema ou sem ter uma solução preliminar elaborada

mentalmente. Consequentemente, em algum ponto, o aluno pode se sentir perdido e, por falta de persistência, desiste de buscar uma solução. Os autores atribuem isso à dificuldade na abstração dos problemas, incapaz de fazer uma conexão adequada com o mundo real.

Em suma, conclui-se que essas três dificuldades mais citadas nos artigos estão inter-relacionadas e são amplamente discutidas pelos autores em estudos nacionais e internacionais, incluindo Ambrósio *et al.* (2011), Cheah (2020), Gomes *et al.* (2008), Grover e Basu (2017), Pontes (2013), Souza, Silveira e Parreira (2018), e Prokopyev *et al.* (2020).

Portanto, é crucial ao desenvolver um plano ou método de ensino considerar estratégias para abordar essas dificuldades, destacadas como principais entre os alunos conforme o Quadro 3. Ao relacionar novamente as habilidades essenciais para programar com as dificuldades mais comuns, nota-se a importância do conhecimento matemático. Segundo Gomes *et al.* (2008), Silva Filho e Coutinho (2021) e Zeki, Nagai e Seraphim (2022), é comum entre os alunos com dificuldades em programação apresentarem desafios em conceitos básicos de matemática, sugerindo que essa habilidade é fundamental para a aprendizagem de lógica de programação.

Entretanto, a necessidade do conhecimento em matemática não se resume à habilidade de realizar diferentes tipos de cálculos, mas à correlação entre a resolução de problemas utilizando matemática e o desenvolvimento de algoritmos, ambos exigindo processos mentais semelhantes, como inferência (Ambrósio *et al.*, 2011).

Para ilustrar essa correlação, considere o exemplo seguinte: resolver um problema matemático envolve dividir o problema em partes menores e estruturar uma solução de forma sequencial e lógica até chegar a um resultado válido. Esses passos são essencialmente os mesmos para o desenvolvimento de algoritmos.

Em seu estudo sobre correlações entre matemática e programação, Gomes *et al.* (2006), citado por Gomes, Areias e Henriques (2008), observaram que alguns alunos tinham dificuldade em transformar a descrição de um

problema textual em fórmulas matemáticas, o que pode ser atribuído a desafios na abstração e raciocínio lógico.

Ao analisar dessa maneira, compreende-se que é por meio do estudo da matemática que os alunos inicialmente desenvolvem o raciocínio lógico na escola (Coutinho, Bonates, Moreira, 2018). Conclui-se, portanto, que o conhecimento matemático é fundamental para adquirir as habilidades necessárias para aprender a programar (Morais, Mendes Neto, Osório, 2020). Com base nessas informações, interpreta-se que a matemática é uma das disciplinas mais intimamente relacionadas à programação.

Outro ponto destacado no Quadro 3 é a dificuldade em utilizar linguagens de programação, mencionada por 7 autores. É comum que os professores utilizem pseudocódigo, como o Portugol no Brasil, para introduzir conceitos iniciais de construção de algoritmos aos alunos (Souza, 2009).

Após a introdução ao pseudocódigo, os alunos são gradualmente familiarizados com linguagens de programação. Em geral, as linguagens de programação possuem uma característica conhecida como sintaxe, que determina a forma correta de escrever o código. Muitos alunos enfrentam dificuldades em escrever códigos sem erros de sintaxe (Gomes *et al.*, 2008; Grover, Basu, 2017).

Outra dificuldade frequente está relacionada à utilização das linguagens de programação, devido à sua sintaxe complexa (Souza, Silveira, Parreira, 2018). É relevante mencionar que as linguagens de programação são predominantemente em inglês, o que pode representar um obstáculo para os alunos brasileiros iniciantes, que frequentemente possuem conhecimento limitado do idioma (Pontes, 2013).

Como mencionado anteriormente, a dificuldade com conceitos técnicos de programação foi destacada em vários estudos analisados. Uma análise mais detalhada permitiu subdividir essa dificuldade em diferentes grupos de conceitos. Destaca-se que o entendimento das estruturas de controle é o conceito mais desafiador para os alunos, citado por 10 autores. Essas estruturas são fundamentais

na programação, permitindo a manipulação dinâmica dos dados, incluindo estruturas de decisão e de repetição.

Em seguida, 7 autores apontaram que os alunos enfrentam dificuldades na manipulação de variáveis. Esse conceito envolve a prática de armazenar dados temporariamente e gerenciá-los de maneira lógica através da entrada, processamento e saída de dados. Compreender o uso de variáveis é crucial, pois elas são utilizadas por outros recursos das linguagens de programação, frequentemente resultando em melhora no desempenho dos alunos quando dominado (Souza, Silveira, Parreira, 2018).

Os operadores e expressões lógicas foram mencionados por 4 autores como fonte de dificuldades. Esses elementos são essenciais nas linguagens de programação para realizar cálculos e comparações, facilitando o processamento de dados e a tomada de decisões críticas de forma eficiente.

As funções desempenham um papel crucial na reutilização de código e na divisão de responsabilidades dentro de um sistema. A utilização de funções não apenas reduz a quantidade de linhas de um algoritmo, mas também facilita a sua manutenção. Esta dificuldade foi destacada por 4 autores.

Por fim, os vetores são estruturas de dados que permitem armazenar múltiplos valores do mesmo tipo. Compreender este conceito é essencial para manipular grandes volumes de informações. Esta dificuldade foi mencionada por 3 autores.

É fundamental realizar uma análise que relacione essas dificuldades técnicas, pois a falta de compreensão de alguns desses conceitos pode resultar em dificuldades ao tentar utilizar outros conceitos relacionados. Grover e Basu (2017) observam que os alunos enfrentam dificuldades ao aprender a utilizar variáveis para interagir com outros conceitos técnicos. Eles apontam que os iniciantes frequentemente não compreendem quando uma variável deve estar posicionada antes, dentro ou após um loop, ou têm dificuldades para entender que variáveis que controlam um loop podem ter valores diferentes em cada iteração.

Um exemplo adicional relevante é a utilização inadequada de operadores e expressões lógicas, o que pode impossibilitar o uso correto de estruturas condicionais, mesmo que o conceito dessas estruturas seja compreendido. Esses recursos são comumente utilizados para escrever blocos de código que realizam operações específicas.

Quando uma operação precisa ser repetida em diferentes partes do sistema, é recomendável encapsular essa parte do código em uma função, o que permite sua reutilização e facilita futuras manutenções. No entanto, a utilização incorreta de qualquer recurso dentro de uma função pode levar a falhas na execução da operação esperada.

Essas diversas dificuldades tornam a aprendizagem de programação uma atividade complexa, frequentemente desmotivando os alunos (Marimuthu, Govender, 2018; Oliveira *et al.*, 2017; Pontes, 2013). Consequentemente, é comum que os alunos não se esforcem o suficiente para superar esses desafios, o que torna o processo de aprendizagem de programação ainda mais desafiador (Maceda, Villicana, Barrera, 2016; Moraes, Mendes Neto, Osório, 2020). Portanto, a falta de motivação está diretamente relacionada ao baixo desempenho dos alunos, resultando em reprovações e desistências, conforme demonstrado anteriormente no Quadro 3 (Kazimoglu, 2020).

Este é um tema de grande relevância a ser explorado no contexto da complexidade do aprendizado de programação. Law, Lee e Yu (2010) definem motivação como a disposição e vontade de uma pessoa em se engajar em determinada atividade, como aprender a programar. Os autores destacam a importância de não apenas focar na estrutura curricular ou no conteúdo de aprendizagem, mas também compreender a motivação individual de cada aluno, pois isso é fundamental para facilitar o processo de aprendizagem.

5 CONCLUSÃO

As dificuldades na aprendizagem dos conceitos básicos de programação têm sido tema de estudo por mais de uma década, tanto no

Brasil quanto internacionalmente. Este é um tema complexo devido a diversos fatores que afetam o desempenho dos alunos. Um dos principais motivos para as dificuldades é a ausência de habilidades fundamentais no início do estudo de programação. Este estudo identificou várias dessas habilidades essenciais, incluindo raciocínio lógico, capacidade de abstração, resolução de problemas, matemática, pensamento criativo, análise, sequenciação lógica e memorização. Dentre elas, as mais cruciais são o raciocínio lógico, a abstração e a resolução de problemas, habilidades que geralmente se desenvolvem gradualmente durante o estudo e prática.

O objetivo principal deste trabalho foi identificar as principais dificuldades enfrentadas por alunos de cursos de tecnologia durante o processo de aprendizagem de lógica de programação. A análise dos artigos revelou diversos desafios, como falta de conhecimento técnico em programação, dificuldade com raciocínio lógico, abstração inadequada, problemas na resolução de algoritmos, compreensão e desenvolvimento de algoritmos, além de dificuldades na aplicação de linguagens de programação e deficiências em matemática.

A capacidade de usar raciocínio lógico e abstração é essencial para a criação bem-sucedida de algoritmos, fundamental na programação. A complexidade dos conceitos técnicos, como estruturas de controle, manipulação de variáveis, operadores lógicos, funções e vetores, também apresenta desafios significativos para os alunos. Além disso, a sintaxe complexa das linguagens de programação contribui para as dificuldades enfrentadas.

A falta de conhecimento matemático também é um obstáculo comum entre os alunos com baixo desempenho em lógica de programação. No entanto, é possível aprender a programar sem um domínio avançado da matemática, embora uma base sólida em matemática possa facilitar o processo de aprendizagem, pois ela proporciona os primeiros contatos com o raciocínio lógico, abstração, resolução de problemas e sequenciação lógica desde o ensino fundamental.

Embora existam esforços contínuos para

desenvolver metodologias eficazes e currículos que facilitem o aprendizado de programação, não há uma solução universalmente eficaz para todos os problemas enfrentados pelos alunos iniciantes. A motivação pessoal desempenha um papel fundamental no sucesso do aprendizado, pois a programação é uma atividade cognitivamente exigente. Alunos desmotivados têm maior probabilidade de desistir diante das primeiras dificuldades. Este estudo oferece *insights* valiosos que podem orientar o desenvolvimento de soluções para facilitar o processo de aprendizagem de programação, abordando as principais dificuldades observadas entre alunos no Brasil e em outros países.

REFERÊNCIAS

- AMBRÓSIO, A. P. L. *et al.* Programação de computadores: compreender as dificuldades de aprendizagem dos alunos. **Revista Galego-Portuguesa de Psicoloxía e Educación**, v. 19, p. 13, 2011).
- BATISTELA, F.; TEIXEIRA, A. C. Programação de computadores e processos auxiliares da aprendizagem: o caso da Escola de Hackers. **ETD - Educação Temática Digital**, v. 20, n. 3, p. 844–861, 2018.
- BOSSE, Y.; GEROSA, M. A. Reprovações e Trancamentos nas Disciplinas de Introdução à Programação da Universidade de São Paulo: Um Estudo Preliminar. In: WORKSHOP SOBRE EDUCAÇÃO EM COMPUTAÇÃO (WEI), 23., 2015, Recife. **Anais [...]**. Porto Alegre: Sociedade Brasileira de Computação, 2015. p. 426-435 - <https://sol.sbc.org.br/index.php/wei/issue/view/541>.
- BRIDI, M. A. O Setor de Tecnologia da Informação: o que há de novo no horizonte do trabalho? **Revista De Ciências Sociais - Política & Trabalho**, v. 2, n. 41, 2015.
- CASTRO, M. B. de O.; SANTOS, V. A. dos. Gamificação como recurso para aprimorar o ensino de lógica de programação em cursos de computação no ensino superior: uma revisão sistemática. **Revista Novas Tecnologias na Educação**, Porto Alegre, v. 21, n. 2, p. 307–318, 2023.

- CHEAH, C. S. Factors contributing to the difficulties in teaching and learning of computer programming: A literature review. **Contemporary Educational Technology**, v. 12, n. 2, p. 1–14, 2020.
- COUTINHO, E.; BONATES, M.; MOREIRA, L. O. Relato sobre o Uso de uma Ferramenta de Desenvolvimento de Jogos para o Ensino Introdutório de Lógica de Programação. **ANAIS DOS WORKSHOPS DO VII CONGRESSO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO (CBIE 2018)**, v. 1, n. Cbie, p. 689, 2018 - <https://sol.sbc.org.br/index.php/cbie/issue/view/784>.
- FRANÇA, R. S. DE; AMARAL, H. J. C. DO. Mineração de Dados na Identificação de Grupos de Estudantes com Dificuldades de Aprendizagem no Ensino de Programação. **Renote**, v. 11, n. 1, p. 1–10, 2013.
- FERREIRA, C.; GONZAGA, F.; SANTOS, R. Um Estudo sobre a Aprendizagem de Lógica de Programação Utilizando Programação por Demonstração. In: **Workshop sobre Educação em Computação - Congresso da Sociedade Brasileira de Computação**, n. July, p. 981–990, 2010 - <https://sol.sbc.org.br/index.php/wie/about>.
- PRIESNITZ FILHO, W. P.; ABEGG, I.; SIMONETTO, E. Uma abordagem diferenciada no ensino de algoritmos através da utilização de uma lousa digital. **Geintec**, p. 129–137, 2012.
- GOMES, A. *et al.* Aprendizagem de programação de computadores: dificuldades e ferramentas de suporte. **Revista Portuguesa de Pedagogia**, p. 161–179, 2008.
- GROVER, S.; BASU, S. Measuring student learning in introductory block-based programming: Examining misconceptions of loops, variables, and Boolean logic. In: **PROCEEDINGS OF THE CONFERENCE ON INTEGRATING TECHNOLOGY INTO COMPUTER SCIENCE EDUCATION, ITICSE**, p. 267–272, 2017 - <https://dl.acm.org/doi/proceedings/10.1145/3430665>.
- HOLVIKIVI, J. Conditions for successful learning. **Key Competencies in the Knowledge Society**, p. 155–164, 2010.
- IEGER, E. M.; BRIDI, M. A. Profissionais de TI: Perfil e o Desafio da Qualificação Permanente. **Revista da ABET**, v. 13, n. 2, p. 262–278, 2014.
- IZEKI, C. A.; NAGAI, W. A.; SERAPHIM, E. Os objetivos como elemento nuclear do currículo: concepções e práticas de docentes na disciplina de programação introdutória em uma universidade brasileira. **Research, Society and Development**, v. 11, n. 6, p. e0111627947–e0111627947, 17 abr. 2022.
- KAZIMOGLU, C. Enhancing confidence in using computational thinking skills via playing a serious game: A case study to increase motivation in learning computer programming. **IEEE Access**, v. 8, p. 221831–221851, 2020.
- LAW, K. M. Y.; LEE, V. C. S.; YU, Y. T. Learning motivation in e-learning facilitated computer programming courses. **Computers and Education**, v. 55, n. 1, p. 218–228, 2010.
- LIMA, M. R. DE; LEAL, M. C. Motivação discente no ensino-aprendizagem de programação de computadores. **Educação & Tecnologia**, v. 17, n. 1, 2013.
- MACEDA, G.; VILLICANA, P.; BARRERA, F. More Time or Better Tools? A Large-Scale Retrospective Comparison of Pedagogical Approaches to Teach Programming. **IEEE Transactions on Education**, v. 59, n. 4, p. 274–281, 2016.
- MARIMUTHU, M.; GOVENDER, P. Perceptions of Scratch Programming among Secondary School Students in KwaZulu-Natal, South Africa. **The African Journal of Information and Communication**, n. 21, p. 51–80, 2018.
- MENDELEY. **Reference Manager Software**. Manager Library Version 1.19.8. Mendeley Ltd, Elsevier (2021). Disponível em: <https://www.mendeley.com/>. Acesso em: 03 fev 2021.

- MENDONÇA NETO, V. D. S. M. A Utilização Da Ferramenta Scratch Como Auxílio Na Aprendizagem De Lógica De Programação. **II CONGRESSO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO (CBIE 2013) WORKSHOPS (WCBIE 2013)** A, n. Cbie, p. 260–269, 2013- <https://sol.sbc.org.br/index.php/cbie/issue/view/789>.
- MORAIS, C. G. B.; MENDES NETO, F. M.; OSÓRIO, A. J. M. Dificuldades e desafios do processo de aprendizagem de algoritmos e programação no ensino superior: uma revisão sistemática de literatura. **Research, Society and Development**, v. 9, n. 10, p. e9429109287, 2020.
- OLIVEIRA, M. et al. Um Curso de Programação a Distância com Metodologias Ativas e Análise de Aprendizagem por Métricas de Software. **Renote**, v. 15, n. 1, p. 1–10, 2017.
- OLIVEIRA, M. L. S. de, SOUZA, A. A. de, BARBOSA, A. F. e BARREIROS, E. F. S. Ensino de lógica de programação no ensino fundamental utilizando o jogo Robotizen: um relato de experiência. In: **XXXIV CONGRESSO DA SOCIEDADE BRASILEIRA DE COMPUTAÇÃO – CSBC 2014 Ensino**, v. 12, p. 51–60, 2020- <https://sol.sbc.org.br/index.php/csbc/issue/view/528>.
- PONTES, H. P. Desenvolvimento de jogos no processo de aprendizado em algoritmos e programação de computadores. **SBGames**, n. December, p. 220–228, 2013.
- PROKOPYEV, M. S. et al. Development of a Programming Course for Students of a Teacher Training Higher Education Institution Using the Programming Language Python. **Propósitos y Representaciones**, v. 8, n. 3, 2020.
- SANTIAGO, A. D.; KRONBAUER, A. H. Um Modelo Lúdico para o Ensino de Conceitos de Programação de Computadores. **Revista Brasileira de Informática na Educação**, v. 25, n. 03, p. 1, 2017.
- SCHERER, R.; SIDDIQ, F.; SÁNCHEZ VIVEROS, B. A meta-analysis of teaching and learning computer programming: Effective instructional approaches and conditions. **Computers in Human Behavior**, v. 109, n. 0318, 2020.
- SILVA FILHO, F. R.; COUTINHO, E. F. Uma análise qualitativa sobre as disciplinas de fundamentos de programação e estrutura de dados com grounded theory. **Renote**, v. 19, n. 1, p. 554–563, 28 jul. 2021.
- SOUZA, C. M. VisuAlg - Ferramenta de Apoio ao Ensino de Programação. **Revista Eletrônica TECCEN**, v. 2, n. 2, p. 01, 2009.
- SOUZA, M. B. e MOREIRA, J. L. G. Integrando Jogos de Lógica Matemática no Ensino de Algoritmos: Relatos de Experimentos. **RCT: Revista de Ciência e Tecnologia**, v. 1, n. 1, p. 16, 2015.
- SOUZA, N. G. DE; SILVEIRA, S. R.; PARREIRA, F. J. Proposta de uma Metodologia para Apoiar os Processos de Ensino e de Aprendizagem de Lógica de Programação na Modalidade de Educação a Distância. **Educação Cultura e Comunicação**, v. 9, n. 18, p. 207–232, 2018.
- SUKAMTO, R. A.; MEGASARI, R. Enhancing programming logic thinking using analogy mapping. **Journal of Physics: Conference Series**, v. 1013, n. 1, 2018.
- ULLAH, Z. et al. A rule-based method for cognitive competency assessment in computer programming using bloom's taxonomy. **IEEE Access**, v. 7, p. 64663–64675, 2019.
- ZANETTI, H.; BORGES, M.; RICARTE, I. Pensamento Computacional no Ensino de Programação: Uma Revisão Sistemática da Literatura Brasileira. **Anais do XXVII Simpósio Brasileiro de Informática na Educação (SBIE 2016)**, v. 1, n. Cbie, p. 21, 2016.

Recebido em 20 de junho de 2024
Aceito em 02 de julho de 2024