

IA Generativa no Ensino de Programação: Mapeamento Sistemático, Inversão do Fluxo Cognitivo e Agenda para a Pesquisa Brasileira

Heloise Acco T. Bedin^{1,2}, Madianita Bogo Mariotti³, Patricia A. Jaques¹

¹ Universidade Federal do Paraná (UFPR) – Curitiba – Paraná – Brasil

² Instituto Federal do Paraná (IFPR) – Palmas – Paraná – Brasil

³ CEULP/ULBRA – Palmas – Tocantins – Brasil

heloise.acco@ufpr.br, madianitab@gmail.com, patricia.jaques@ufpr.br

Abstract. *This Systematic Literature Mapping analyzes 101 studies (2021–2025) on Generative Artificial Intelligence (GenAI) in programming education, retrieved from seven digital libraries and classified into four thematic categories. Technological Evaluation is present in 79.2% of studies and Psychosocial Impacts in only 5.9%. The central finding is the inversion of cognitive flow, in which students generate code before understanding the problem, with documented risk of competence illusion. Brazilian production represents 4% of the corpus, concentrated in Technological Evaluation, with no study primarily focused on Pedagogical Strategies or Psychosocial Impacts, a picture from which we derive a national research agenda of five structural gaps.*
Keywords Generative AI, Programming Education, Systematic Mapping, Brazilian Context, Computing Education

Resumo. *Este Mapeamento Sistemático da Literatura analisa 101 estudos (2021–2025) sobre Inteligência Artificial Generativa (GenAI) no ensino de programação, recuperados em sete bibliotecas digitais e classificados em quatro categorias temáticas. Avaliação Tecnológica está presente em 79,2% dos estudos e Impactos Psicossociais em apenas 5,9%. O achado central é a inversão do fluxo cognitivo, em que estudantes geram código antes de compreender o problema, com risco documentado de ilusão de competência. A produção brasileira representa 4% do corpus, concentrada em Avaliação Tecnológica e sem foco predominante em Estratégias Pedagógicas e Impactos Psicossociais, quadro do qual derivamos uma agenda nacional em cinco lacunas estruturais de pesquisa.*

Palavras-Chave IA Generativa, Ensino de Programação, Mapeamento Sistemático, Contexto Brasileiro, Informática na Educação

1. Introdução

O ensino de programação é amplamente reconhecido como um dos mais desafiadores na área da Computação [Moon et al. 2023, Banic et al. 2023], e a integração de Inteligência Artificial Generativa (GenAI) gerou um volume expressivo de pesquisas em curto espaço de tempo [Prather et al. 2023, Manorat et al. 2025]. Compreender o que essa comunidade prioriza investigar é condição necessária para orientar agendas nacionais e evitar a replicação acrítica de padrões internacionais em contextos distintos.

Cerca de 50% da pesquisa em Inteligência Artificial na Educação (AIED) origina-se da América do Norte, contra 26% da Ásia e 19% da Europa [Uema et al. 2025]. A diversidade geográfica é vital para criar ferramentas de IA culturalmente sensíveis e aplicáveis globalmente, o que exige que profissionais de regiões sub-representadas contribuam e adaptem esses achados aos seus contextos locais [Banihashem et al. 2025]. No campo específico do ensino de programação, uma revisão sistemática de 119 publicações sobre inteligência artificial e aprendizado de máquina documentou representação expressiva do Brasil, que ocupou o segundo lugar no *ranking* global com 11 artigos identificados [Manorat et al. 2025]. Esse volume contrasta com a baixa representação temática da pesquisa nacional nas sínteses internacionais e com as lacunas que este artigo documenta.

Este trabalho apresenta os resultados de um Mapeamento Sistemático da Literatura (MSL) [Kitchenham e Charters 2007, Petersen et al. 2015] conduzido em sete bibliotecas digitais, cobrindo 101 estudos revisados por pares (2021–2025), organizados em quatro contribuições complementares: (1) caracterização quantitativa da distribuição temática do corpus em quatro categorias, com análise de coocorrência e tendências temporais; (2) síntese das transformações cognitivas documentadas e das estratégias de integração pedagógica emergentes; (3) posicionamento bibliométrico da produção brasileira no corpus do mapeamento, revelando sua concentração temática e ausências estruturais; e (4) agenda de pesquisa nacional em cinco lacunas estruturais, derivada do cruzamento entre os padrões globais e as especificidades do contexto nacional.

2. Trabalhos Relacionados

As sínteses disponíveis sobre GenAI no ensino de programação convergem em dois achados, a concentração temática e as lacunas de equidade. Uma revisão de 119 artigos (2012–2024) identificou concentração em monitoramento de desempenho, avaliação e *feedback*, com lacuna acentuada em design de cursos [Manorat et al. 2025]. Outra síntese, de 38 estudos (2022–2024), constatou que questões de acesso e equidade permanecem subexploradas [Aruleba et al. 2025]. Dois modos predominantes de integração já foram documentados, o uso pelos estudantes para escrita e depuração de código e o uso pelos professores para criação de recursos didáticos e *feedback* automatizado [Prather et al. 2023].

Nenhum desses trabalhos combina distribuição temática com recorte da produção brasileira, nem aborda o viés anglófono que sub-representa sistematicamente o Sul Global [Slimi 2026]. No contexto lusófono, documenta-se que a pesquisa brasileira e portuguesa carece de visibilidade internacional e que a agenda de AIED precisa considerar as realidades socioeconômicas de diferentes contextos [Teodoro e Freire 2025]. Este artigo distingue-se desses trabalhos ao posicionar a produção brasileira no mapa temático global e derivar uma agenda estruturada a partir das lacunas do contexto nacional.

3. Metodologia

Este estudo adota o protocolo de Mapeamento Sistemático de Literatura [Kitchenham e Charters 2007, Petersen et al. 2015], reportado segundo as diretrizes PRISMA 2020 [Page et al. 2021]. A busca foi conduzida em sete bibliotecas digitais (ACM, DBLP, IEEE Xplore, ScienceDirect, Scopus, SOL/SBC e SpringerLink),

com uma *string* organizada em três eixos conceituais (domínio de conhecimento, tecnologias de GenAI e contexto educacional) e janela de publicação de 2020 a outubro de 2025, quando as buscas foram encerradas. A *string* completa, suas adaptações por base, os critérios de elegibilidade, o instrumento de avaliação de rigor, o diagrama de fluxo PRISMA 2020 e os registros de triagem por estudo estão disponíveis no depósito público dos artefatos [Bedin et al. 2026b].

Foram incluídos os estudos que relatassem o uso de GenAI na aprendizagem de programação, descrevessem ferramentas de GenAI aplicadas ao ensino de programação, apresentassem estratégias pedagógicas ou metodologias de integração, propusessem métricas para avaliar seu impacto na aprendizagem ou discutissem tendências, desafios ou limitações de sua adoção. Foram excluídos os registros que não se relacionassem à aprendizagem de programação assistida por GenAI, que estivessem em idioma distinto do português ou do inglês, que tivessem sido publicados antes de 2020, que não constituíssem artigo completo revisado por pares ou que fossem duplicatas.

A seleção foi conduzida independentemente por duas pesquisadoras na plataforma Porifera [de Campos et al. 2023], em três filtros sequenciais, com divergências resolvidas por consenso. O filtro F1 consistiu na triagem por título e resumo. O filtro F2 consistiu na análise de texto completo, verificando os critérios de inclusão e a presença de avaliação educacional empírica. O filtro F3 consistiu na avaliação da qualidade com instrumento de dez critérios binários (escore 0–10), que excluiu estudos empíricos com escore 0–4, enquanto estudos conceituais, teóricos ou normativos foram avaliados por critérios adaptados à sua natureza não empírica e não submetidos a esse corte. Dos 102 estudos avaliados em F3, 78 (76,5%) alcançaram alto rigor (8–10), 23 (22,5%) rigor médio (5–7) e 1 (1,0%) rigor insuficiente, sendo este último excluído. A concordância inter-avaliadoras atingiu $\kappa = 0,978$ em F1 e $\kappa = 0,914$ em F2 [Landis e Koch 1977], com concordância total em F3. Na extração de dados, o Google NotebookLM atuou como apoio preliminar, sem participar de decisões de inclusão ou exclusão, sob protocolo validado de uso responsável de GenAI em revisões sistemáticas [Bedin et al. 2026a], com verificação humana obrigatória dos 101 estudos e triangulação das informações críticas nos artigos originais pelas duas pesquisadoras.

Os quatro objetivos do estudo foram operacionalizados de modo distinto. Em **(OBJ1)**, os estudos foram classificados indutivamente em quatro categorias temáticas não mutuamente exclusivas (Avaliação Tecnológica e Estudos Empíricos, Design de Sistemas, Estratégias Pedagógicas e Impactos Psicossociais), com análise de coocorrência e tendências temporais. Quando um estudo tangenciava mais de uma categoria, a atribuição seguiu o critério de foco predominante, definido pelo objeto central da investigação e pelo desenho metodológico adotado. Em **(OBJ2)**, a síntese das transformações cognitivas e pedagógicas emergentes foi derivada da análise qualitativa das categorias Avaliação Tecnológica e Estratégias Pedagógicas. Em **(OBJ3)**, os estudos brasileiros foram identificados por afiliação institucional e analisados quanto à categoria, ferramenta, nível educacional e resultado predominante. Em **(OBJ4)**, os padrões temáticos foram cruzados com as especificidades do contexto educacional brasileiro, derivando cinco lacunas estruturais de pesquisa.

4. Resultados

O processo de busca e triagem retornou 1.331 registros em sete bases digitais, dos quais 101 estudos integraram o corpus final após a remoção de duplicatas e a aplicação dos três filtros sequenciais. As bases que mais alimentaram o corpus final são o IEEE Xplore (35 estudos; 34,7%), a ACM (28; 27,7%) e a DBLP (14; 13,9%). Os estudos incluídos distribuem-se entre 2021 e 2025, sem que nenhum registro de 2020 tenha sido aprovado na triagem.

4.1. Objetivo 1 – Distribuição Temática do Corpus

A classificação temática dos 101 estudos revelou uma distribuição assimétrica entre as quatro categorias, com concentração expressiva em Avaliação Tecnológica e Estudos Empíricos e presença marginal de Impactos Psicossociais, conforme a Tabela 1.

Tabela 1. Distribuição de estudos por categorias e combinações

Categoria isolada	N	%	Combinação	N	%
(1) Avaliação Tecn. e Estudos Empíricos	36	35,6	(1)+(2) Aval.+Design	21	20,8
(2) Design de Sistemas	6	5,9	(1)+(3) Aval.+Estratégias	10	9,9
(3) Estratégias Pedagógicas	13	12,9	(1)+(4) Aval.+Psicossocial	4	4,0
(4) Impactos Psicossociais	1	1,0	(2)+(3) Design+Estratégias	1	1,0
			(1)+(2)+(3) Aval.+Des.+Estrat.	8	7,9
			(1)+(2)+(4) Aval.+Des.+Psicos.	1	1,0
Total				101	100,0

Avaliação Tecnológica está presente em 80 estudos (79,2%, diretos e combinados), dominando o corpus. Design de Sistemas e Estratégias Pedagógicas ocupam posições intermediárias, com 37 (36,6%) e 32 estudos (31,7%), respectivamente. Impactos Psicossociais aparecem em apenas 6 estudos (5,9%, diretos e combinados), configurando a lacuna mais expressiva, pois o campo avalia extensivamente as propriedades das ferramentas e quase não examina seus efeitos sobre o bem-estar, a motivação e os processos cognitivos dos aprendizes. Quarenta e cinco estudos (44,6%) abrangem múltiplas categorias, com a combinação Avaliação + Design sendo a mais frequente (21 estudos). A coocorrência entre Design de Sistemas e Impactos Psicossociais restringe-se a um único estudo [Kuttal et al. 2021], que avalia os efeitos de um agente conversacional sobre a autoeficácia em programação em pares, com amostra mista de estudantes universitários e programadores profissionais. A Figura 1 apresenta a evolução anual por categoria. O campo cresceu acentuadamente a partir de 2023, com 2024 como pico (41 estudos). Os dados de 2025 são parciais, referentes ao período de janeiro a outubro, e não permitem comparação direta com os anos anteriores.

Mais relevante do que o volume é a mudança de composição, com a categoria (3) Estratégias Pedagógicas passando de 1 estudo em 2022 para 8 em 2023 e consolidando-se em 12 estudos em 2024 e 11 em 2025, sinalizando uma transição do campo de avaliações técnicas isoladas para investigações orientadas à integração pedagógica da GenAI no ensino de programação.

4.2. Objetivo 2 – Transformações Cognitivas e Estratégias de Integração

O achado mais consistente e teoricamente relevante do corpus é a **inversão do fluxo cognitivo tradicional de aprendizagem de programação** [Robins et al. 2003,

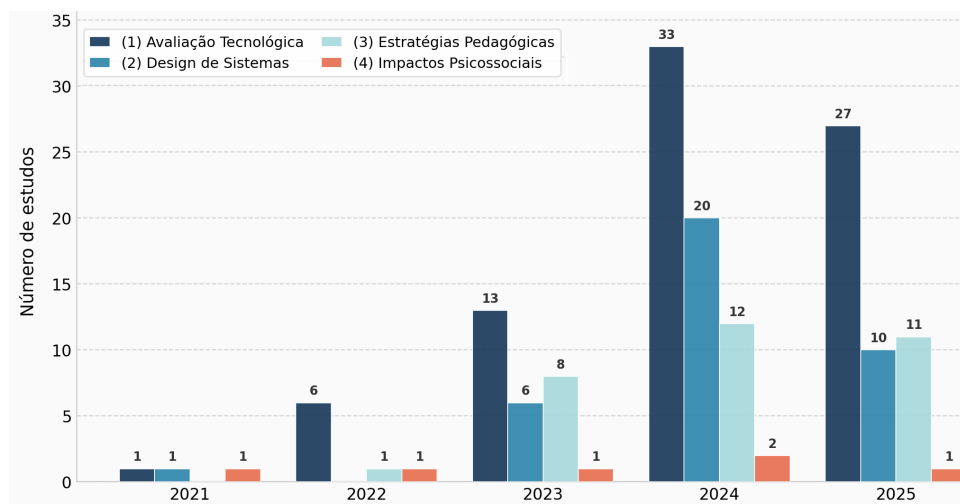


Figura 1. Evolução dos estudos por categoria temática (2021–2025). Um estudo com múltiplas categorias é contabilizado em cada categoria que possui.

Yilmaz e Karaoglan Yilmaz 2023]. Tradicionalmente, a pedagogia de programação segue uma progressão linear, que parte da (1) compreensão conceitual do problema, passa pela (2) construção do modelo mental da solução e pela (3) codificação, e culmina na (4) depuração e teste. Com GenAI, essa sequência é sistematicamente invertida, pois o aprendiz *gera* código funcional antes de *compreender* o problema, a ponto de a ferramenta permitir que o estudante crie sem entender [Pesovski et al. 2024], sabendo o que o código faz sem saber como ou por que funciona.

Na cadeia mediada por GenAI, o aprendiz (a) interage com o grande modelo de linguagem (do inglês *Large Language Model*, LLM) para gerar código, (b) inspeciona e tenta explicar a solução produzida, (c) modifica e adapta conforme a necessidade, e (d) valida e testa o resultado, sequência que reproduz a lógica de uso e modificação documentada no corpus [Kazemitabaar et al. 2023]. Essa inversão redefine o papel cognitivo do aprendiz, de *autor* que constrói raciocínio para *editor* que avalia e adapta código externo [Paludo e Montresor 2024]. Essa transição exige habilidades metacognitivas mais sofisticadas do que o ensino tradicional demanda, pois o aprendiz precisa avaliar código não autoral, identificar limitações e integrar conscientemente as saídas da IA ao seu próprio raciocínio [Paludo e Montresor 2024].

A inversão também altera o fluxo de *feedback*, pois, enquanto a pedagogia tradicional posiciona a compilação e execução como *produto* do entendimento, a GenAI as torna *ponto de partida* [Pesovski et al. 2024]. O artefato é executado com sucesso, e o estudante pode confundir esse resultado com a compreensão conceitual [Prather et al. 2024], risco detalhado na subseção seguinte.

4.2.1. Respostas Pedagógicas à Inversão do Fluxo Cognitivo

Dois *frameworks* do corpus operacionalizam essas respostas. O modelo **TAIC** (*Teacher–AI Co-Creation*) [Zhuang et al. 2025] organiza a colaboração docente-IA em quatro fases, a definição de objetivos a partir da análise de dados dos estudantes, o design do conteúdo

por instrução baseada em projetos, a construção de cenários interativos e a verificação dos resultados de aprendizagem. Em cada fase, a IA executa tarefas de suporte de alto volume, enquanto o professor retém as decisões pedagógicas críticas e assume a liderança no design instrucional, pois a IA apresenta limitações sistêmicas ao integrar conteúdos teóricos a projetos complexos.

A **Taxonomia de Bloom Reversa** [Pesovski et al. 2024] complementa o TAIC ao inverter intencionalmente a hierarquia cognitiva tradicional, pois, em vez de progredir de habilidades de ordem inferior para superior, os estudantes *iniciam* pela criação de software assistida por IA, frequentemente sem experiência prévia de codificação. A partir das soluções produzidas, percorrem as etapas de avaliação, análise crítica e melhoria, chegando à compreensão dos conceitos subjacentes e à programação autônoma.

A eficácia pedagógica de ambos os modelos depende criticamente da mediação docente, pois, sem técnicas de engenharia reversa guiadas pelo professor, os estudantes tendem a solicitar soluções diretas à IA, reduzindo oportunidades de pensamento profundo [Pesovski et al. 2024]. Evidências empíricas confirmam esse risco, uma vez que 90% dos estudantes de um dos estudos do corpus reconheceram que a dependência excessiva restringe a criatividade e gera compreensão superficial dos conceitos de programação [Park et al. 2025].

4.2.2. A Ilusão de Competência como Risco Central

A inversão cognitiva introduz o risco mais documentado do corpus, a **ilusão de competência** [Koriat 1997, Amoozadeh et al. 2024], viés pelo qual o aprendiz superestima sua compreensão porque avalia o próprio conhecimento enquanto a solução ainda está visível, sensação de domínio que se revela injustificada quando ele precisa agir de forma independente. Cerca de um terço dos estudantes submete descrições completas de problemas à IA sem esforço prévio e raramente verifica a veracidade das soluções recebidas [Amoozadeh et al. 2024]. Essa falsa percepção é denominada “competência emprestada” (*borrowed competence*) [Barcaui 2025], pois a fluência do código gerado infla o domínio percebido sem consolidação real da memória, e seu efeito longitudinal é descrito como “dívida cognitiva acumulada” [Kosmyna et al. 2025], em que o esforço mental diferido resulta em queda de desempenho quando o suporte da IA é removido. Estudantes com mais dificuldades são os mais expostos à ilusão, seja por concluírem tarefas sem entender por que funcionam [Prather et al. 2024], seja por recorrerem mais à IA por medo de falhar [Margulieux et al. 2024], o que agrava déficits em habilidades metacognitivas essenciais à programação [Gong et al. 2025].

Os próprios estudantes reconhecem que a IA pode torná-los mais dependentes e que os fundamentos deveriam ser aprendidos sem assistência algorítmica [Valový e Buchalceva 2023]. Essa dissonância entre o que sabem ser correto e o comportamento real está documentada no corpus, que distingue dois perfis comportamentais [Dawson et al. 2025]. Os estudantes *orientados ao significado* são cautelosos e buscam compreensão profunda, enquanto os *orientados à aplicação* usam a IA intensivamente para resultados rápidos e ficam mais expostos à ilusão de competência.

4.2.3. Benefícios Documentados e Resultados Mistos

Os riscos cognitivos não esgotam o quadro empírico. A distribuição dos resultados no corpus revela heterogeneidade estrutural, pois 28,7% dos estudos reportam resultados predominantemente positivos, 49,5% reportam resultados mistos, em que benefícios de curto prazo coexistem com riscos cognitivos, 19,8% não avaliam efetividade, incluindo *position papers*, propostas conceituais, estudos de viabilidade técnica e investigações exploratórias, e apenas 2,0% reportam resultados predominantemente negativos.

Entre os estudos com resultados positivos, a literatura empírica reporta ganhos substanciais em produtividade e qualidade do código. Iniciantes apoiados por IA alcançaram taxas de conclusão e pontuações superiores em tarefas de autoria de código, sem redução no desempenho em testes posteriores de retenção [Kazemitabaar et al. 2023]. Em ensaio controlado, alunos apoiados por IA concluíram 90% dos projetos, contra 60% no grupo de controle, e resolveram *bugs* em quase metade do tempo [Masetty e Vallamulla 2025]. A GenAI opera também como tutor virtual, reduzindo barreiras de entrada e o estresse associado a impasses sintáticos e lógicos [Valový e Buchalcevova 2023, Zhao et al. 2025], e sistemas pedagogicamente customizados superam consistentemente os LLMs genéricos em adequação pedagógica, acurácia e utilidade [Feng et al. 2024].

A prevalência de resultados mistos, contudo, evidencia que a eficácia da GenAI não é uma propriedade intrínseca da tecnologia, mas um fenômeno condicional e emergente, dependente de como a ferramenta é pedagogicamente mediada [Zhuang et al. 2025].

A síntese dos resultados permite derivar diretrizes operacionais para a integração pedagógica da GenAI no ensino de programação. A **orientação estruturada (D1)** é condição necessária para resultados positivos, pois o uso não supervisionado é associado a dependência e à ilusão de competência [Amoozadeh et al. 2024]. O **modelo bifásico de progressão (D2)** estabelece que a aprendizagem sem GenAI precede a exposição às ferramentas [Vadaparty et al. 2025, Zhou et al. 2025], pois iniciantes com exposição prematura experienciam erosão de fundamentos, enquanto estudantes avançados utilizam a IA estrategicamente. O **letramento em *prompting* (D3)**, formular e estruturar instruções para guiar modelos generativos, é uma nova literacia essencial [Cheng et al. 2025], para a qual já existem exercícios específicos, como os *Prompt Problems* [Denny et al. 2024], e cuja urgência é ampliada no Brasil pelas barreiras linguísticas discutidas na Seção 4.2.5. No **equilíbrio entre automação e autonomia (D4)**, o *framework Use-Modify-Create* [Kazemitabaar et al. 2023] preserva o esforço cognitivo produtivo ao converter tarefas de criação em atividades de uso e modificação, enquanto o *paradigma do hipócrita* [Dawson et al. 2025], no qual estudantes apresentam aos pares as formas mais e menos efetivas de usar LLMs, promove a autorregulação pelo confronto entre práticas. Os **LLMs pedagogicamente customizados (D5)** superam os genéricos por alinhamento explícito com os objetivos de aprendizagem [Borchers e Shou 2025], como o CourseAssist [Feng et al. 2024], que infere a intenção do aprendiz via Geração Aumentada por Recuperação (do inglês *Retrieval-Augmented Generation*, RAG) e fornece dicas em vez de soluções completas.

4.2.4. Estratégias de Integração Curricular

Além dos princípios gerais expressos nas diretrizes D1–D5, o corpus documenta modelos concretos de integração curricular, que variam em escopo e grau de customização pedagógica, mas convergem para um padrão dominante.

O padrão de integração mais amplamente adotado é o modelo bifásico de progressão [Vadaparty et al. 2025, Prather et al. 2024, Valový e Buchalceva 2023, Groothuijsen et al. 2024]. Na Fase 1 (Fundamentos sem IA), os estudantes praticam sintaxe, depuração manual e raciocínio algorítmico por meio de exercícios de *tracing* e pseudocódigo, enquanto o professor estrutura aulas expositivas com *quizzes* formativos e prática guiada. A IA fica desabilitada ou restrita a explicações conceituais de alto nível. Na Fase 2 (Aplicação com IA), os estudantes utilizam LLMs para codificação assistida, depuração baseada em dicas e atividades reflexivas. O professor fornece modelos de *prompt*, *feedback human-in-the-loop* e orientação ética, enquanto a IA assume o papel de tutor adaptativo com RAG e *prompts* metacognitivos.

Estudos comparativos entre dinâmicas mediadas por IA e práticas tradicionais revelam resultados contrastantes. Atividades de interação dialógica baseadas no ChatGPT superaram a programação em pares tradicional nas dimensões comportamental, cognitiva e emocional [Zhang et al. 2025], com o LLM atuando como andaime (*scaffolding*) dinâmico que reduziu comportamentos irrelevantes e minimizou a ansiedade. Em contrapartida, ao liberar a IA durante a programação em pares, os estudantes tendem a delegar o papel de navegador ao LLM [Groothuijsen et al. 2024, Simaremare et al. 2024], comprometendo a colaboração autêntica entre colegas e levantando preocupações sobre dependência excessiva.

Além do modelo bifásico, o corpus documenta implementações variadas, de ferramentas externas complementares [Groothuijsen et al. 2024, Gandhi e Muldner 2025] a sistemas especializados como o *KOGI* [Kuramitsu et al. 2023], que integra assistência baseada em ChatGPT. Três recomendações são propostas para o uso efetivo do ChatGPT [Ramabu e Malebane 2024], a definição de parâmetros precisos do problema, incluindo objetivos do algoritmo e formatos de entrada e saída, o uso da IA para avaliação qualitativa do código do estudante e a geração de especificações de problemas para novos exercícios e testes.

4.2.5. Implicações das Diretrizes para o Currículo Brasileiro

As diretrizes D1–D5 foram derivadas predominantemente de contextos anglófonos com infraestrutura consolidada. D1 (orientação estruturada) e D4 (equilíbrio entre automação e autonomia) aplicam-se ao contexto nacional sem adaptações estruturais relevantes, pois dependem primariamente de decisões pedagógicas do professor, não de infraestrutura ou proficiência linguística. D2, D3 e D5, contudo, pressupõem condições que o contexto brasileiro frequentemente não possui. D2 (fundamentos sem IA) colide com a realidade de estudantes que podem ter acesso à GenAI *antes* de qualquer fundamento consolidado, especialmente em contextos de alta desigualdade. D3 (letramento em *prompting*) esbarra em barreiras linguísticas estruturais, uma vez que os LLMs possuem maior capacidade de compreensão de intenção em inglês, o que penaliza estudantes que formulam *prompts*

em português [Barros et al. 2025], somando-se à barreira do idioma já documentada no ensino de programação no Brasil [Medeiros et al. 2020].

Os cursos introdutórios de programação nas instituições de ensino públicas brasileiras divergem da tendência internacional [Silva et al. 2023], o que agrava a restrição de D2, pois, enquanto no exterior predominam Java e Python, no Brasil prevalecem a linguagem C e o paradigma procedural. Diferentemente de linguagens de alto nível como Python ou Java, a linguagem C exige que o programador lide com tarefas de baixo nível, incluindo interação direta com o *hardware* e gerenciamento manual de memória, o que demanda um modelo mental sólido da arquitetura do computador e dos sistemas operacionais [Park et al. 2025]. Quando um LLM gera código procedural correto em C instantaneamente, resolve exatamente essa fase de “luta cognitiva”, com o risco de privar o estudante do processo formativo que consolida os fundamentos algorítmicos.

A alta competência das ferramentas em programação procedural agrava esse risco, pois estudos confirmam que LLMs identificam erros lógicos em C [MacNeil et al. 2024], geram código funcional correto em tarefas introdutórias brasileiras [Carvalho e Oliveira 2024] e obtêm aprovação em provas de calouros em C/C++ [Logozar et al. 2023]. O resultado é que o aluno pode submeter código funcional sem ter desenvolvido o raciocínio procedural subjacente, o que é especialmente problemático no contexto brasileiro, em que a aprovação nesses cursos é pré-requisito para toda a cadeia curricular subsequente.

A terceira limitação estrutural, D5 (sistemas customizados), pressupõe conectividade de alta velocidade e dispositivos de ponta que grande parte das instituições públicas brasileiras não possui [Uema et al. 2025, Medeiros et al. 2020]. Pesquisadores têm proposto *frameworks* de contingência, como o *AIED Unplugged* [Uema et al. 2025], que buscam adaptar soluções de IA para funcionar de forma assíncrona ou em *smartphones* de baixo custo, mitigando o risco de a inteligência artificial converter-se em uma nova fonte de exclusão digital no Sul Global.

4.3. Objetivo 3 – Posicionamento da Produção Brasileira

Apesar de o Brasil ocupar posição expressiva no volume global de publicações sobre IA no ensino de programação [Manorat et al. 2025], sua representação no corpus deste mapeamento é reduzida e tematicamente concentrada. Quatro estudos do corpus envolvem afiliação institucional brasileira, o que corresponde a 4,0% dos 101 estudos, distribuídos em apenas duas das quatro categorias temáticas, conforme a Tabela 2.

Tabela 2. Estudos com afiliação institucional brasileira no corpus.

Estudo	Cat.	Instituição	Ferramenta	Aplicação
[Dos Santos e Cury 2023]	(1)	UFES	ChatGPT	Com estudantes brasileiros
[Norton et al. 2024]	(2)	UEA	LLM + AVA	Com estudantes brasileiros (TEA)
[Carvalho e Oliveira 2024]	(1)	Mackenzie	ChatGPT, Copilot, Gemini	Avaliação sem aplicação local
[Phung et al. 2023]	(1)	UFCEG	ChatGPT, GPT-4	Colaboração internacional sem aplicação local

A produção brasileira concentra-se em Avaliação Tecnológica (três estudos) e Design de Sistemas (um estudo), sem nenhum estudo cujo foco predominante seja Estratégias Pedagógicas ou Impactos Psicossociais. A natureza das contribuições é ainda mais restritiva do que sua distribuição temática, pois apenas dois dos quatro estudos foram efetivamente aplicados em contexto real de ensino com estudantes brasileiros [Dos Santos e Cury 2023, Norton et al. 2024], ambos com resultados positivos condicionados à mediação docente.

Essa concentração temática reflete restrições estruturais históricas. A superlotação de turmas, os laboratórios deficientes e a falta de acesso estável à internet são barreiras recorrentes no ensino superior brasileiro de programação [Medeiros et al. 2020], dificultando intervenções pedagógicas longitudinais e estudos empíricos de larga escala. A dependência de conectividade pelas ferramentas atuais de GenAI corre o risco de aprofundar a pobreza digital [Portela et al. 2024], motivando respostas estruturais como o *AIED Unplugged*.

A literatura nacional recente sinaliza, contudo, diversificação emergente. Embora a maioria das pesquisas nacionais ainda utilize o ChatGPT sem abordagem pedagógica estruturada, iniciativas que integram IA a metodologias ativas, como programação em pares com IA como colega virtual, já emergem na literatura [Silva et al. 2024]. O quadro de ausências é agravado pela especificidade curricular já discutida na Seção 4.2.5, uma configuração distinta daquela em que as ferramentas de GenAI foram majoritariamente avaliadas no corpus.

5. Agenda de Pesquisa Nacional

O cruzamento entre a distribuição temática do corpus, as transformações cognitivas documentadas e o perfil da produção brasileira permite derivar cinco lacunas estruturais prioritárias, focadas nas categorias temáticas e curriculares sub-representadas na literatura nacional e internacional.

Lacuna 1 – Avaliação Tecnológica em contexto procedural e língua portuguesa: embora a Avaliação Tecnológica seja a categoria em que a produção brasileira mais se faz presente no corpus, a investigação nacional do desempenho de ferramentas GenAI em tarefas com enunciado e interação em português, no paradigma procedural em C, permanece restrita a avaliações pontuais de correteude em exercícios introdutórios [Carvalho e Oliveira 2024], sem cobertura sistemática de diferentes níveis de dificuldade, tipos de tarefa e critérios pedagógicos, ainda que a competência dos LLMs nessa linguagem já esteja confirmada (Seção 4.2.5). Avaliar sistematicamente como diferentes LLMs se comportam nesse cenário é condição necessária para fundamentar políticas de integração curricular no Brasil.

Lacuna 2 – Design de Sistemas com RAG Curricular: dos seis estudos dedicados exclusivamente a Design de Sistemas (Tabela 1), a maioria foca em LLMs genéricos treinados em bases internacionais, cujo conhecimento paramétrico não está ancorado em materiais de referência verificáveis do contexto local [Lewis et al. 2020]. A lacuna brasileira é o design de arquiteturas RAG alimentadas por materiais didáticos locais, apostilas e enunciados de juízes online de universidades públicas, garantindo alinhamento à ementa do professor e ao paradigma procedural em C.

Lacuna 3 – Impactos Psicossociais em populações brasileiras: com apenas

5,9% do corpus abordando Impactos Psicossociais, isoladamente ou em combinação, e nenhum estudo brasileiro nessa categoria, a dimensão humana da integração de GenAI no ensino de programação permanece sem investigação sistemática no contexto nacional. Investigar a ansiedade de desempenho, a autoeficácia e o bem-estar em estudantes de programação usando GenAI, especialmente em estudantes de primeira geração universitária e em contextos de alta desigualdade, é prioritário diante de um cenário de abandono historicamente crítico e ainda atual. Dos ingressantes de 2011 na área geral de Ciências, Matemática e Computação, apenas 5,5% concluíram o curso [INEP 2013] e, mais de uma década depois, a taxa de desistência acumulada em licenciaturas de exatas correlatas, como Matemática e Física, permanece em 70% e 73%, respectivamente [INEP 2024].

Lacuna 4 – Políticas de Avaliação: as cinco diretrizes de integração pedagógica identificadas no corpus (D1–D5, Seção 4.2) pressupõem adaptações profundas nas formas de avaliar. Faltam estudos sobre como adaptar as políticas de avaliação institucional para focar no processo de desenvolvimento algorítmico, e não apenas na submissão de código funcional que pode ter sido integralmente gerado por IA, reformulando critérios que privilegiem raciocínio, documentação de decisões e depuração manual, prioridade prática para a qual a pesquisa nacional ainda não oferece respostas sistemáticas.

Lacuna 5 – Longitudinalidade: os dois estudos brasileiros do corpus efetivamente aplicados com estudantes são de curto prazo, restritos a uma intervenção ou a um semestre. Não há evidências sobre os efeitos de longo prazo do uso de GenAI no ensino de programação em instituições brasileiras.

Quatro das lacunas compartilham alta urgência, definida aqui pela representação zero ou mínima do tema na produção nacional e pelo potencial de impacto sobre o perfil de evasão documentado. A Lacuna 5 (longitudinalidade) é classificada como urgência média, por depender de condições institucionais de longo prazo, independentes da agenda de pesquisa.

6. Ameaças à Validade

A interpretação dos achados requer atenção a quatro ameaças à validade inerentes ao desenho de mapeamento sistemático adotado.

Validade de construto: a classificação temática envolveu julgamento interpretativo. Taxonomias alternativas poderiam resultar em agrupamentos distintos, especialmente em áreas de sobreposição. Para minimizar a subjetividade, as categorias foram derivadas indutivamente a partir do corpus e discutidas iterativamente pelas duas pesquisadoras até consenso.

Validade interna: a identificação de estudos brasileiros por afiliação ou contexto pode ter omitido colaborações internacionais sem declaração explícita. O uso do NotebookLM na extração pode ter introduzido viés na identificação de elementos relevantes, risco mitigado por três controles. Primeiro, a ferramenta não participou de nenhuma decisão de inclusão ou exclusão, todas tomadas pelas duas pesquisadoras nos três filtros. Segundo, houve verificação humana obrigatória da totalidade dos 101 estudos, com triangulação nas fontes originais. Terceiro, a concordância inter-avaliadoras próxima do máximo ($\kappa = 0,978$ e $\kappa = 0,914$) evidencia a consistência do julgamento humano.

Validade externa: a predominância de estudos em língua inglesa no corpus limita a generalização [Banihashem et al. 2025]. A restrição linguística (inglês e português) pode ter excluído estudos relevantes em outros idiomas [Teodoro e Freire 2025]. Os achados das Seções 4.2 e 4.3 refletem primariamente contextos de alta renda com infraestrutura consolidada, o que reforça a pertinência da agenda derivada na Seção 5. Por fim, as buscas foram encerradas em outubro de 2025 e não foram reexecutadas, de modo que estudos publicados posteriormente não integram o corpus, limitação relevante em um campo de evolução acelerada, no qual novas gerações de modelos podem alterar os padrões documentados.

Validade de conclusão: a proporção reduzida de resultados negativos (2,0%) sugere viés de publicação, enquanto a sub-representação de Impactos Psicossociais (5,9%) sugere viés de agenda de pesquisa no campo. Para reduzir esses riscos, resultados contraditórios ou neutros foram explicitamente preservados na síntese.

7. Conclusão

Este mapeamento sistemático de 101 estudos (2021–2025) demonstrou que a pesquisa em GenAI e ensino de programação é estruturalmente desequilibrada, com Avaliação Tecnológica presente em 80 estudos (79,2%) e Impactos Psicossociais em apenas 6 (5,9%). O campo avalia extensivamente as ferramentas e quase não examina seus efeitos humanos, assimetria que começa a ser corrigida com o crescimento de Estratégias Pedagógicas a partir de 2023, categoria que passou de 1 para 8 estudos entre 2022 e 2023 e se manteve em patamar elevado em 2024 e 2025.

O achado teórico central, a inversão do fluxo cognitivo, tem implicações curriculares específicas para o Brasil, pois, em cursos introdutórios com predominância de linguagem C procedural, a alta competência dos LLMs em gerar código funcional instantaneamente agrava o risco de ilusão de competência ao suprimir a fase de “luta cognitiva” que consolida o raciocínio algorítmico. O modelo TAIC e a Taxonomia de Bloom Reversa oferecem respostas pedagógicas a essa inversão, mas pressupõem formação docente e infraestrutura que o contexto brasileiro raramente oferece. As cinco diretrizes baseadas em evidências (D1–D5) e o modelo bifásico de progressão fornecem um roteiro operacional, condicionado à mediação ativa do professor.

Para o Brasil, o cenário é de contraste, pois, apesar da representação expressiva em volume de publicações [Manorat et al. 2025], as quatro contribuições nacionais do corpus restringem-se a duas das quatro categorias temáticas, e apenas duas delas foram efetivamente aplicadas com estudantes brasileiros. A agenda em cinco lacunas convida a comunidade de Informática na Educação a diversificar sua produção em quatro frentes prioritárias, avaliações tecnológicas em português e paradigma procedural, design de sistemas RAG alinhados ao currículo nacional, investigação dos impactos psicossociais em estudantes brasileiros e reformulação das políticas de avaliação frente ao uso de GenAI, às quais se soma a necessidade de estudos longitudinais que dependem de condições institucionais de prazo mais longo.

Disponibilidade de Artefatos

O protocolo completo do mapeamento, incluindo a *string* de busca com as adaptações por base, os critérios de inclusão e exclusão, o instrumento de avaliação

de rigor metodológico, o diagrama de fluxo PRISMA 2020 e os registros de decisão de triagem para as 1.331 referências recuperadas, está disponível em <https://doi.org/10.5281/zenodo.21342736>.

Declaração sobre uso de Inteligência Artificial

O Google NotebookLM auxiliou na extração preliminar de dados, com validação manual integral. O Claude (Anthropic) apoiou revisão linguística. Todo o conteúdo foi criticamente revisado e finalizado pelas autoras, que assumem total responsabilidade pelo conteúdo apresentado.

Agradecimentos

O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Código de Financiamento 001.

Referências

- Amoozadeh, M., Nam, D., Prol, D., Alfageeh, A., Prather, J., Hilton, M., Srinivasa Ragavan, S., e Alipour, A. (2024). Student-AI interaction: A case study of CS1 students. In *Proceedings of the 24th Koli Calling International Conference on Computing Education Research*. ACM.
- Aruleba, K., Oyelere, S. S., Obaido, G. R., e Sanusi, I. T. (2025). A scoping review of student and educator engagement with large language models in introductory programming education. In *UKICER 2025*. ACM.
- Banic, B., Konecki, M., e Konecki, M. (2023). Pair programming education aided by ChatGPT. In *MIPRO 2023*. IEEE.
- Banihashem, S. K., Bond, M., Bergdahl, N., Khosravi, H., e Noroozi, O. (2025). A systematic mapping review at the intersection of artificial intelligence and self-regulated learning. *International Journal of Educational Technology in Higher Education*, 22(1):50.
- Barcaui, A. (2025). ChatGPT as a cognitive crutch: Evidence from a randomized controlled trial on knowledge retention. *Social Sciences & Humanities Open*, 12:102287.
- Barros, A., Monteiro, M., Rodrigues, L., Dermeval, D., Isotani, S., e Bittencourt, I. I. (2025). Evaluating large language model quality in resource-constrained environments: An educational stakeholders' survey on accuracy, completeness, and readability in Brazil. In *Artificial Intelligence in Education*, Lecture Notes in Computer Science, pages 218–232. Springer.
- Bedin, H. A. T., Canedo, E. D., e Jaques, P. A. (2026a). Responsible use of generative AI in systematic reviews: A practical methodological guideline. In *CSEDU 2026*, pages 63–74. SCITEPRESS.
- Bedin, H. A. T., Mariotti, M. B., e Jaques, P. A. (2026b). Artefatos suplementares do mapeamento sistemático sobre IA Generativa no ensino de programação. Zenodo. <https://doi.org/10.5281/zenodo.21342736>.

- Borchers, C. e Shou, T. (2025). Can large language models match tutoring system adaptivity? A benchmarking study. In *AIED 2025*, volume 15878, pages 407–420. Springer.
- Carvalho, S. e Oliveira, I. C. (2024). O uso e o potencial de inteligências artificiais no aprendizado de algoritmos e programação de computadores. In *Anais do XXXV Simpósio Brasileiro de Informática na Educação*, pages 3049–3058. SBC.
- Cheng, G., Wong, W., Luo, L., e Yu, M. (2025). Integrating a scaffolding-based, LLM-driven chatbot into programming education: A university case study. In *ISET 2025*, pages 1–5. IEEE.
- Dawson, M. G., Deer, R., e Boguslawski, S. (2025). Cognitive dissonance in programming education: A qualitative exploration of the impact of generative AI on application-directed learning. *Computers in Human Behavior Reports*, 19:100724.
- de Campos, T. P., Damasceno, E. F., e Valentim, N. M. C. (2023). Evaluating a collaborative IS to support systematic reviews and mapping studies (avaliando um SI colaborativo para apoiar revisões sistemáticas da literatura e mapeamentos sistemáticos da literatura). *iSys: Revista Brasileira de Sistemas de Informação*, 16(1):8:1–8:33.
- Denny, P., Leinonen, J., Prather, J., Luxton-Reilly, A., Amarouche, T., Becker, B. A., e Reeves, B. N. (2024). Prompt problems: A new programming exercise for the generative AI era. In *Proceedings of the 55th ACM Technical Symposium on Computer Science Education V. 1*, page 296–302. ACM.
- Dos Santos, O. L. e Cury, D. (2023). Challenging the confirmation bias: Using ChatGPT as a virtual peer for peer instruction in computer programming education. In *FIE 2023*. IEEE.
- Feng, T., Liu, S., e Ghosal, D. (2024). CourseAssist: Pedagogically appropriate AI tutor for computer science education. In *VGCE 2024*, page 310–311. ACM.
- Gandhi, A. e Muldner, K. (2025). "that's not the way I would explain it": A teacher-researcher's autoethnography of learning to program with ChatGPT. In *ICER 2025*. ACM.
- Gong, X., Xu, W., e Qiao, A. (2025). Exploring undergraduates' computational thinking and human-computer interaction patterns in generative progressive prompt-assisted programming learning. *International Journal of Educational Technology in Higher Education*, 22(51).
- Groothuijsen, S., van den Beemt, A., Remmers, J. C., e van Meeuwen, L. W. (2024). AI chatbots in programming education: Students' use in a scientific computing course and consequences for learning. *Computers and Education: Artificial Intelligence*, 7:100290.
- INEP (2013). Censo da educação superior 2011: Resumo técnico. Technical report, Instituto Nacional de Estudos e Pesquisas Educacionais Anísio Teixeira.
- INEP (2024). Resumo técnico do censo da educação superior 2023. Technical report, Instituto Nacional de Estudos e Pesquisas Educacionais Anísio Teixeira.
- Kazemitabaar, M., Chow, J., Ma, C. K. T., Ericson, B. J., Weintrop, D., e Grossman, T. (2023). Studying the effect of AI code generators on supporting novice learners in

- introductory programming. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*, page 1–23. ACM.
- Kitchenham, B. e Charters, S. (2007). Guidelines for performing systematic literature reviews in software engineering. EBSE Technical Report EBSE-2007-01, Keele University and University of Durham.
- Koriat, A. (1997). Monitoring one’s own knowledge during study: A cue-utilization approach to judgments of learning. *Journal of Experimental Psychology: General*, 126(4):349–370.
- Kosmyna, N., Hauptmann, E., Yuan, Y., et al. (2025). Your brain on ChatGPT: Accumulation of cognitive debt when using an AI assistant for essay writing task. arXiv preprint arXiv:2506.08872.
- Kuramitsu, K., Obara, Y., Sato, M., e Obara, M. (2023). KOGI: A seamless integration of ChatGPT into Jupyter environments for programming education. In *Proceedings of the 2023 ACM SIGPLAN International Symposium on SPLASH-E*, page 50–59. ACM.
- Kuttal, S. K., Sedhain, A., e AuBuchon, J. (2021). Designing a gender-inclusive conversational agent for pair programming: An empirical investigation. *Lecture Notes in Computer Science*, 12797 LNAI:59 – 75.
- Landis, J. R. e Koch, G. G. (1977). The measurement of observer agreement for categorical data. *Biometrics*, 33(1):159–174.
- Lewis, P., Perez, E., Piktus, A., et al. (2020). Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Advances in Neural Information Processing Systems*, volume 33, pages 9459–9474.
- Logozar, R., Mikac, M., e Hizak, J. (2023). ChatGPT on the Freshman Test in C/C++ Programming. In *SISY 2023*. IEEE.
- MacNeil, S., Denny, P., Tran, A., Leinonen, J., Bernstein, S., Hellas, A., Sarsa, S., e Kim, J. (2024). Decoding logic errors: A comparative study on bug detection by students and large language models. In *ACE 2024*, pages 11–18. ACM.
- Manorat, P., Tuarob, S., e Pongpaichet, S. (2025). Artificial intelligence in computer programming education: A systematic literature review. *Computers and Education: Artificial Intelligence*, 8:100403.
- Margulieux, L. E., Prather, J., Reeves, B. N., et al. (2024). Self-regulation, self-efficacy, and fear of failure interactions with how novices use LLMs to solve programming problems. In *ITiCSE 2024*, pages 276–282. ACM.
- Masetty, S. V. P. e Vallamulla, S. (2025). Enhancing novice programming education through AI-driven mentorship and project-based learning. In *2025 IEEE Integrated STEM Education Conference (ISEC)*, pages 1–7.
- Medeiros, R. P., Falcão, T. P., e Ramalho, G. L. (2020). Ensino e aprendizagem de introdução à programação no ensino superior brasileiro: Revisão sistemática da literatura. In *Workshop Sobre Educação em Computação (WEI)*, pages 186–190. SBC.
- Moon, J., Yang, R., Cha, S., e Kim, S. B. (2023). ChatGPT vs mentor: Programming language learning assistance system for beginners. In *2023 IEEE 8th International*

Conference On Software Engineering and Computer Systems (ICSECS), pages 106–110.

- Norton, V., Honda, F., Pessoa, M., e Pires, F. (2024). CodeX: auxiliando estudantes com TEA em programação por meio de um ambiente virtual de aprendizagem (AVA) integrado à um LLM. In *Anais Estendidos do XIII Congresso Brasileiro de Informática na Educação*, pages 155–158. SBC.
- Page, M. J., McKenzie, J. E., Bossuyt, P. M., Boutron, I., et al. (2021). The PRISMA 2020 statement: an updated guideline for reporting systematic reviews. *BMJ*, 372:n71.
- Paludo, G. e Montresor, A. (2024). Fostering metacognitive skills in programming: Leveraging AI to reflect on code. In *AIxEDU 2024*, volume 3879 of *CEUR Workshop Proceedings*. CEUR-WS.org.
- Park, S. M., Ho, M., Lin, M. P.-C., e Ryoo, J. (2025). Evaluating the impact of assistive AI tools on learning outcomes and ethical considerations in programming education. In *2025 IEEE Global Engineering Education Conference (EDUCON)*, pages 1–10.
- Pesovski, I., Vorkel, D., e Trajkovik, V. (2024). AI-powered education: Rethinking the way programming is taught using AI tools and reversed Bloom’s taxonomy. In *2024 21st International Conference on Information Technology Based Higher Education and Training (ITHET)*, pages 1–6.
- Petersen, K., Vakkalanka, S., e Kuzniarz, L. (2015). Guidelines for conducting systematic mapping studies in software engineering: An update. *Information and Software Technology*, 64:1–18.
- Phung, T., Padurean, V.-A., Cambroner, J., Gulwani, S., Kohn, T., Majumdar, R., Singla, A., e Soares, G. (2023). Generative AI for programming education: Benchmarking ChatGPT, GPT-4, and human tutors. In *ICER '23 V. 2*, pages 41–42. ACM.
- Portela, C., Palomino, P., Challco, G., Sobrinho, Á., Cordeiro, T., Mello, R., Dermeval, D., Bittencourt, I., e Isotani, S. (2024). AI in education unplugged support equity between rural and urban areas in Brazil. In *ICTD 2024*, pages 143–154.
- Prather, J., Denny, P., Leinonen, J., Becker, B. A., Albluwi, I., Craig, M., et al. (2023). The robots are here: Navigating the generative AI revolution in computing education. In *ITiCSE-WGR 2023*, page 108–159. ACM.
- Prather, J., Reeves, B. N., Leinonen, J., MacNeil, S., Randrianasolo, A. S., Becker, B. A., Kimmel, B., Wright, J., e Briggs, B. (2024). The widening gap: The benefits and harms of generative AI for novice programmers. In *ICER 2024*, pages 469–486. ACM.
- Ramabu, T. e Malebane, T. (2024). Guidelines for effective use of ChatGPT in introductory programming education. In *IST-Africa 2024*, pages 1–8.
- Robins, A., Rountree, J., e Rountree, N. (2003). Learning and teaching programming: A review and discussion. *Computer Science Education*, 13(2):137–172.
- Silva, E. P. d., Caceffo, R., e Azevedo, R. (2023). Uma Análise Curricular de Disciplinas de CS1 das Universidades Públicas Brasileiras. *RBIE*, 31:407–436.
- Silva, T. L. d., Vidotto, K. N. S., Tarouco, L. M. R., e Silva, P. F. d. (2024). Potencialidades do uso de Inteligência Artificial Generativa como apoio ao Ensino de Programação. In *SBIE 2024*, pages 1942–1956.

- Simaremare, M., Pardede, C., Tampubolon, I., Manurung, P., e Simangunsong, D. (2024). Pair programming in programming courses in the era of generative AI: Students' perspective. In *APSEC 2024*, pages 507–511.
- Slimi, Z. (2026). A systematic critical review of generative AI's impact on authorship, pedagogy, and integrity (2023–2025). *Frontiers in Education*, 11:1769680.
- Teodoro, P. V. e Freire, V. (2025). Artificial intelligence and qualitative reasoning in science education: A systematic review of lusophone literature (2000–2024). In *Artificial Intelligence in Education*, Lecture Notes in Computer Science, pages 449–456. Springer.
- Uema, M. A., Souza, T. d. P. C. d., Dermeval, D., Bittencourt, I. I., e Isotani, S. (2025). Designing for meaningful access: Towards a framework for AI in education unplugged. In Cristea, A. I. et al., editors, *Artificial Intelligence in Education*, volume 15880 of *Lecture Notes in Artificial Intelligence (LNAI)*, pages 307–320. Springer Nature Switzerland.
- Vadaparty, A., Geng, F., Smith, D. H., Benario, J. G., Zingaro, D., e Porter, L. (2025). Achievement goals in CS1-LLM. In *ACE 2025*, page 144–153. ACM.
- Valový, M. e Buchalceva, A. (2023). The psychological effects of AI-assisted programming on students and professionals. In *ICSME 2023*, pages 385–390.
- Yilmaz, R. e Karaoglan Yilmaz, F. G. (2023). The effect of generative artificial intelligence (AI)-based tool use on students' computational thinking skills, programming self-efficacy and motivation. *Computers and Education: Artificial Intelligence*, 4:100147.
- Zhang, L., Jiang, Q., Xiong, W., e Zhao, W. (2025). Effects of ChatGPT-based human-computer dialogic interaction programming activities on student engagement. *Journal of Educational Computing Research*, 63(4).
- Zhao, Y., Ling, X., Zhu, H., Zhou, F., e Deng, Z. (2025). Exploration of computer programming teaching reform based on large language models. In *ICCAI 2025*, pages 359–363.
- Zhou, X., Liang, P., Zhang, B., Li, Z., Ahmad, A., Shahin, M., e Waseem, M. (2025). Exploring the problems, their causes and solutions of AI pair programming: A study on GitHub and stack overflow. *Journal of Systems and Software*, 219:112204.
- Zhuang, Y., Ji, Y., Huang, Z., Shang, Y., Li, X., e Su, X. (2025). Exploring the integration of artificial intelligence and programming education: A practice-oriented study on the development of computational thinking based on human-AI co-creation. In *SEAI 2025*, pages 325–329.