

Pensamento Computacional e Soft Skills no Contexto do Ensino de Programação: Um Mapeamento Sistemático

Antonio Rege^{1,2}, Luciana Salgado¹, José Viterbo¹

¹Instituto de Computação – Universidade Federal Fluminense (UFF)
Av. Gal. Milton Tavares de Souza, s/nº – Niterói – RJ – Brasil

²Grupo de Informática para Pesquisa em Computação – Instituto Federal do Acre (IFAC)
Rio Branco – AC – Brazil

{antonio.rsantos}@ifac.edu.br, {luciana,viterbo}@ic.uff.br

Abstract. *In recent years, the teaching of computational thinking has gained prominence in scientific literature, while soft skills have been recognized as one of the major challenges in the field of Computing in Education. This paper presents a systematic review of the literature with the aim of analyzing how computational thinking and soft skills have been addressed in programming education. The results indicate a concentration on technical competencies, with little integration of interpersonal skills. Furthermore, there is a lack of well-defined pedagogical approaches and tools to guide teachers in adopting planned strategies. Therefore, the need for integrated educational proposals that promote a more comprehensive student education and expand programming teaching practices is reinforced.*

Resumo. *Nos últimos anos, o ensino de pensamento computacional tem ganhado destaque nas discussões da literatura científica, enquanto as soft skills vêm sendo reconhecidas como um dos grandes desafios na área de Computação em Educação. Este artigo apresenta um mapeamento sistemático da literatura com o objetivo de analisar como o pensamento computacional e as soft skills têm sido abordados no ensino de programação. Os achados indicam uma concentração em competências técnicas, com pouca integração de habilidades interpessoais. Observa-se, ainda, a falta de abordagens e ferramentas pedagógicas bem definidas que orientem os professores na adoção de estratégias planejadas. Reforça-se, portanto, a necessidade de propostas educativas integradas, que promovam uma formação mais completa dos estudantes e ampliem as práticas no ensino de programação.*

1. Introdução

Nas últimas décadas, o ensino de programação tem se consolidado como componente estratégico na educação de crianças, jovens e adultos, impulsionado pelo avanço da inteligência artificial, pela crescente complexidade dos sistemas e pela dependência de software. Mais do que uma habilidade técnica, programar contribui para o desenvolvimento de capacidades cognitivas, como resolução de problemas, raciocínio lógico e autonomia intelectual, essenciais para a formação de cidadãos críticos em uma sociedade mediada pela tecnologia [Bié et al. 2023, Ribeiro et al. 2022, Rum and Ismail 2017].

Nesse cenário, o Pensamento Computacional (PC) surge como competência indispensável, pois envolve práticas como decomposição, reconhecimento de padrões, abstração e algoritmos [Wing 2006]. No Brasil, desde 2022, o ensino de computação tornou-se obrigatório na educação básica, incorporando o PC ao currículo [Brasil 2022]. Desenvolvido por meio da programação, o PC não apenas constrói conhecimento técnico, mas também estimula estratégias de raciocínio que apoiam a resolução de problemas complexos em contextos reais.

Paralelamente, cresce a valorização das Soft Skills (SS), que abrangem habilidades interpessoais, comportamentais e socioemocionais [Alex 2009, Desidério et al. 2023]. Complementares às competências técnicas, essas habilidades são fundamentais para o desenvolvimento profissional, especialmente frente às exigências do mercado contemporâneo [da Silva Devincenzi et al. 2023, Desidério et al. 2023]. A BNCC também passou a incluir diretrizes para SS, promovendo uma formação mais integral dos estudantes [BRASIL 2021].

Em 2025, a Sociedade Brasileira de Computação destacou que, embora os currículos de computação no Brasil reconheçam a importância das SS, persiste uma lacuna entre a teoria e a prática pedagógica. As metodologias tradicionais muitas vezes não favorecem o desenvolvimento dessas competências, resultando em estudantes com dificuldades de comunicação, colaboração, empatia e resolução de conflitos. Soma-se a isso a resistência de parte do corpo docente em adotar abordagens inovadoras, seja por sobrecarga de trabalho, apego a práticas convencionais ou descrença na eficácia de metodologias alternativas [SBC 2025].

Pesquisas nacionais também apontam que a implementação do PC no currículo brasileiro é incipiente, com dificuldades práticas de consolidação nas escolas [Carvalho and Braga 2022, Rege et al. 2023]. Além disso, embora a BNCC enfatize o desenvolvimento do PC, não apresenta diretrizes claras sobre sua integração com as SS, o que amplia os desafios para os docentes.

Diante desse cenário, observa-se que, apesar dos avanços, ainda são raros os estudos que investigam a aplicação conjunta de PC e SS no ensino de programação. Esse panorama revela a necessidade de propostas pedagógicas sistematizadas e validadas. Nesse contexto, o presente trabalho realiza um MSL com o objetivo de identificar, organizar e analisar publicações que exploram a relação entre essas competências e o ensino de programação.

O artigo está estruturado da seguinte forma: na seção 2, serão abordados os trabalhos relacionados; na seção 3, será detalhada a metodologia do estudo, descrevendo como o trabalho foi conduzido; na seção 4, serão apresentados os resultados e as discussões, destacando as principais descobertas do estudo. Por fim, na seção 5, serão apresentadas as considerações finais, perspectivas para pesquisas futuras e limitações identificadas.

2. Trabalhos Relacionados

Na revisão da literatura, foram identificados um número limitado de estudos que, embora dialoguem com aspectos abordados neste trabalho, apresentam enfoques distintos. O estudo de [Cruz et al. 2024] investiga o efeito da promoção do PC no desenvolvimento de habilidades do século XXI, enquanto [Versuti et al. 2020] explora a relação entre habilidades socioemocionais e o uso de tecnologias educacionais em contextos escolares.

Apesar das contribuições desses trabalhos, observa-se que ambos tratam os temas de forma isolada, sem aprofundar o desenvolvimento do PC e SS no ensino de programação. Com base nesse cenário, o presente estudo se diferencia ao realizar um MSL com foco específico nessas duas abordagens, buscando compreender como têm sido desenvolvidas conjuntamente em propostas educacionais voltadas à programação.

3. Metodologia

A metodologia adotada neste MSL foi estruturada com base nas diretrizes de [Petersen et al. 2008, Kitchenham and Charters 2007]. O processo foi dividido em três etapas principais: planejamento, condução e análise dos resultados.

3.1. Planejamento

Na fase de planejamento, foram definidos os objetivos do estudo e elaborado o protocolo da pesquisa, incluindo as perguntas de investigação, a estratégia de busca, a construção da string, a escolha das bases de dados e os critérios de inclusão e exclusão. O software Parsifal foi utilizado para apoiar a organização dos dados e a aplicação dos critérios definidos.

3.1.1. Questões da pesquisa

Para compreender como a literatura tem abordado o escopo deste estudo, foi elaborado um conjunto de perguntas de pesquisa, organizadas em uma questão principal e questões secundárias que direcionam a investigação. A pergunta principal é: *De que maneira a literatura aborda o PC e as SS no ensino de programação?*

Para responder à questão principal, definiram-se as seguintes perguntas secundárias:

PS1: Com que frequência são publicadas pesquisas que abordam o PC e as SS no ensino de programação?; **PS2:** Quais contextos (níveis de ensino, modalidades, ambientes) exploram o desenvolvimento de PC e SS?; **PS3:** Quais competências de PC e SS são mais abordadas nos estudos?; **PS4:** Quais desafios professores ou estudantes enfrentam no desenvolvimento de PC e SS?; **PS5:** Quais são as evidências sobre o desenvolvimento de PC e SS no ensino de programação?

3.1.2. Estratégia e String de Busca

A estratégia de busca foi desenvolvida com base na análise de artigos da literatura. Com isso, definiram-se três eixos principais para a construção da string: computational thinking, soft skills e programming. Observou-se que os termos relacionados a SS eram utilizados de forma variada, o que exigiu a combinação de diferentes expressões.

A base de dados Scopus foi empregada nos testes iniciais de refinamento da string. Termos que não apresentaram resultados relevantes foram excluídos e, após sucessivas iterações, a versão final foi validada por especialista da área de computação em educação, assegurando seu alinhamento com os objetivos do estudo. A string de busca final foi definida da seguinte forma: (“computational thinking” AND (“programming” OR “coding”))

) AND (“soft skill*” OR “social skill*” OR “non-cognitive skill*” OR “personal skill*” OR “essential skill*” OR “interpersonal skill*” OR “social-emotional competenc*” OR “21st century skill*” OR “21st century competenc*”).

3.1.3. Base de Dados

A escolha das bases considerou critérios de relevância para a área de computação em educação. Optou-se pelo recorte de 2020 a 2024 a fim de assegurar a atualidade dos estudos analisados, considerando avanços recentes no ensino de programação, como a inclusão do PC na educação básica, e o crescente destaque das SS na educação e no mercado de trabalho. As bases de dados utilizadas neste estudo foram: ACM Digital Library, ERIC, IEEE Xplore, ISI Web of Science, Scopus e SBC OpenLib (SOL).

3.1.4. Critérios de Inclusão e Exclusão

Com base nos objetivos da pesquisa, estabeleceram-se critérios de inclusão e exclusão a fim de garantir a relevância e alinhamento dos estudos selecionados ao escopo investigado.

Critérios de Inclusão: 1. Estudos publicados entre 2020 e 2024. 2. Estudos que abordam o desenvolvimento de PC e SS no ensino de programação. 3. Estudos realizados em contextos educacionais formais, incluindo educação infantil, ensino fundamental, médio, técnico ou superior.

Critérios de Exclusão: 1. Estudos com 3 páginas ou menos (short papers). 2. Estudos com textos completos indisponíveis para download. 3. Estudos secundários. 4. Estudos que não estejam escritos em inglês ou português. 5. Estudos duplicados (somente uma cópia de cada estudo foi incluída). 6. Estudos que não sejam relevantes para as questões de pesquisa. 7. Estudos de literatura cinza, resumos de conferência, relatórios técnicos, livros ou capítulos de livro.

3.2. Condução

As buscas sistemáticas foram realizadas automaticamente nas bases selecionadas, adaptando-se às particularidades de cada uma. A string foi aplicada no texto completo ou em campos como título, resumo e palavras-chave. A Tabela 1 resume o processo em cada base.

Tabela 1. Aplicação da String de Busca nas Bases de Dados

Base de Dados	Modo de Aplicação da String	Data da última Busca
ACM Digital Library	Texto Completo	12 de outubro de 2024
IEEE Digital Library	Texto Completo	13 de outubro de 2024
ERIC	Título, Resumo e Palavras-Chave	14 de outubro de 2024
ISI Web of Science	Título, Resumo e Palavras-Chave	13 de outubro de 2024
Scopus	Título, Resumo e Palavras-Chave	14 de outubro de 2024
SOL	Texto Completo	12 de outubro de 2024

3.2.1. Resultados das Buscas

Após a aplicação da string de busca, foram identificados 320 estudos. A seleção seguiu três etapas com apoio da ferramenta Parsifal. Inicialmente, 71 duplicatas foram removidas, restando 249 estudos. Em seguida, a triagem por título e resumo excluiu 199 trabalhos, reduzindo o número para 50. Após a leitura completa, 32 estudos foram descartados, totalizando 18 estudos incluídos para extração de dados. A Figura 1 apresenta as etapas detalhadas do processo de seleção dos estudos.

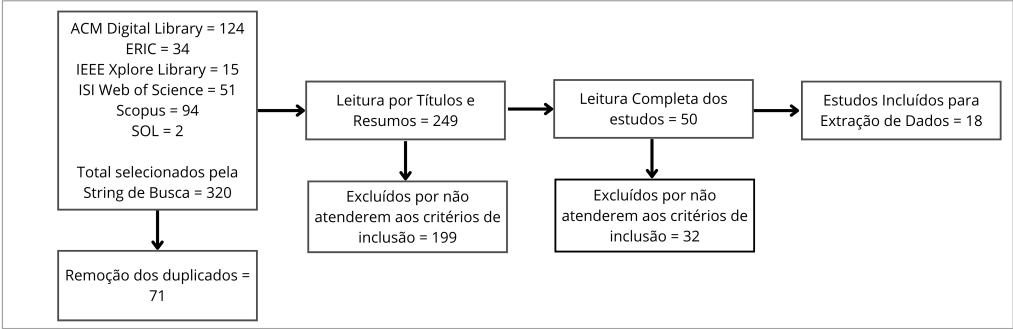


Figura 1. Processo de seleção dos estudos

3.2.2. Estudos selecionados

A Tabela 2 apresenta os autores, anos e títulos dos estudos selecionados conforme os critérios da pesquisa.

Tabela 2. Estudos Selecionados

Autor(es) e Ano	Título do Estudo
[E01] Kukul e Çakır (2020)	Exploring the Development of Primary School Students' Computational Thinking and 21st Century Skills through Scaffolding: Voices from the Stakeholders
[E02] Nouri et al. (2020)	Development of Computational Thinking, Digital Competence and 21st Century Skills When Learning Programming in K-9
[E03] Wong e Cheung (2020)	Exploring Children's Perceptions of Developing Twenty-First Century Skills through Computational Thinking and Programming
[E04] Ciancarini, Missiroli e Russo (2020)	Exploiting Agile Practices to Teach Computational Thinking
[E05] Manikutty (2021)	My Robot can Tell Stories: Introducing Robotics and Physical Computing to Children Using Dynamic Dioramas
[E06] Valls Pou, Canaleta e Fonseca (2022)	Computational Thinking and Educational Robotics Integrated into Project-Based Learning
[E07] Hsu et al. (2022)	Effects of a Pair Programming Educational Robot-Based Approach on Students' Interdisciplinary Learning of Computational Thinking and Language Learning
[E08] Fernández e Roa Martín (2022)	Educational Policy Framework to Promote Computational Thinking Towards STEAM in Public Schools in Boyacá - Colombia
[E09] Maitz et al. (2022)	Simultaneously Fostering Computational Thinking and Social-Emotional Competencies in 4th Graders Using Scratch: A Feasibility Study
[E10] Delzanno et al. (2022)	Experience-Based Training in Computer Science Education via Online Multiplayer Games on Computational Thinking
[E11] Wang (2023)	The Role of Computer-Supported Project-Based Learning in Students' Computational Thinking and Engagement in Robotics Courses
[E12] Rocha et al. (2023)	Coding Together: On Co-located and Remote Collaboration Between Children with Mixed-Visual Abilities
[E13] Bodaker e Rosenberg-Kima (2023)	Online Pair-Programming: Elementary School Children Learning Scratch Together Online
[E14] Schwartz et al. (2023)	Teaching Computational Thinking with a Tangible Development Platform: An Exploratory Field Study at School with Kniwwelino
[E15] Vasconcelos et al. (2023)	Scratch4All Project – Educate for an All-Inclusive Digital Society
[E16] Yang (2024)	Coding With Robots or Tablets? Effects of Technology-Enhanced Embodied Learning on Preschoolers' Computational Thinking and Social-Emotional Competence
[E17] Jalinus e Putra et al. (2024)	Implementation of Project-Based Learning Computational Thinking Models in Mobile Programming Courses
[E18] Kalluri et al. (2024)	Developing Future Computational Thinking in Foundational CS Education: A Case Study from a Liberal Education University in India

3.3. Análise dos Resultados

Com o objetivo de responder à questão de pergunta principal do estudo: “*De que maneira a literatura aborda o PC e as SS no ensino de programação?*”, são apresentadas a seguir as questões secundárias e suas respectivas respostas, que aprofundam a análise do tema.

3.3.1. PS1. Com que frequência são publicadas pesquisas que abordam o PC e as SS no ensino de programação?

Conforme mostra a Figura 2, o período entre 2020 e 2024 apresentou oscilações no número de publicações. Após uma queda expressiva em 2021, possivelmente associada aos impactos da pandemia de Covid-19, observou-se uma recuperação relevante nos anos de 2022 e 2023, que concentraram o maior volume de estudos. Em 2024, embora o número de publicações se mantenha elevado, os dados correspondem apenas ao período até o encerramento das buscas desta pesquisa, não refletindo o total do ano. Ainda assim, os resultados sugerem uma possível tendência de estabilização no interesse da comunidade científica pelo tema.

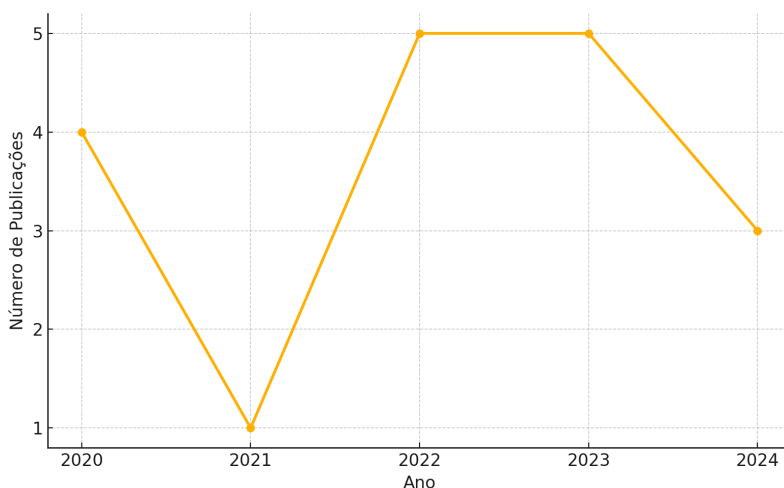


Figura 2. Frequência de publicações por ano

3.3.2. PS2. Quais contextos (níveis de ensino, modalidades, ambientes) exploram o desenvolvimento de PC e SS?

A Figura 3 apresenta a distribuição dos níveis de ensino abordados nos estudos analisados. Observa-se que o ensino fundamental é o mais representado, com 57,1% das publicações, o que evidencia um maior interesse da literatura nessa etapa da formação escolar. O ensino superior aparece em seguida, com 19,0%, enquanto a educação infantil responde por 14,3% dos estudos, indicando um movimento relevante também nas fases iniciais do aprendizado. O ensino médio, por sua vez, corresponde a apenas 9,5% das investigações, sendo o menos explorado entre os níveis identificados. Destaca-se ainda a ausência de estudos voltados especificamente ao ensino técnico de nível médio, o que revela uma possível lacuna e aponta para oportunidades de pesquisas futuras nesse segmento educacional.

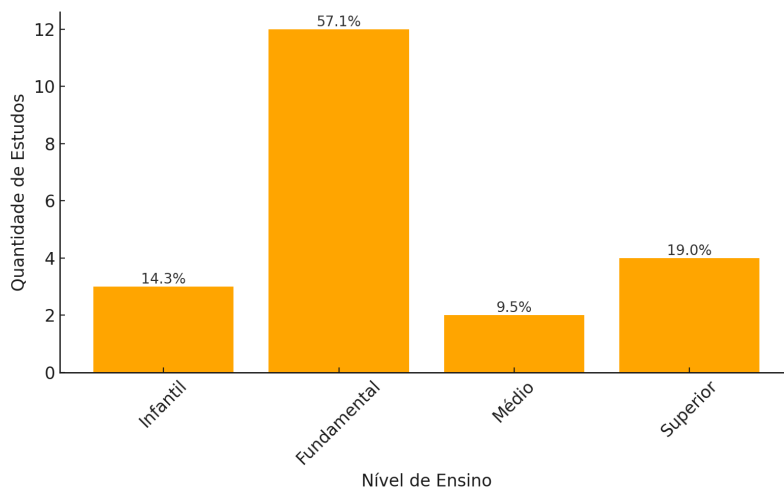


Figura 3. Distribuição de estudos por nível de ensino

3.3.3. PS3. Quais abordagens ou ferramentas são utilizadas para desenvolver o PC e SS?

A Tabela 3 mostra que, em cada nível de ensino, há uma concentração de abordagens educacionais específicas, alinhadas às características dos estudantes. No entanto, embora essas práticas sejam eficazes para o ensino de programação e o desenvolvimento de competências em PC, a promoção direcionada de SS ainda é limitada. A maioria dos estudos prioriza aspectos técnicos, sem diretrizes claras para o desenvolvimento sistematizado das SS.

Tabela 3. Abordagem e ferramentas dos estudos

Abordagem	Infantil	Fundamental	Médio	Superior
Aprendizagem Autodirigida		[E04]	[E04]	
Aprendizagem Experiencial		[E10]	[E18]	
Aprendizagem Baseada em Projetos		[E06, E08]		[E11, E17]
Aprendizagem Baseada em Problemas		[E08]		
Jogos	[E03]	[E07, E08, E10, E12]		
Programação em Pares		[E07, E13]	[E04]	[E04, E18]
Robótica Educacional	[E16]	[E02, E05, E06, E07, E12, E15]		[E11]
Metodologias Ágeis			[E16]	[E16]
Kniwwelino	[E14]	[E14]	[E14]	
KoduGameLab	[E03]	[E01]		
Scratch	[E03, E16]	[E02, E05, E06, E08, E09, E13, E15]		
Storytelling		[E05]		
Scaffolding		[E01]		
Teoria do Construcionismo	[E03]	[E05]		

Na educação infantil, destacam-se Robótica Educacional e Scratch [E03, E16], usados como estratégias lúdicas para introduzir conceitos computacionais de forma visual e interativa. No ensino fundamental, observa-se maior diversidade metodológica, com o uso de Robótica Educacional [E02, E05, E06, E07, E12, E15], Scratch [E02, E05, E06, E08, E09, E13, E15] e Jogos Educacionais [E07, E08, E10, E12], que favorecem lógica, raciocínio algorítmico e engajamento. Apesar do potencial para estimular SS, os estudos não evidenciam aplicação sistemática com esse foco.

No ensino médio, sobressaem Metodologias Ágeis e Programação em Pares [E04, E16], com ênfase em colaboração, gestão de tarefas e resolução de problemas. No ensino

superior, são recorrentes Programação em Pares, Robótica Educacional e Aprendizagem Baseada em Projetos [E04, E11, E17, E18], voltadas à prática e à autonomia. Em todos os níveis, as abordagens priorizam PC e domínio técnico, enquanto a integração estruturada de SS é pouco explorada.

3.3.4. PS4. Quais competências de PC e SS são mais abordadas nos estudos?

A Figura 4 mostra as competências mais recorrentes identificadas nos estudos. Na Figura 4a, observa-se que “Design de Algoritmos” e “Decomposição” são as competências de PC mais mencionadas. Outras competências, como “Abstração” e “Reconhecimento de Padrões”, também aparecem com destaque, ainda que em menor proporção, enquanto “Depuração”, “Paralelismo” e “Avaliação” são menos exploradas.

Na Figura 4b, no eixo das SS, destacam-se “Colaboração” e “Comunicação” como as habilidades não técnicas mais frequentes, o que reforça a importância de aspectos relacionais no ensino de programação. Outras competências, como “Criatividade”, “Trabalho em equipe”, “Resolução de problemas” e “Pensamento crítico”, aparecem de forma complementar. Habilidades como “Empatia”, “Paciência” e “Flexibilidade” são menos frequentes, mas ainda presentes em alguns estudos, compondo o panorama de competências mapeadas nesta etapa da análise.

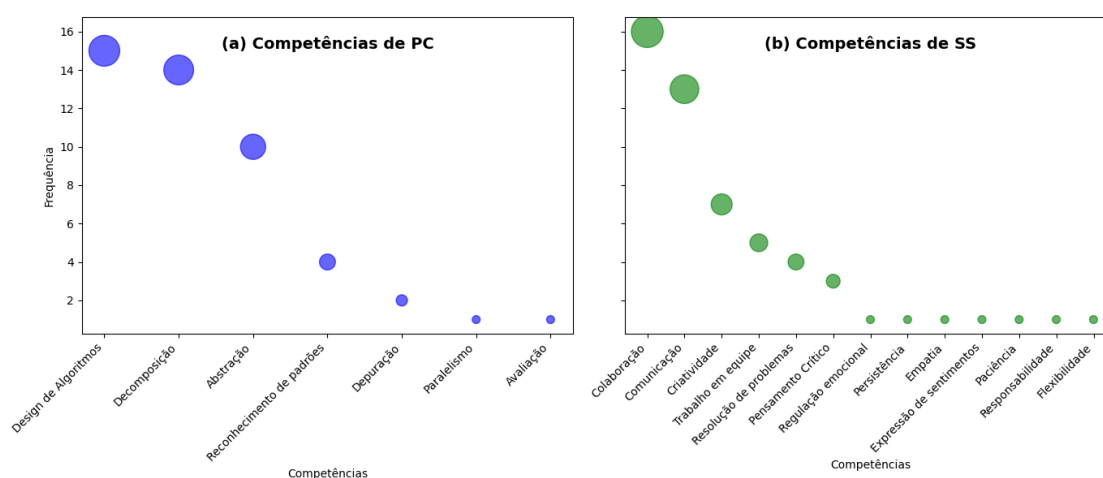


Figura 4. Frequência das Competências de PC e SS

3.3.5. PS5. Quais desafios os professores enfrentam na implementação de PC e SS no ensino de programação?

Nos estudos analisados, observou-se uma atenção mais frequente aos desafios enfrentados pelos alunos, enquanto as dificuldades dos professores ainda são pouco exploradas pela literatura. No estudo [E16], por exemplo, os estudantes apresentaram problemas na regulação emocional e na adaptação a diferentes formas de interação, o que exige dos docentes estratégias que favoreçam ambientes de aprendizagem mais acolhedores e responsivos. Já em [E11], a principal dificuldade observada é o engajamento de iniciantes

em atividades de robótica e programação, especialmente entre aqueles com baixo domínio de PC, demandando acompanhamento contínuo e mais próximo por parte do professor.

A insuficiência de infraestrutura tecnológica é outro aspecto recorrente. O estudo [E05] evidencia a escassez de equipamentos nas escolas, enquanto [E12] destaca as limitações de comunicação e coordenação em contextos remotos. Tais fragilidades impactam tanto o processo de aprendizagem quanto a efetiva aplicação de metodologias inovadoras pelos educadores.

Estudos como [E04] e [E17] enfatizam entraves diretamente ligados à prática docente. Em [E04], o uso de práticas ágeis exigiu a revisão de métodos tradicionais, a inclusão de estratégias colaborativas e o desenvolvimento de critérios para avaliar a contribuição individual dos alunos em grupo. Já [E17] aponta dificuldades na integração de projetos com foco em PC e na manutenção do engajamento dos estudantes diante de tarefas mais complexas.

A transferência de competências de PC e SS para outras disciplinas também se mostra desafiadora. O estudo [E03] revela que muitos alunos têm dificuldade em aplicar conceitos de programação em áreas como matemática, indicando a necessidade de formação interdisciplinar dos docentes. De forma semelhante, [E15] aponta a carência de recursos humanos e materiais, o que limita o desenvolvimento de atividades colaborativas e abordagens pedagógicas inovadoras.

Outros trabalhos reforçam a importância da formação docente. Em [E02], a ausência de domínio técnico por parte dos professores compromete o suporte necessário a estudantes com dificuldades em programação. Já [E08] identifica barreiras institucionais, como a falta de recursos e de formação continuada, dificultando a inserção efetiva do PC no currículo escolar.

Assim, ainda que a literatura enfatize os problemas vivenciados pelos alunos, torna-se evidente a necessidade de investigações mais aprofundadas sobre os obstáculos enfrentados pelos professores. Compreender essas barreiras pode subsidiar a elaboração de políticas públicas, programas de capacitação e materiais pedagógicos mais eficazes, promovendo um ensino de programação mais qualificado, equitativo e sustentável em diferentes contextos educacionais.

4. Resultados e Discussões

Com base nos estudos analisados, observa-se uma ênfase no desenvolvimento de competências técnicas do PC, como decomposição, abstração e algoritmos. Ferramentas como Scratch, robótica educacional e jogos são amplamente utilizadas com esse propósito. No entanto, essas estratégias, embora com potencial para desenvolver SS, raramente são aplicadas de forma planejada e sistemática com esse objetivo.

Verifica-se uma carência de orientações claras sobre como estruturar atividades que promovam simultaneamente o desenvolvimento de PC e SS. Em relação às SS, a SBC, em 2025, já apontou que essas habilidades representam um dos grandes desafios educacionais a serem superados até 2035, o que reforça a urgência de propostas pedagógicas que auxiliem os educadores tanto no planejamento didático quanto na escolha de ferramentas adequadas.

Além disso, a maioria dos estudos revisados concentra-se nas dificuldades en-

frentadas pelos alunos, como falta de engajamento, desafios de adaptação e ausência de suporte, dedicando pouca atenção às necessidades dos professores. São escassos os trabalhos que oferecem diretrizes práticas para a implementação de estratégias integradas que favoreçam o desenvolvimento de SS em sala de aula.

Outro ponto observado é a ausência de indicações objetivas sobre quais competências de PC e SS devem ser priorizadas em cada contexto educativo, bem como de orientações sobre como aplicá-las de forma sistemática no ensino de programação. A falta de propostas que promovam, de maneira planejada, o desenvolvimento integrado de PC e SS, pode comprometer a formação integral dos estudantes.

Estudos indicam que apenas reunir alunos em grupos não garante o desenvolvimento de competências como colaboração e pensamento crítico; é necessário ensiná-las de forma intencional [da Silva Devincenzi et al. 2023]. A literatura também destaca que o domínio dessas habilidades está diretamente relacionado à empregabilidade dos futuros profissionais [Nudin et al. 2022, Schwartz 2016], o que reforça a importância de abordagens pedagógicas que integrem PC e SS no processo de ensino e aprendizagem.

Principais lacunas identificadas:

- **Foco concentrado no ensino fundamental:** há escassez de estudos em outros níveis, o que limita a diversidade de práticas pedagógicas.
- **Falta de integração entre habilidades técnicas e não técnicas:** poucas abordagens ou ferramentas promovem o desenvolvimento conjunto de PC e SS.
- **Ausência de diretrizes práticas para professores:** os estudos não apresentam orientações claras para apoiar a implementação de estratégias integradas em sala de aula.
- **Ênfase nas dificuldades dos alunos:** as intervenções priorizam os desafios dos estudantes, com escassa atenção às demandas pedagógicas dos docentes.

5. Considerações Finais

Este estudo realizou um MSL com o objetivo de investigar como o PC e as SS têm sido abordados no ensino de programação. A análise dos estudos permitiu identificar lacunas relevantes e orientar caminhos para futuras pesquisas. Verificou-se que a maioria das publicações está concentrada no ensino fundamental, com foco predominante em competências técnicas do PC, especialmente decomposição, abstração e algoritmos. Abordagens como o uso de robótica, Scratch e jogos educacionais são frequentes, mas, em geral, não incluem estratégias estruturadas para o desenvolvimento de SS.

Também foi constatada a escassez de diretrizes práticas que auxiliem professores no planejamento didático e na escolha de ferramentas para promover o PC e as SS de forma integrada. Além disso, os estudos analisados tendem a focar nas dificuldades enfrentadas pelos alunos, com pouca atenção às necessidades dos docentes.

Por fim, é importante destacar algumas limitações deste estudo, uma vez que a seleção baseada em critérios previamente definidos pode ter excluído estudos relevantes abordados de forma indireta ou em outros contextos. Também pode haver vieses de interpretação na triagem dos títulos, resumos e palavras-chave. Além disso, o recorte temporal de cinco anos, embora justificado, pode ter deixado de fora publicações importantes anteriores ao período analisado.

uma vez que a seleção baseada em critérios previamente definidos pode ter excluído estudos relevantes

Como trabalhos futuros, recomenda-se a realização de estudos empíricos que validem abordagens pedagógicas integradas de PC e SS em diferentes contextos, por meio de metodologias como design science research, estudos de caso ou quase-experimentais. Também se sugere a formulação de diretrizes práticas aplicáveis à docência, incluindo manuais, cursos de formação continuada e plataformas digitais de apoio. Além disso, a ampliação das pesquisas para diferentes níveis educacionais pode oferecer uma compreensão mais abrangente dos desafios e possibilidades do ensino de programação.

Referências

- Alex, K. (2009). *Soft skills*. S. Chand Publishing.
- Bié, E., Souto, E., Braga, D., Oliveira, E., and Carvalho, L. (2023). Ensino de programação para alunos nos anos escolares entre ensino fundamental ii e ensino médio: Um mapeamento sistemático. In *Anais do XXXIV Simpósio Brasileiro de Informática na Educação*, pages 414–427, Porto Alegre, RS, Brasil. SBC.
- BRASIL (2021). *Manual de Implementação das Competências Socioemocionais*. Disponível em: https://www.gov.br/mec/pt-br/brasil-na-escola/manual_de_implementacao_socioemocional.pdf. Acessado em: 13/11/2024.
- Brasil (2022). Parecer homologado despacho do ministro-mec. Disponível em: http://portal.mec.gov.br/index.php?Itemid=30192&alias=235511-pceb002-22&category_slug=fevereiro-2022-pdf&option=com_docman&view=download. Acessado em 07/04/2024.
- Carvalho, F. and Braga, M. (2022). Pensamento computacional na educação brasileira: um olhar segundo artigos do congresso brasileiro de informática na educação. *Revista Brasileira de Informática na Educação*, 30:237–261.
- Cruz, L. S. d., Santana, B. S. d., Bittencourt, R. A., Barreto, A. R., Gomes, K. C., and Santos, J. A. M. (2024). Efeito da promoção do pensamento computacional nas habilidades do século xxi: uma revisão sistemática da literatura. *Revista Brasileira de Informática na Educação*, 32:295–335. Acesso em: 5 jun. 2025.
- da Silva Devincenzi, S., Kwecko, V. R., da Costa, A. A., and de Lima Bicho, A. (2023). Desenvolvimento de soft skills: Um programa de formação universitária na era da capacitação 4.0. In *Anais do XXXIV Simpósio Brasileiro de Informática na Educação*, pages 1074–1085. SBC.
- Desidério, S. B., Lelis, M. R. L., Rodrigues, M. E. M., dos Santos, F. K. F., and Marques, A. B. (2023). Investigando o desenvolvimento de soft skills em projetos parceiros do programa meninas digitais: um estudo exploratório. In *Anais do XVII Women in Information Technology*, pages 171–181. SBC.
- Kitchenham, B. and Charters, S. (2007). Guidelines for performing systematic literature reviews in software engineering (ebse 2007-001). *Keele University and Durham University Joint Report*, pages 1–53.

- Nudin, S. R., Warsito, B., and Wibowo, A. (2022). Impact of soft skills competencies to predict graduates getting jobs using random forest algorithm. In *2022 1st International Conference on Information System & Information Technology (ICISIT)*, pages 49–54. IEEE.
- Petersen, K., Feldt, R., Mujtaba, S., and Mattsson, M. (2008). Systematic mapping studies in software engineering. In *12th international conference on evaluation and assessment in software engineering (EASE)*. BCS Learning & Development.
- Rege, A., de Castro Salgado, L. C., and Viterbo, J. (2023). Pensamento computacional no contexto da educação brasileira: um mapeamento sistemático da literatura nos diferentes n'iveis de ensino. *Revista Novas Tecnologias na Educação*, 21(2):413–422.
- Ribeiro, N., Martins, L., and Berssanette, J. (2022). Panorama brasileiro do ensino e aprendizagem de programação de computadores na educação básica. In *Anais do XXX Workshop sobre Educação em Computação*, pages 346–356, Porto Alegre, RS, Brasil. SBC.
- Rum, S. N. M. and Ismail, M. A. (2017). Metocognitive support accelerates computer assisted learning for novice programmers. *Journal of Educational Technology & Society*, 20(3):170–181.
- SBC (2025). *Grandes Desafios da Educação em Computação 2025-2035: Resumo Executivo*. Sociedade Brasileira de Computação (SBC), Porto Alegre.
- Schwartz, J. L. (2016). Preparing high school students for college while training engineering students in “soft skills”. In *2016 IEEE Integrated STEM Education Conference (ISEC)*, pages 112–115. IEEE.
- Versuti, F. M., Dalle Mulle, R. L., Guerreiro, C. A. R., Martins, F. P., and Peralta, D. A. (2020). Habilidades socioemocionais e tecnologias educacionais: revisão sistemática de literatura. *Revista Brasileira de Informática na Educação*, 28:1086–1104.
- Wing, J. M. (2006). Computational thinking. *Communications of the ACM*, 49(3):33–35.