

Problemas de Parsons na Educação em Computação: Uma Atualização de Revisão Sistemática da Literatura (2022–2026)

Marcelo Ferreira Ortega^{1,3}, Leandro Silva Galvão de Carvalho²,
David Fernandes², Maristela Terto de Holanda³

¹ IF Goiano – Instituto Federal de Educação, Ciência e Tecnologia Goiano
Rod. GO-154, Km 03, Zona Rural, 76300-000, Ceres – GO – Brasil

²Instituto de Computação – Universidade Federal do Amazonas (UFAM)
Av. Gal. Rodrigo Octávio, 6200, Coroado I – 69080-900 – Manaus – AM – Brasil

³Departamento de Ciência da Computação – Universidade de Brasília (UnB)
Prédio CIC/EST – 70910-900 – Brasília – DF – Brasil

marcelo.ortega@ifgoiano.edu.br, galvao@icomp.ufam.edu.br, mholanda@unb.br

Resumo. Este trabalho apresenta uma atualização da Revisão Sistemática da Literatura (RSL) sobre Problemas de Parsons (PP) conduzida por [Ericson et al. 2022a]. O objetivo é identificar as evidências produzidas entre janeiro de 2022 e abril de 2026 sobre esse tipo de exercício de programação, especialmente sobre suas variantes. Foram utilizados os mesmos mecanismos de busca originais e os critérios de inclusão e exclusão, adicionando-se dois assistentes de IA como novas fontes de busca. Entre os 53 trabalhos encontrados, novas variantes foram identificadas, como micro Parsons e chunky Parsons. Além disso, abordagens adaptativas com apoio da Inteligência Artificial (IA) generativa apresentam potencial para ampliar o engajamento e a personalização da aprendizagem. Apesar dos avanços, ainda há lacunas relacionadas ao uso dos PP em áreas como IA e cibersegurança, bem como estudos voltados à equidade e inclusão nesse tipo de exercício visual.

Abstract. This work presents an update to the Systematic Literature Review (SLR) on Parsons Problems (PP) conducted by [Ericson et al. 2022a]. The objective is to identify the evidence produced between January 2022 and April 2026 on this type of programming exercise, especially its variants. The same original search mechanisms and inclusion and exclusion criteria were used, adding two AI assistants as new search sources. Among the 53 studies found, new variants were identified, such as micro Parsons and chunky Parsons. Furthermore, adaptive approaches supported by generative AI show potential to enhance engagement and personalization of learning. Despite the advances, there are still gaps related to the use of PP in areas such as AI and cybersecurity, as well as studies focused on equity and inclusion in this type of visual exercise.

1. Introdução

Aprender a programar é uma tarefa complexa que envolve diversas habilidades, desde compreender a sintaxe de uma linguagem e projetar soluções algorítmicas até depurar

erros e testar código [Henklain et al. 2025]. Para desenvolver essa competência, atividades de escrita de código são amplamente utilizadas; entretanto, a necessidade de escrever código a partir do zero pode elevar a carga cognitiva de estudantes iniciantes.

Introduzidos por [Parsons and Haden 2006], os Problemas de Parsons (PP) são exercícios de programação nos quais os alunos recebem blocos de código embaralhados e devem organizá-los para formar uma solução funcional [Ericson et al. 2022a]. Diferentemente dos exercícios tradicionais de programação, os PP restringem o espaço de busca da solução, expondo o estudante a padrões de código e sintaxe correta sem exigir memorização exaustiva [Haynes-Magyar 2024]. Estudos fundamentados na Teoria da Carga Cognitiva [Sweller et al. 2011] indicam que essa abordagem reduz o esforço mental exigido no processamento das informações, permitindo maior foco na lógica e no fluxo de controle do programa [Ericson et al. 2022a]. Além disso, há evidências de que resolver PP é mais eficiente do que escrever código manualmente, demandando menos tempo e produzindo ganhos de aprendizagem equivalentes [Bender et al. 2022a, Sheng et al. 2023]. Os PP também favorecem o engajamento dos estudantes [Gutierrez et al. 2024, Szabo et al. 2025].

Em 2022, [Ericson et al. 2022a] conduziram a mais abrangente Revisão Sistemática da Literatura (RSL) sobre PP, consolidando evidências de 141 estudos primários selecionados entre mais de mil artigos. O estudo identificou lacunas relevantes, como a necessidade de pesquisas multinacionais e multi-institucionais, além da investigação de variantes em contextos e da programação introdutória. Contudo, desde a publicação dessa RSL, a pesquisa em Educação em Computação avançou, impulsionada sobretudo pela ascensão dos Modelos de Linguagem de Grande Escala (LLMs), reforçando a necessidade de atualizar o estado da arte [Reeves et al. 2023]. Estudos recentes passaram a investigar o uso de ferramentas como ChatGPT e Codex para gerar ou resolver PP [Hou et al. 2024a], levantando novos desafios para a integridade acadêmica e o design instrucional [Hou et al. 2024b]. Paralelamente, ganharam destaque pesquisas sobre acessibilidade para estudantes neurodiversos e pessoas com deficiência visual [Bender et al. 2022a, Haynes-Magyar 2024].

Este trabalho atualiza a RSL de [Ericson et al. 2022a], mapeando a literatura publicada entre janeiro de 2022 e abril de 2026. Foram mantidos os mesmos mecanismos de busca (ACM Digital Library, IEEE Xplore e Scopus) e os critérios de inclusão e exclusão, incluindo a explicitação de critérios tácitos adotados. As questões de pesquisa que norteiam este trabalho foram adaptadas das originais, visando capturar as mudanças recentes do campo:

- **QP1:** Como os PP têm sido estudados em novos contextos educacionais e com quais características emergentes nos últimos quatro anos?
- **QP2:** Quais novas questões de pesquisa emergiram sobre PP após 2022?
- **QP3:** Quais novas evidências sobre benefícios e limitações dos PP foram produzidas, e qual a qualidade dessas evidências?

2. Trabalhos relacionados

O principal estudo secundário sobre PP é a RSL conduzida por [Ericson et al. 2022a], que consolidou o conhecimento produzido sobre a abordagem até 2022. Essa RSL tornou-se

a principal referência da área e serviu de base metodológica para o presente estudo, tanto na definição da estratégia de busca quanto na estrutura de análise dos resultados.

Após a publicação dessa RSL, foram identificados poucos trabalhos secundários relacionados ao tema. Entre eles, destaca-se [Muldner et al. 2022], que analisou diferentes formas de exemplos trabalhados utilizadas no ensino de programação, incluindo os PP como uma estratégia de suporte pedagógico (*scaffolding*). Também foi identificado o trabalho de [Szabo et al. 2025], que discute a relação entre PP e teorias de aprendizagem frequentemente utilizadas em Educação em Computação.

Além desses estudos, [Ericson et al. 2023] consolidou resultados obtidos por diferentes instituições e países, fornecendo evidências complementares sobre a aplicação da abordagem em múltiplos contextos educacionais. Entretanto, esse trabalho não possui caráter de RSL e concentra-se na síntese de estudos conduzidos por grupos participantes da iniciativa. Apesar dessas contribuições, não foram identificadas RSL que atualizassem os resultados reportados por [Ericson et al. 2022a]. Consequentemente, temas que ganharam relevância após 2022, como Inteligência Artificial (IA) Generativa, *Proof Blocks*, acessibilidade, personalização adaptativa e novas variantes dos PP, ainda não haviam sido sistematizados em uma revisão secundária, lacuna coberta pelo presente trabalho.

3. Metodologia

A metodologia foi atualizada em relação à proposta de [Ericson et al. 2022a] em dois aspectos: adoção do *Software Engineering Guidelines for Reporting Secondary Studies* (SEGRESS) [Kitchenham et al. 2023], uma atualização de [Kitchenham and Charters 2007], e utilização das ferramentas de IA Elicit¹ e Consensus² para ampliar a identificação de estudos e apoiar a comparação de resultados. O processo foi organizado em três fases: planejamento, condução e relato dos resultados.

3.1. Estratégia de Busca e Identificação

A coleta de dados abrangeu o período de janeiro de 2022 a abril de 2026. Foram utilizados os mesmos mecanismos de busca da RSL original (ACM Digital Library, IEEE Xplore e Scopus), mantendo-se os termos “parson problems”, “parson puzzles” e “parson programming”. Como extensão do protocolo original, foram realizadas buscas complementares com as ferramentas de IA Elicit e Consensus, visando ampliar a recuperação de estudos e apoiar a validação do processo de busca. Em cada ferramenta, foi executada uma única busca, cujos resultados foram posteriormente incorporados ao conjunto de registros. Adicionalmente, realizou-se uma busca complementar por meio da técnica de *forward snowballing*, utilizando como trabalhos-semente os estudos de [Parsons and Haden 2006] e [Ericson et al. 2022a]. Nesse procedimento, foram examinados os trabalhos posteriores que citaram esses dois estudos, e os registros potencialmente relevantes foram submetidos aos mesmos critérios de inclusão e exclusão adotados nas demais etapas da RSL.

3.2. Critérios de Inclusão e Exclusão

Foram adotados os mesmos critérios de inclusão e exclusão definidos por [Ericson et al. 2022a], com pequenas adaptações para explicitar critérios que estavam

¹<https://elicit.com/>

²<https://consensus.app/>

implícitos na RSL original. Foram incluídos estudos que: (IC1) apresentassem resultados empíricos sobre o uso de PP ou coletassem dados relacionados à sua utilização; (IC2) descrevessem sistemas ou ferramentas para apresentação ou aplicação de PP; ou (IC3) relatassem o uso de PP no ensino de programação.

Foram excluídos estudos que: não estivessem escritos em inglês ou português (EC1); não correspondessem a artigos completos (EC2); não tivessem sido submetidos à revisão por pares (EC3); fossem teses ou dissertações (EC4); utilizassem PP sem relação direta com os objetivos ou QP do estudo, sem apresentar discussão relevante nos métodos ou resultados (EC5); não fossem relacionados a PP (EC6); não estivessem disponíveis para acesso integral (EC7); ou correspondessem a registros duplicados recuperados durante o processo de busca (EC8).

3.3. Extração e Avaliação de Qualidade

Os estudos foram identificados e selecionados conforme o processo descrito a seguir. Inicialmente, realizaram-se buscas nas bases, complementadas por *forward snowballing* a partir dos trabalhos de [Parsons and Haden 2006] e [Ericson et al. 2022a], com apoio da ferramenta Litmaps³. Após a remoção de duplicatas, o conjunto de fontes foi ampliado por meio das ferramentas Elicit e Consensus. Uma nova etapa de deduplicação resultou em 234 artigos para a triagem inicial, disponíveis em planilha⁴.

Antes da aplicação dos critérios de inclusão e exclusão, os dois primeiros autores realizaram uma etapa de calibração utilizando uma amostra de 19 dos 234 artigos identificados inicialmente. Em seguida, os critérios foram aplicados independentemente ao restante do corpus, o que produziu um coeficiente Kappa de Cohen de 0,87, indicando elevada concordância entre os avaliadores. Os casos divergentes foram posteriormente discutidos, levando à seleção final de 53 artigos (Figura 1).

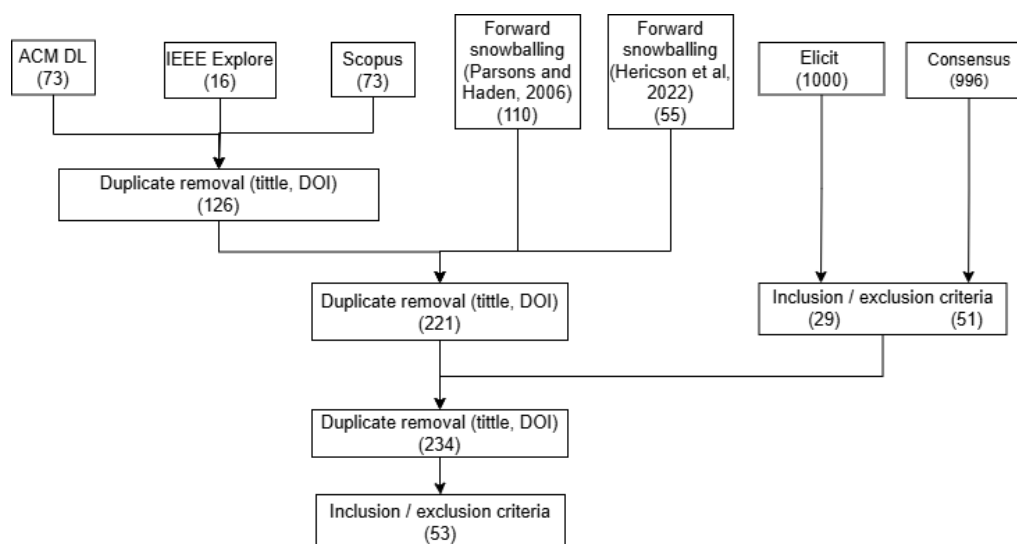


Figura 1. Visão geral do processo de busca completa ilustrando o número de artigos resultantes de cada etapa.

³ <https://www.litmaps.com/>

⁴ https://docs.google.com/spreadsheets/d/1_rTggOPShDp6tEIWZUVqtjhh9pum7QTj/edit?usp=sharing&oid=104376265368637159930&rtfpof=true&sd=true

4. Resultados e Análise

4.1. Visão geral dos trabalhos filtrados

A Tabela 1 apresenta os estudos identificados, distribuídos por ano de publicação. A RSL de [Ericson et al. 2022a] constatou uma tendência de crescimento: de 2006 a 2014, foram publicados apenas 17 trabalhos sobre PP, ao passo que, de 2015 a 2021, esse número subiu para 118, com pico de 27 trabalhos em 2021. Os resultados desta atualização, entretanto, não indicam a continuidade desse crescimento. Embora o número de estudos tenha permanecido expressivo entre 2022 e 2024, observa-se uma redução em 2025. Os trabalhos de 2026 não foram incluídos na tabela, uma vez que o período de busca se encerrou em abril daquele ano e sua inclusão poderia sugerir um declínio artificial, dado que novos estudos ainda podem ser publicados e indexados até o fim do ano. Ainda assim, a continuidade da produção e a diversificação das abordagens investigadas indicam que os PP permanecem um tópico relevante na área de Educação em Computação.

Tabela 1. Distribuição dos artigos encontrados, por ano de publicação.

Ano	Quantidade de artigos	Percentual
2022	14	26%
2023	12	23%
2024	16	30%
2025	8	15%
Total	50	94%

A Tabela 2 apresenta a distribuição de citações dos estudos analisados, considerando os dados obtidos diretamente do Google Acadêmico até abril de 2026, seguindo abordagem semelhante a de [Ericson et al. 2022a]. Os estudos mais citados concentram-se em temas relacionados a variantes adaptativas dos PP, *scaffolding* e integração com ambientes utilizados no ensino introdutório de programação (CS1), o que sugere a importância dessas frentes. Observou-se também crescimento nas citações de estudos sobre IA Generativa e *Proof Blocks*.

Tabela 2. Artigos mais citados até abril de 2026, segundo o Google Acadêmico.

Artigo	Citações
[Reeves et al. 2023]	95
[Hou et al. 2024c]	64
[Hou et al. 2024a]	42
[Muldner et al. 2022]	52
[Hou et al. 2022]	39
[Smith IV and Zilles 2023]	16
[Fromont et al. 2023]	14

Assim como observado por [Ericson et al. 2022a], a Tabela 3 mostra que os Estados Unidos permaneceram como principal origem das publicações, com 45% do total, seguidos por países como Nova Zelândia, Finlândia e Austrália. Entretanto, a literatura recente também evidencia a ampliação gradual da participação internacional, com estudos conduzidos em contextos latino-americanos, asiáticos e europeus.

Tabela 3. Distribuição dos estudos por país de afiliação principal

País	Quantidade de artigos	Percentual
Estados Unidos	24	45%
Nova Zelândia	8	15%
Finlândia	5	9%
Austrália	4	8%
Canadá	3	6%
Brasil	2	4%
Reino Unido	2	4%
Outros	5	10%
Total	53	100%

No contexto latino-americano, foram identificados ainda poucos estudos em comparação com a produção internacional. Destacam-se o sistema *ParsonFlow* desenvolvido no Brasil [Santos et al. 2025] e o estudo conduzido na Universidade do Chile por [Sanhueza et al. 2024]. Apesar dessas iniciativas, a produção regional permanece limitada e concentrada em poucos grupos de pesquisa.

4.2. QP1: Como os PP têm sido estudados em novos contextos educacionais e com quais características emergentes nos últimos quatro anos?

Para responder à QP1, primeiramente são discutidas as principais linguagens e ambientes utilizados. Em seguida, são descritas as variantes recentes dos PP, bem como os diferentes contextos educacionais e tecnológicos nos quais a abordagem vem sendo aplicada.

A Tabela 4 apresenta a distribuição das linguagens mais utilizadas nos estudos analisados nesta atualização da RSL. Em comparação com a RSL de [Ericson et al. 2022a], observa-se que Python permaneceu como a linguagem mais utilizada em pesquisas envolvendo PP. Entretanto, essa comparação deve ser interpretada com cautela, pois os 29,17% reportados por [Ericson et al. 2022a] correspondem a estudos publicados entre 2006 e 2022, enquanto os 40% identificados nesta RSL referem-se apenas ao período de 2022 a 2026. Ainda assim, os resultados sugerem a manutenção e até mesmo o fortalecimento da predominância de Python nas pesquisas recentes. Uma possível explicação para esse cenário é que a linguagem possui sintaxe relativamente simples e concisa, permitindo expressar algoritmos básicos com um número reduzido de linhas de código. No contexto dos PP, isso reduz o número de blocos a serem manipulados pelos estudantes e mantém o espaço de solução cognitivamente mais simples.

Outra diferença significativa em relação à RSL de 2022 refere-se ao aumento do uso de ambientes visuais e abordagens baseadas em blocos. Enquanto Scratch representava apenas 1,79% dos estudos identificados por [Ericson et al. 2022a], nesta atualização o ambiente corresponde a 17% dos artigos analisados. Além disso, estudos classificados como “Lógica / Proof Blocks” também alcançaram 17%, evidenciando uma expansão recente dos PP para contextos de lógica formal, construção de provas e raciocínio computacional. Entretanto, essa categoria concentra-se em um único grupo de pesquisa: dos nove artigos identificados sobre lógica, oito foram desenvolvidos por Poulsen et al. descrevendo o *Proof Blocks*. Assim, embora promissora, essa linha de investigação permanece associada a um grupo específico e demanda maior validação por pesquisadores indepen-

centes. Por outro lado, linguagens tradicionalmente utilizadas em CS1, como Java e C, apresentaram redução em comparação com a RSL anterior, caindo de 19,05% e 17,86% para 8% e 2%, respectivamente.

Tabela 4. Linguagens de programação utilizadas

Linguagem	Quantidade	Percentual
Python	21	40%
Scratch	9	17%
Lógica / Proof Blocks	9	17%
Java	4	8%
Múltiplas linguagens	3	6%
SQL	2	4%
Haskell / Programação Funcional	1	2%
C	1	2%
JavaScript	1	2%
Pseudocódigo	1	2%
Não focado em linguagem específica	1	2%
Total	53	100 %

O período analisado evidenciou uma diversificação das variantes de PP, refletindo a busca por abordagens adequadas a diferentes objetivos pedagógicos. Destacam-se os *Micro PP*, aplicados ao ensino de expressões regulares e SQL [Wu et al. 2023, Wu and Ericson 2024]; os PP com Lacunas (*Faded Parsons Problems*), que combinam reordenação de linhas e preenchimento parcial de código [Lojo and Fox 2022, Caraco et al. 2024, Ericson et al. 2022a]; os PP *Chunky*, organizados em blocos maiores para apoiar a resolução de problemas [Shabrina et al. 2023]; e os PP com Blocos Aninhados, introduzidos no ensino de programação funcional [Bell et al. 2026]. Também foram propostas ferramentas de autoria automática, como o FPPgen [Caraco et al. 2024], e mecanismos adaptativos capazes de ajustar dinamicamente o nível de dificuldade conforme o desempenho dos estudantes [Sheng et al. 2023, Ericson et al. 2022b].

4.3. QP2: Quais novas questões de pesquisa emergiram sobre PP após 2022?

Dentre as novas questões de pesquisa, destacam-se a integração entre PP e tecnologias baseadas em IA e as investigações sobre acessibilidade e adaptação a diferentes modalidades de ensino, apresentadas a seguir.

Parte dos estudos identificados no período analisado investigou a relação entre PP e tecnologias baseadas em IA, especialmente LLMs. Entre essas iniciativas, destaca-se o PuzzleMakerPy, que utiliza o GPT-3.5 da OpenAI para gerar automaticamente PP contextualizados e personalizados para estudantes de programação introdutória. Os resultados reportados indicam que a contextualização temática dos exercícios contribuiu positivamente para o engajamento dos estudantes [Gutierrez et al. 2024]. Outros trabalhos concentraram-se na avaliação da capacidade de LLMs em resolver PP. [Reeves et al. 2023] analisaram o desempenho do OpenAI Codex na reorganização correta de blocos de código e observaram que o modelo resolveu aproximadamente metade dos problemas avaliados, desempenho consideravelmente inferior ao obtido em tarefas tradicionais de geração livre de código. De maneira semelhante, [Poulsen et al. 2024] investigaram o desempenho do GPT-3.5 e do GPT-4 em problemas de blocos de prova

matemática. Os resultados mostraram limitações significativas do GPT-3.5, que alcançou apenas 11,4% de acerto, enquanto o GPT-4 obteve 64,8%, embora ainda apresentasse dificuldades relacionadas à ordenação lógica das linhas de prova.

Apesar do crescente interesse na integração entre LLMs e PP, não foram identificados estudos que utilizassem explicitamente essa abordagem para o ensino direto de conceitos de IA. Dessa forma, a relação observada na literatura recente concentra-se predominantemente no uso da IA como mecanismo de apoio à geração, adaptação ou resolução automática de problemas, e não como conteúdo instrucional ensinado por meio desse paradigma.

Os estudos recentes também evidenciaram esforços para ampliar o uso de PP em contextos além dos ambientes tradicionais de sala de aula, com destaque para iniciativas relacionadas à acessibilidade e à aprendizagem em larga escala. [Kluthe et al. 2026] propuseram uma nova tecnologia de blocos para plataformas móveis em conformidade com as regulamentações de acessibilidade estabelecidas pelo Departamento de Justiça dos Estados Unidos em 2024. Complementarmente, [Haynes-Magyar 2024] conduziram um estudo exploratório com programadores neurodiversos e identificaram recomendações voltadas à melhoria da acessibilidade cognitiva dos PP, incluindo a redução de exigências relacionadas à aritmética mental em determinados perfis de estudantes. Além disso, a literatura recente indica interesse crescente na aplicação de PP em ambientes digitais de larga escala. [Wu et al. 2023] implementaram Micro PP em um MOOC e observaram taxas de evasão significativamente menores em comparação às abordagens tradicionais. De forma semelhante, o sistema *ParsonFlow* foi desenvolvido com suporte multiplataforma, permitindo sua utilização em computadores, tablets e smartphones [Santos et al. 2025].

No que se refere às discussões sobre equidade, diversidade e inclusão, a temática ainda aparece de maneira limitada no corpus analisado. [Bender et al. 2022b] destacam-se por abordar explicitamente a promoção de uma educação em pensamento computacional mais equitativa e inclusiva. Entretanto, poucos estudos investigaram de forma aprofundada fatores relacionados a gênero, raça ou desigualdades socioeconômicas. Embora [Sanhueza et al. 2024] tenham a distribuição de gênero em sua amostra de estudantes, não foram realizadas análises específicas sobre possíveis diferenças associadas a esse fator.

4.4. QP3 : Quais novas evidências sobre benefícios e limitações dos PP foram produzidas, e qual a qualidade dessas evidências?

Para responder à QP3, esta seção analisa as evidências empíricas identificadas nos estudos publicados entre 2022 e 2026 sobre os benefícios, limitações e impactos pedagógicos dos PP. Diferentemente da RSL conduzida por [Ericson et al. 2022a], que concentrou sua análise principalmente em aspectos relacionados à redução da carga cognitiva, aumento do engajamento e apoio à aprendizagem introdutória de programação, os estudos recentes passaram a investigar novas dimensões, incluindo pensamento computacional, habilidades metacognitivas, acessibilidade, variantes adaptativas e integração com IA Generativa.

4.4.1. Pensamento Computacional e Habilidades Cognitivas

Os PP têm sido investigados como instrumentos para o desenvolvimento do pensamento computacional, especialmente em contextos de educação básica e aprendizagem intro-

dutória de programação. Enquanto a RSL de [Ericson et al. 2022a] enfatizava predominantemente aspectos relacionados à aprendizagem sintática e reorganização de código, os estudos recentes indicam o uso para habilidades cognitivas mais amplas, incluindo decomposição de problemas, planejamento, raciocínio lógico e metacognição.

[Bender et al. 2022a] analisaram o uso de PP baseados em blocos no ambiente Scratch e observaram que versões contendo feedback imediato e ausência de distratores promoveram maior eficiência na aprendizagem de conceitos relacionados ao pensamento computacional. Resultados semelhantes foram encontrados por [Bender et al. 2022b], que investigaram a integração de PP ao Scratch e identificaram ganhos significativos no desenvolvimento de habilidades computacionais introdutórias.

Em uma perspectiva mais voltada ao raciocínio estruturado, [Shabrina et al. 2023] propuseram os *Chunky Parsons Problems* como estratégia para apoiar habilidades de decomposição e recomposição de problemas. Os autores observaram que estudantes expostos a explicações associadas aos fragmentos de código desenvolveram soluções mais estruturadas e eficientes. Esses resultados sugerem que variantes dos PP podem atuar como mecanismos de *scaffolding* cognitivo mais sofisticados do que aqueles identificados na RSL de [Ericson et al. 2022a].

Além das habilidades operacionais de programação, estudos recentes passaram a explorar aspectos metacognitivos associados aos PP. [Reeves et al. 2023] verificaram que esse tipo de atividade contribuiu para o aprimoramento do conhecimento metacognitivo relacionado ao planejamento de tarefas. Em uma abordagem mais teórica, [Szabo et al. 2025] estabeleceram conexões explícitas entre PP e teorias da aprendizagem em Computação, incluindo máquinas nocionais e taxonomias de engajamento.

Em conjunto, esses achados sugerem os PP vêm sendo compreendidos não apenas como exercícios de reorganização de código, mas também como instrumentos capazes de apoiar o desenvolvimento de habilidades cognitivas mais amplas relacionadas à resolução de problemas e ao raciocínio computacional.

4.4.2. Evidências sobre Variantes dos PP

Ao sistematizar as características dos PP, [Ericson et al. 2022a] afirmam que um PP típico possui um objetivo claro e espaço de soluções limitado, que facilita a construção em direção a esse objetivo. Por sua vez, as variantes, seja quais forem, podem: (1) limitar o espaço de soluções, a fim de prover suporte pedagógico adicional, ou (2) ampliar o espaço de soluções, expondo o estudante a dificuldades controladas.

Partindo dessa estrutura, [Ericson et al. 2022a] identificaram as seguintes variantes: 1) PP horizontais, nos quais os blocos são fornecidos na ordem correta e uma solução deve ser construída indentando-se os blocos [Park et al. 2016]; 2) PP 2D, nos quais os blocos devem ser posicionados na ordem correta na vertical e indentados na horizontal [Ihantola and Karavirta 2011]; 3) PP adaptativos, cujo nível de dificuldade se ajusta ao desempenho de cada estudante [Ericson and Haynes-Magyar 2022]; 4) PP em nível de projeto, em que os blocos representam etapas abstratas em nível de projeto, em vez de código-fonte [Garcia et al. 2018]; 5) PP com distratores, que acrescentam blocos incorretos ou desnecessários [Smith IV et al. 2023]; e 6) PP com lacunas (*faded*), que fornecem

alguns blocos incompletos nos quais o estudante precisa digitar o código ausente dentro do bloco para completar a solução [Weinman et al. 2021]. Neste trabalho, mais duas variantes foram identificadas: 7) *Chunky Parsons*, que agrupam trechos maiores de código em um só bloco [Shabrina et al. 2023]; e 8) *Micro PP*, centrados em uma única linha de código [Wu et al. 2023].

Desde a RSL de 2022, novas evidências ampliaram o conhecimento sobre o efeito das variantes dos PP sobre a aprendizagem. Estudos envolvendo distratores mostraram que seu impacto depende do desenho instrucional, podendo aumentar a carga cognitiva dos estudantes [Smith IV et al. 2023]. Nos PP com lacunas, a combinação entre ordenação de blocos e preenchimento gradual de trechos de código favoreceu a transição para a escrita de programas [Fromont et al. 2023, Caraco et al. 2024]. Entre as novas variantes, os *Chunky Parsons*, que se mostraram úteis para exercitar a decomposição e a recomposição de problemas [Shabrina et al. 2023], e os *Micro PP* apresentaram resultados promissores no ensino de expressões regulares e SQL [Wu et al. 2023, Wu and Ericson 2024]. Esses resultados reforçam que a eficácia das diferentes variantes depende do contexto de aplicação e das escolhas de design instrucional, corroborando a necessidade de estudos em larga escala destacada por [Ericson et al. 2022a].

4.5. Evidências sobre Inteligência Artificial Generativa

A integração entre PP e IA Generativa constitui uma das principais diferenças em relação à RSL de [Ericson et al. 2022a]. Os estudos recentes passaram a investigar tanto o uso de LLMs para geração automática de problemas quanto sua capacidade de resolver atividades tradicionalmente propostas a estudantes.

As evidências disponíveis indicam que modelos como GPT-4 apresentam desempenho superior ao GPT-3.5 em tarefas de reorganização lógica de blocos, embora ainda encontrem dificuldades em problemas que exigem raciocínio mais elaborado, como os *Proof Blocks* [Poulsen et al. 2024]. Além disso, sistemas de personalização baseados em LLMs mostraram potencial para aumentar o engajamento dos estudantes por meio da contextualização automática dos exercícios [Du et al. 2020, Hou et al. 2024c].

Apesar desses resultados promissores, as evidências ainda são predominantemente exploratórias e concentradas em protótipos ou estudos de curta duração. Assim, diferentemente dos benefícios pedagógicos dos PP já consolidados, ainda não existem evidências robustas sobre os impactos educacionais de longo prazo da integração entre LLMs e PP.

4.5.1. Acessibilidade, Inclusão e Escalabilidade

Nesse sentido, [Haynes-Magyar 2024] conduziram um estudo exploratório com programadores neurodiversos e identificaram recomendações voltadas à melhoria da acessibilidade cognitiva dos PP. Entre os resultados, sobressaíram dificuldades relacionadas à aritmética mental e à organização visual dos blocos.

Também foram encontradas evidências sobre o potencial dos PP em ambientes digitais escaláveis. [Wu et al. 2023] implementaram Micro PP em um MOOC com 3.752 participantes e observaram menores taxas de evasão em comparação às abordagens tradicionais. Além disso, [Santos et al. 2025] desenvolveram o sistema *ParsonFlow* com suporte multiplataforma para computadores, *tablets* e *smartphones*.

Apesar desses avanços, os estudos relacionados à equidade, diversidade e inclusão ainda permanecem limitados. Embora [Bender et al. 2022b] tenham discutido explicitamente aspectos relacionados à inclusão no ensino de pensamento computacional, poucos trabalhos investigaram fatores relacionados a gênero, raça ou desigualdade socioeconômica. Assim, as evidências atuais ainda são insuficientes para compreender plenamente como diferentes perfis de estudantes interagem com PP em contextos diversos.

4.5.2. Limitações e Qualidade das Evidências

A produção sobre PP continuou crescendo desde a RSL de [Ericson et al. 2022a], mas persistem limitações metodológicas importantes. Muitos estudos se concentram em contextos específicos de CS1, com amostras relativamente pequenas e duração limitada.

Em comparação à RSL original, observou-se crescimento no número de estudos experimentais, avaliações multi-institucionais e análises quantitativas em larga escala. Destaca-se, em particular, o trabalho de [Ericson et al. 2023], que consolidou resultados multi-institucionais envolvendo diferentes países e contextos educacionais. Entretanto, ainda são escassos estudos longitudinais capazes de avaliar impactos duradouros dos PP no aprendizado de programação.

Além disso, os resultados encontrados na literatura nem sempre são uniformemente positivos. Estudos envolvendo distratores identificaram que determinados desenhos instrucionais podem aumentar significativamente a carga cognitiva dos estudantes [Smith IV et al. 2023]. Da mesma forma, pesquisas relacionadas à acessibilidade indicaram que certos formatos de PP podem representar barreiras para estudantes neurodiversos ou com deficiência visual [Haynes-Magyar 2024, Kluthe et al. 2026]. Outro aspecto importante refere-se às evidências envolvendo IA Generativa. Embora os estudos indiquem potencial para personalização e automação, as avaliações ainda permanecem preliminares e frequentemente limitadas a cenários experimentais controlados. Consequentemente, ainda não existem evidências robustas sobre os impactos pedagógicos de longo prazo da integração entre LLMs e PP.

De maneira geral, as evidências produzidas após 2022 reforçam os benefícios pedagógicos já identificados por [Ericson et al. 2022a], especialmente no suporte à aprendizagem introdutória de programação, redução da carga cognitiva e aumento do engajamento. Entretanto, a literatura recente amplia significativamente o escopo investigativo ao incorporar questões relacionadas à acessibilidade, IA Generativa, personalização, pensamento computacional e *scaffolding* adaptativo. Apesar desses avanços, permanecem limitadas as evidências longitudinais e multinacionais capazes de avaliar impactos duradouros dos PP em diferentes perfis de estudantes e contextos educacionais.

4.6. PP no Contexto Educacional Brasileiro

Para encontrar trabalhos sobre PP produzidos no Brasil, replicou-se a *string* de busca na biblioteca SBC-OpenLib (SOL)⁵, que retornou um único resultado: o trabalho de [Santos et al. 2025], já encontrado pelo *forward snowballing* a partir do artigo seminal de [Parsons and Haden 2006]. Em seguida, explorou-se a literatura cinzenta por meio

⁵ <https://sol.sbc.org.br/index.php/indice>

do Google Acadêmico, no qual foram localizados apenas três trabalhos adicionais, todos monografias de graduação: [Beltrame 2023] e [Izidoro 2024] desenvolveram e aprimoraram uma plataforma web para resolução de problemas de Parsons⁶, enquanto [Reis 2023] mapeou a literatura para catalogar variações de PP. Em suma, os resultados indicam que a produção acadêmica brasileira ainda é exígua, o que aponta oportunidades para investigações sensíveis às características específicas do contexto educacional brasileiro.

5. Limitações deste estudo

Esta RSL apresenta algumas limitações. Embora tenha seguido a estratégia de busca da RSL de [Ericson et al. 2022a], a atualização ficou restrita aos estudos publicados entre janeiro de 2022 e abril de 2026 e indexados nas bases ACM Digital Library, IEEE Xplore e Scopus. Assim, trabalhos relevantes disponíveis em outras bases ou na literatura cinzenta podem não ter sido recuperados. Embora as ferramentas Elicit e Consensus tenham sido utilizadas para ampliar a cobertura da busca, elas também estão sujeitas a limitações inerentes aos mecanismos automatizados de recuperação de informação. Outra limitação refere-se ao idioma: estudos relevantes não indexados ou publicados em línguas distintas do inglês e do português foram excluídos. Por fim, a subjetividade inerente ao julgamento humano nas etapas de triagem e extração de dados foi mitigada por duas rodadas de discussão conjunta, voltadas à calibração de critérios e à checagem de discrepâncias, e pela avaliação independente de dois revisores, com cálculo do índice de concordância.

6. Conclusão

Este trabalho atualizou a RSL de [Ericson et al. 2022a], analisando a evolução das pesquisas sobre Problemas de Parsons entre janeiro de 2022 e abril de 2026. Os novos estudos encontrados mostram a ampliação do uso dos PP para domínios além de CS1, incluindo lógica proposicional, SQL, expressões regulares, programação funcional, engenharia de software e provas matemáticas, bem como o surgimento de novas variantes, como Micro PP, PP *Chunky* e PP com Lacunas.

Outro resultado relevante foi a crescente integração da IA Generativa. Estudos recentes passaram a investigar o uso de LLMs tanto para geração quanto para resolução automática de PP, ampliando possibilidades de personalização da aprendizagem e automação educacional, mas também introduzindo desafios relacionados à integridade acadêmica, avaliação da aprendizagem e desenho instrucional. Os resultados também reforçam evidências sobre o potencial da abordagem para apoiar o desenvolvimento do pensamento computacional e da aprendizagem de programação, além de indicar crescimento de pesquisas em acessibilidade e adaptação para diferentes perfis de estudantes.

Apesar dos avanços observados, permanecem lacunas importantes, especialmente quanto à realização de estudos longitudinais e multicêntricos, bem como investigações sobre acessibilidade, inclusão, diversidade e equidade. Além disso, novas variantes dos PP e abordagens emergentes, como o uso de IA Generativa para geração, personalização e avaliação de atividades, ainda demandam validação em diferentes contextos educacionais. Em síntese, esta RSL confirma que os PP seguem como uma abordagem relevante para o ensino de computação e oferece um panorama atualizado para subsidiar futuras pesquisas e aplicações desse tipo de exercício.

⁶ <https://ramonxxii.github.io/TCC/>

Agradecimentos

O presente trabalho foi realizado com os seguintes apoios: 1) Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (AUXPE-CAPES-PROEX), Código de Financiamento 001; 2) Fundação de Amparo à Pesquisa do Estado do Amazonas – FAPEAM, por meio dos projetos PDPG-CAPES e POSGRAD 2025-2026 e POSGRAD 2026-2027, e 3) Projeto Ciência de Dados Aplicado à Saúde e Educação na Amazônia (Convênio para Pesquisa, Desenvolvimento e Inovação PD&I Nº 27/2025/UFAM-FAEPI).

Uso de Inteligência Artificial

As plataformas Elicit e Consensus foram empregadas como mecanismos complementares de busca bibliográfica, auxiliando na identificação de estudos potencialmente relevantes. Adicionalmente, a ferramenta NotebookLM foi utilizada experimentalmente na aplicação preliminar dos critérios de inclusão e exclusão, em comparação ao processo humano. Entretanto, todas as decisões de seleção, análise e interpretação dos estudos foram realizadas e validadas pelos autores. Ferramentas de IA generativa também foram utilizadas para apoio à revisão textual e organização preliminar da escrita, sem substituição da análise crítica, interpretação científica ou elaboração intelectual conduzida pelos pesquisadores.

Referências

- Bell, J., Deschamps, A., Kapoor, E., Karki, S., Kim, J., González, N. M., Pitchford, W., Sturua, E., Wade, C., and Rebelsky, S. A. (2026). *An Innovative Approach to Parsons Problems for Teaching and Learning Functional Programming*, pages 114–120. Association for Computing Machinery.
- Beltrame, J. V. F. (2023). Uma plataforma web com parsons puzzles para apoio ao aprendizado de programação. Trabalho de Conclusão de Curso (Bacharelado em Ciência da Computação), Universidade Federal de Santa Maria (UFSM), Santa Maria, RS.
- Bender, J., Dziena, A., and Kaiser, G. (2022a). Learning computational thinking efficiently with block-based Parsons puzzles. In Iyer, S., Shih, J.-L., Chen, W., and Md Khambhari, M. N., editors, *30th International Conference on Computers in Education Conference, ICCE 2022 – Proceedings*, volume 1, pages 439–448. Asia-Pacific Society for Computers in Education.
- Bender, J., Zhao, B., Dziena, A., and Kaiser, G. (2022b). Integrating Parsons puzzles within Scratch enables efficient computational thinking learning. *Research and Practice in Technology Enhanced Learning*.
- Caraco, S., Lojo, N., Verdicchio, M., and Fox, A. (2024). Generating multi-part autogradable faded Parsons problems from code-writing exercises. In *SIGCSE 2024 - Proceedings of the 55th ACM Technical Symposium on Computer Science Education*, volume 1, pages 179 – 185. Association for Computing Machinery, Inc.
- Du, Y., Luxton-Reilly, A., and Denny, P. (2020). A review of research on Parsons problems. In Luxton-Reilly, A. and Szabo, C., editors, *ACE 2020, Proceedings of the Twenty-Second Australasian Computing Education Conference, Melbourne, VIC, Australia, February 4-6, 2020*, pages 195–202. ACM.

- Ericson, B. and Haynes-Magyar, C. (2022). Adaptive Parsons problems as active learning activities during lecture. In *Proceedings of the 27th ACM Conference on on Innovation and Technology in Computer Science Education Vol. 1*, pages 290–296. Association for Computing Machinery.
- Ericson, B. J., Denny, P., Prather, J., Duran, R., Hellas, A., Leinonen, J., Miller, C. S., Morrison, B. B., Pearce, J. L., and Rodger, S. H. (2022a). Parsons problems and beyond: Systematic literature review and empirical study designs. In *Proceedings of the 2022 Working Group Reports on Innovation and Technology in Computer Science Education*, pages 191–234. Association for Computing Machinery.
- Ericson, B. J., Maeda, H., and Dhillon, P. S. (2022b). Detecting struggling students from interactive ebook data: A case study using CSAwesome. In *Proceedings of the 53rd ACM Technical Symposium on Computer Science Education - Volume 1*, pages 418–424. Association for Computing Machinery.
- Ericson, B. J., Pearce, J. L., Rodger, S. H., Csizmadia, A., Garcia, R., Gutierrez, F. J., Liaskos, K., Padiyath, A., Scott, M. J., Smith IV, D. H., Warriem, J. M., and Bernuy, A. Z. (2023). Multi-institutional multi-national studies of Parsons problems. In *Proceedings of the 2023 Working Group Reports on Innovation and Technology in Computer Science Education*, pages 57–107. Association for Computing Machinery.
- Fromont, F., Jayamanne, H., and Denny, P. (2023). Exploring the difficulty of faded Parsons problems for programming education. In *ACM International Conference Proceeding Series*, pages 113–122. Association for Computing Machinery.
- Garcia, R., Falkner, K., and Vivian, R. (2018). Scaffolding the design process using Parsons problems. In *Proceedings of the 18th Koli Calling International Conference on Computing Education Research*, Koli Calling '18, New York, NY, USA. Association for Computing Machinery.
- Gutierrez, A. d. C., Denny, P., and Luxton-Reilly, A. (2024). Automating personalized Parsons problems with customized contexts and concepts. In *Proceedings of the 2024 on Innovation and Technology in Computer Science Education V. 1*, pages 688–694. Association for Computing Machinery.
- Haynes-Magyar, C. (2024). Neurodiverse programmers and the accessibility of Parsons problems: An exploratory multiple-case study. In *SIGCSE 2024 – Proceedings of the 55th ACM Technical Symposium on Computer Science Education*, volume 1, pages 491 – 497. Association for Computing Machinery, Inc.
- Henklain, M. H. O., Carvalho, L. S. G., and Feitosa, E. L. (2025). Dificuldades no aprendizado de programação: Um exame baseado em mapeamento sistemático da literatura e interpretação analítico-comportamental. *Revista Brasileira de Informática na Educação*, 33:1484–1521.
- Hou, I., Man, O., Mettelle, S., Gutierrez, S., Angelikas, K., and MacNeil, S. (2024a). More robots are coming: Large multimodal models (ChatGPT) can solve visually diverse images of Parsons problems. In *ACM International Conference Proceeding Series*, pages 29 – 38. Association for Computing Machinery.
- Hou, X., Ericson, B. J., and Wang, X. (2022). Using adaptive Parsons problems to scaffold write-code problems. In *Proceedings of the 2022 ACM Conference on International*

- Computing Education Research - Volume 1*, pages 15–26. Association for Computing Machinery.
- Hou, X., Wu, Z., Wang, X., and Ericson, B. (2024b). CodeTailor: Personalized Parsons puzzles are preferred over ai-generated solutions to support learning. *arXiv.org*.
- Hou, X., Wu, Z., Wang, X., and Ericson, B. J. (2024c). CodeTailor: LLM-powered personalized Parsons puzzles for engaging support while learning programming. In *L@S 2024 - Proceedings of the 11th ACM Conference on Learning @ Scale*, pages 51 – 62. Association for Computing Machinery, Inc.
- Ihantola, P. and Karavirta, V. (2011). Two-dimensional Parson’s puzzles: The concept, tools, and first observations. *Journal of Information Technology Education. Innovations in Practice*, 10:119.
- Izidoro, R. G. (2024). Extensão de uma plataforma web para criação de parsons puzzles de apoio ao ensino de programação para iniciantes. Trabalho de Conclusão de Curso (Bacharelado em Ciência da Computação), Universidade Federal de Santa Maria (UFSM), Santa Maria, RS.
- Kitchenham, B. and Charters, S. (2007). Guidelines for performing systematic literature reviews in software engineering. Technical Report EBSE Technical Report EBSE-2007-01, Engineering and Physical Sciences Research Council, Keele University and Durham University.
- Kitchenham, B., Madeyski, L., and Budgen, D. (2023). SEGRESS: Software Engineering Guidelines for REporting Secondary Studies. *IEEE Transactions on Software Engineering*, 49(3):1273–1298.
- Kluthe, T., Moore, W. J. R., Hofman, A., Gorlewicz, J. L., Contreras, G., Roy, N., Smith, D., Giudice, N. A., Allee, W., Williams, H., Blaser, B., and Stefk, A. (2026). Toward accessible Parsons problems on mobile platforms. In *SIGCSE TS 2026 - Proceedings of the 57th ACM Technical Symposium on Computer Science Education V.1*, pages 589 – 595. Association for Computing Machinery, Inc.
- Lojo, N. and Fox, A. (2022). Teaching test-writing as a variably-scaffolded programming pattern. In *Proceedings of the 27th ACM Conference on on Innovation and Technology in Computer Science Education Vol. 1*, pages 498–504. Association for Computing Machinery.
- Muldner, K., Jennings, J., and Chiarelli, V. (2022). A review of worked examples in programming activities. *ACM Trans. Comput. Educ.*, 23.
- Park, T. H., Kim, M. C., Chhabra, S., Lee, B., and Forte, A. (2016). Reading hierarchies in code: Assessment of a basic computational skill. In *Proceedings of the 2016 ACM Conference on innovation and technology in computer science education*, pages 302–307.
- Parsons, D. and Haden, P. (2006). Parson’s programming puzzles: a fun and effective learning tool for first programming courses. In *Proceedings of the 8th Australasian Conference on Computing Education - Volume 52*, ACE ’06, page 157–163, AUS. Australian Computer Society, Inc.

- Poulsen, S., Sarsa, S., Prather, J., Leinonen, J., Becker, B. A., Hellas, A., Denny, P., and Reeves, B. N. (2024). Solving proof block problems using large language models. In *Proceedings of the 55th ACM Technical Symposium on Computer Science Education V. 1*, pages 1063–1069. Association for Computing Machinery.
- Reeves, B., Sarsa, S., Prather, J., Denny, P., Becker, B. A., Hellas, A., Kimmel, B., Powell, G., and Leinonen, J. (2023). Evaluating the performance of code generation models for solving Parsons problems with small prompt variations. In *Proceedings of the 2023 Conference on Innovation and Technology in Computer Science Education V. 1*, pages 299–305. Association for Computing Machinery.
- Reis, A. C. L. (2023). Uso de problemas de parson no ensino de programação introdutória. Trabalho de Conclusão de Curso (Bacharelado em Engenharia da Computação), Universidade Federal do Amazonas (UFAM), Manaus, AM.
- Sanhueza, F., Gutierrez, F. J., and Vasquez, A. (2024). Exploring the acceptance and effectiveness of Parsons problems on scaffolding CS1 retakers. In *Proceedings of the 2024 on Innovation and Technology in Computer Science Education V. 1*, pages 681–687. Association for Computing Machinery.
- Santos, A. M. S., Souza, C., and Araujo, L. G. J. (2025). Parsonflow: Um jogo baseado em problemas de parson para auxiliar o processo de ensino-aprendizagem em programação. *Brazilian Symposium on Games and Digital Entertainment*.
- Shabrina, P., Mostafavi, B., Tithi, S. D., Chi, M., and Barnes, T. (2023). Learning problem decomposition-recomposition with data-driven chunky Parsons problems within an intelligent logic tutor. *Proceedings of the International Conference on Educational Data Mining*, pages 40 – 54.
- Sheng, Y., Li, B., Wu, Z., Zhong, P., and Duan, G. (2023). A C language learning platform based on Parsons problems. *Communications in Computer and Information Science*, 1812 CCIS:541 – 552.
- Smith IV, D. H., Fowler, M., and Zilles, C. (2023). Investigating the role and impact of distractors on Parsons problems in CS1 assessments. In *Proceedings of the 2023 Conference on Innovation and Technology in Computer Science Education V. 1*, pages 417–423. Association for Computing Machinery.
- Smith IV, D. H. and Zilles, C. (2023). Discovering, autogenerating, and evaluating distractors for Python Parsons problems in CS1. In *Proceedings of the 54th ACM Technical Symposium on Computer Science Education V. 1*, pages 924–930. Association for Computing Machinery.
- Sweller, J., Ayres, P., and Kalyuga, S. (2011). *Cognitive Load Theory*. Explorations in the Learning Sciences, Instructional Systems and Performance Technologies. Springer New York.
- Szabo, C., Sheard, J., Malmi, L., and Kinnunen, P. (2025). Parsons problems and computing education learning theories. In J., L. and R., D., editors, *Proceedings of 25th International Conference on Computing Education Research, Koli Calling 2025*. Association for Computing Machinery, Inc.
- Weinman, N., Fox, A., and Hearst, M. A. (2021). Improving instruction of programming patterns with faded Parsons problems. In *Proceedings of the 2021 CHI Conference on*

Human Factors in Computing Systems, CHI '21, New York, NY, USA. Association for Computing Machinery.

Wu, Z. and Ericson, B. J. (2024). SQL puzzles: Evaluating micro Parsons problems with different types of feedback as practice for novice programmers. In *Conference on Human Factors in Computing Systems - Proceedings*. Association for Computing Machinery.

Wu, Z., Ericson, B. J., and Brooks, C. (2023). Using micro Parsons problems to scaffold the learning of regular expressions. In *Proceedings of the 2023 Conference on Innovation and Technology in Computer Science Education V. 1*, pages 457–463. Association for Computing Machinery.