

Competições Femininas em Tecnologia como Estratégia para o Desenvolvimento do Pensamento Computacional na Educação Básica

Cibele Soares de Almeida¹, Francielle de Mattos¹, Lina Garcés¹

¹ Inst. Ciências Matemáticas e de Computação – Universidade de São Paulo (ICMC-USP) 13560-500 – São Carlos – SP – Brazil

{cibelesoares, francielle.mattos, linagarcés}@usp.br

Abstract. *This paper investigates the learning of Computational Thinking in projects by female elementary school students in a technology competition. A mixed-methods case study was conducted to analyze the influence of age and competitive context on the development of these skills. Twenty-two projects were analyzed via a systematic inspection protocol, mapping conditionals, loops, functions, and data access. The qualitative analysis, based on the four pillars of Computational Thinking, revealed cognitive evidence, age differences, and that the skills emerge even in initial contexts, with students without prior experience producing projects of similar quality.*

Resumo. *Este artigo investiga a aprendizagem do Pensamento Computacional em projetos de alunas do ensino básico em uma competição tecnológica. Foi realizado um estudo de caso com métodos mistos para analisar a influência da faixa etária e do contexto competitivo no desenvolvimento dessas habilidades. Analisaram-se 22 projetos via protocolo de inspeção sistemática, mapeando condicionais, laços, funções e acesso a dados. A análise qualitativa, baseada nos quatro pilares do Pensamento Computacional, revelou evidências cognitivas, diferenças etárias e que as habilidades emergem mesmo em contextos iniciais, com alunas sem experiência prévia produzindo projetos de qualidade semelhante.*

1. Introdução

Nos últimos anos, observa-se um aumento na participação das mulheres nas áreas de Ciência, Tecnologia, Engenharia e Matemática (CTEM) [CNN Brasil 2023]. Contudo, apesar desse avanço, a presença feminina ainda permanece inferior quando comparada a áreas como saúde e ciências humanas [Araújo 2019, Wang 2021].

O reduzido número de mulheres na área de Ciência da Computação nem sempre foi uma realidade. As mulheres tiveram papel protagonista na história da computação, porém fatores históricos, culturais e estruturais contribuíram, ao longo do tempo, para seu afastamento desse campo e para a consolidação de estereótipos de gênero. Esses estereótipos reforçaram a associação das mulheres a funções ligadas ao cuidado, enquanto os homens passaram a ser mais vinculados a atividades relacionadas à tecnologia, inovação e raciocínio lógico [Misa 2021, Santos 2023, Viegas 2025].

Apesar desse afastamento histórico, a relevância das mulheres na computação é amplamente reconhecida. Gurer (2002) destaca a atuação de mulheres pioneiras, como

Ada Byron (Lady Lovelace), considerada a primeira programadora da história, Grace Murray Hopper, fundamental para o desenvolvimento do primeiro compilador e da linguagem COBOL, além das programadoras que integraram a equipe responsável pelo ENIAC.

A literatura aponta que a baixa participação feminina na computação pode ser enfrentada por meio de iniciativas que incentivem meninas, desde a infância, a romper com estereótipos de gênero e a acreditar em sua capacidade. Nesse sentido, diferentes iniciativas têm contribuído para a ampliação da participação feminina na área, como Meninas Digitais¹, GRACE², Women in Technology³, IT Women Mentoring Program⁴ e Girls Who Code⁵.

Uma dessas iniciativas, foco deste estudo, é a TechSchool, um programa com duração de três meses, estruturado no formato de competição tecnológica baseada em projetos. Nessa iniciativa, meninas de 8 a 18 anos desenvolvem soluções para problemas sociais por meio da programação de aplicativos móveis. Embora estratégias dessa natureza tenham demonstrado potencial para estimular o interesse feminino pelas áreas de tecnologia, engenharia e computação, ainda são escassos os estudos que investigam, de forma aprofundada, o impacto dessas competições no desenvolvimento de habilidades de pensamento computacional nessa faixa etária.

Diante desse cenário, este estudo tem como objetivo investigar elementos do Pensamento Computacional nos projetos desenvolvidos por alunas do ensino básico durante a competição, analisando como esses elementos se manifestam no contexto da aprendizagem de programação em uma iniciativa voltada à formação de meninas. Para nortear o estudo, foram definidas as seguintes questões de pesquisa: (i) em que medida as alunas aplicam os elementos do Pensamento Computacional durante a realização da competição tecnológica? e (ii) se a faixa etária influencia o nível de compreensão e a forma como esses elementos são aplicados no desenvolvimento dos projetos.

Este estudo contribui para o campo da Informática na Educação ao investigar práticas de avaliação do Pensamento Computacional em contextos de aprendizagem não formais. O artigo está organizado da seguinte forma: a Seção 2 apresenta a fundamentação teórica; a Seção 3 discute os trabalhos relacionados; a Seção 4 descreve os procedimentos metodológicos adotados; a Seção 5 apresenta e discute os resultados obtidos; e, por fim, a Seção 6 apresenta as considerações finais e possibilidades de pesquisas futuras.

2. Pensamento Computacional: Fundamentos Teóricos e Método de Avaliação

O Pensamento Computacional (PC) consolidou-se como um tema central nas pesquisas em Educação e Computação. Embora não exista uma definição única e consensual, há concordância na literatura quanto à sua relevância para a formação dos indivíduos. O PC pode ser compreendido tanto como uma habilidade cognitiva, conforme proposto por

¹<https://meninas.sbc.org.br>

²<https://grace.icmc.usp.br>

³<https://www.womenintechology.org/>

⁴<https://www.lacnic.net/4883/2/lacnic/it-women-mentoring>

⁵<https://girlswhocode.com>

Wing (2006, 2008) e Román González (2015), quanto como uma competência educacional, quando analisado sob a perspectiva de documentos curriculares nacionais, como a BNCC (2018), e de frameworks internacionais, como a CSTA (2016) e a ISTE (2016).

As bases conceituais do PC remontam ao construcionismo de Papert (1971), ao definir o “pensar com o computador”. Posteriormente, Wing (2006) sistematizou o termo Pensamento Computacional como uma habilidade cognitiva essencial, comparável à leitura, escrita e ao cálculo, relacionada à resolução de problemas, ao desenvolvimento de sistemas e à compreensão de comportamentos fundamentados nos princípios da Ciência da Computação. Em trabalhos posteriores, Wing (2011) amplia essa compreensão ao definir o Pensamento Computacional como os processos de pensamento envolvidos na formulação de problemas e de suas soluções de modo que estas possam ser executadas por humanos ou agentes computacionais. Essa perspectiva reforça o caráter cognitivo, estratégico e interdisciplinar do Pensamento Computacional, para além do simples aprendizado de programação.

A partir dessa consolidação, diferentes autores ampliaram e refinaram o conceito. Grover e Pea (2013, 2018) definem o PC como um conjunto de competências cognitivas necessárias para formular problemas passíveis de resolução por métodos computacionais. Brennan e Resnick (2012) compreendem o PC como um conjunto de práticas, conceitos e perspectivas desenvolvidas em atividades de programação. De modo convergente, a Sociedade Brasileira de Computação (SBC) (2019) caracteriza o PC como um processo cognitivo que envolve a formulação de problemas e a expressão de soluções executáveis por humanos ou máquinas.

No contexto educacional, diversos pesquisadores buscaram identificar os componentes estruturantes do Pensamento Computacional. Entre essas contribuições, destaca-se o trabalho de Brackmann (2017), que sistematiza quatro pilares fundamentais: decomposição, reconhecimento de padrões, abstração e algoritmos. Esses pilares oferecem uma base conceitual amplamente adotada em propostas pedagógicas e instrumentos de avaliação do PC.

No presente estudo, os quatro pilares do Pensamento Computacional fundamentaram a construção dos critérios de avaliação utilizados na análise dos códigos em programação em blocos desenvolvidos pelas alunas durante a competição. Os critérios analíticos não foram criados originalmente nesta pesquisa, mas elaborados e adaptados a partir das contribuições teóricas de Wing (2006, 2011), Brennan e Resnick (2012), Sociedade Brasileira de Computação (2019) e Brackmann (2017). As definições e categorias analíticas foram ajustadas às especificidades da programação em blocos, às plataformas utilizadas e aos objetivos investigativos desta pesquisa.

A **decomposição** refere-se à capacidade de dividir problemas complexos em partes menores, manifestando-se na criação de procedimentos e na organização modular dos blocos. O **reconhecimento de padrões** envolve a identificação de similaridades entre soluções, observável na reutilização de estruturas lógicas, como condicionais. A **abstração** diz respeito à seleção de elementos relevantes, evidenciada pelo uso de procedimentos reutilizáveis, componentes independentes e estruturas de dados. Por fim, os **algoritmos** correspondem à definição de sequências lógicas de passos, expressas na organização coerente dos blocos, no uso de decisões e de estruturas de repetição na

construção das soluções.

3. Trabalhos Relacionados

A avaliação do Pensamento Computacional (PC) em contextos educacionais tem sido amplamente investigada por meio de diferentes instrumentos e abordagens metodológicas, sobretudo em atividades de programação em blocos [Sousa 2020, França 2015]. Nesse sentido, diversos estudos propõem frameworks e ferramentas voltados à identificação de evidências do desenvolvimento do PC a partir da análise de artefatos produzidos pelas alunas, como códigos e projetos.

Entre os instrumentos mais conhecidos, destaca-se o Dr. Scratch [Moreno-León 2015], que analisa automaticamente projetos desenvolvidos em Scratch e atribui pontuações associadas a conceitos do Pensamento Computacional. Embora amplamente utilizado, esse instrumento concentra-se no código final, com menor atenção ao processo formativo e ao contexto pedagógico. O *Progression of Early Computational Thinking* (PECT) [Seiter 2013], por sua vez, descreve níveis progressivos de desenvolvimento do Pensamento Computacional em crianças iniciantes, baseando-se na observação de comportamentos e padrões emergentes. Já o *Computational Thinking Test* (CT-Test) [Román-González 2015] propõe uma avaliação estruturada por meio de questões de múltipla escolha relacionadas a conceitos computacionais, sendo amplamente utilizado em investigações diagnósticas sobre o desenvolvimento do Pensamento Computacional.

Estudos mais recentes também apontam limitações dessas abordagens avaliativas quando centradas exclusivamente em códigos estáticos ou em respostas objetivas. Nesse sentido, [Raabe 2020] destacam que instrumentos tradicionais podem não capturar adequadamente os processos cognitivos mobilizados pelas alunas durante a resolução de problemas, como estratégias, tentativas, experimentações e tomada de decisões ao longo da atividade.

Apesar de suas contribuições, esses instrumentos assumem, em geral, contextos de aprendizagem homogêneos e tarefas padronizadas, não contemplando explicitamente ambientes não formais, competitivos ou mediados por mentoria. Além disso, observa-se uma lacuna de estudos que investiguem o desenvolvimento do PC em plataformas de programação em blocos voltadas à criação de aplicativos móveis, como o MIT App Inventor⁶ e o Thunkable⁷, especialmente no contexto brasileiro.

Estudos recentes também têm investigado estratégias de avaliação do Pensamento Computacional em ambientes de programação em blocos e desenvolvimento de aplicativos móveis. Nesse sentido, [Araujo 2024] analisam a relação entre fundamentos de programação e conceitos do Pensamento Computacional a partir de portfólios desenvolvidos em App Inventor, utilizando rubricas avaliativas aplicadas a projetos de alunas da Educação Básica.

No que se refere ao ensino de Pensamento Computacional por meio da Aprendizagem Baseada em Projetos (Project-Based Learning – PBL), estudos indicam que essa abordagem favorece o engajamento, a autonomia e a aprendizagem significativa ao promover a resolução de problemas autênticos e a construção de artefatos computacionais,

⁶<https://appinventor.mit.edu/>

⁷<https://thunkable.com/>

contribuindo para o desenvolvimento de habilidades como decomposição, abstração, reconhecimento de padrões e algoritmos [Hussin 2018, Santos 2019, Alves 2019]. Contudo, a literatura ainda carece de investigações que articulem PBL, avaliação sistemática do PC e contextos educacionais não formais e competitivos.

Pesquisas recentes também destacam a relevância de iniciativas de gênero na Computação, evidenciando o papel de programas extracurriculares, competições e ações de mentoria na promoção do interesse, da permanência e da autoconfiança de meninas nas áreas de Ciência e Tecnologia [Santos 2023, França 2021, Viegas 2025]. Entretanto, grande parte desses trabalhos concentra-se em análises de participação ou motivação, havendo menos estudos que examinem, de forma sistemática, os artefatos computacionais produzidos como evidência de aprendizagem.

Diante desse cenário, o presente estudo se diferencia ao investigar o desenvolvimento do Pensamento Computacional em um contexto competitivo e com abordagem baseada em projetos, voltado a alunas do gênero feminino de ensino básico sem experiência prévia em programação. Para isso, propõe-se a aplicação de um protocolo de inspeção sistemática adaptado, que permite analisar evidências dos quatro pilares do Pensamento Computacional — decomposição, reconhecimento de padrões, abstração e algoritmos — a partir de códigos desenvolvidos em plataformas de programação em blocos para aplicativos móveis. Dessa forma, o trabalho contribui para o avanço das discussões sobre avaliação do Pensamento Computacional em contextos não formais de aprendizagem, articulando Aprendizagem Baseada em Projetos, programação em blocos e iniciativas voltadas à formação de meninas na Computação.

4. Metodologia

A presente pesquisa adotou um estudo de caso, conforme definido por Yin [Yin 2015], abordagem adequada para a investigação de fenômenos contemporâneos em contextos reais. Essa estratégia permitiu analisar como alunas do ensino básico mobilizam o Pensamento Computacional no contexto de uma competição tecnológica.

O estudo seguiu uma abordagem de métodos mistos, combinando análise quantitativa, baseada na contagem e classificação das estruturas de código, e análise qualitativa, voltada à interpretação de decisões cognitivas associadas aos quatro pilares do Pensamento Computacional: decomposição, reconhecimento de padrões, abstração e algoritmos.

O contexto investigado caracteriza-se por uma competição tecnológica baseada em projetos, alinhada aos princípios da Aprendizagem Baseada em Projetos, possibilitando observar a aplicação de conceitos computacionais em um ambiente educacional não formal e colaborativo, inclusive entre alunas sem experiência prévia em programação.

4.1. Contexto da Competição Tecnológica

O contexto empírico desta pesquisa é a TechSchool, uma competição anual de tecnologia voltada a meninas do ensino básico, realizada ao longo de 12 semanas e estruturada segundo a Aprendizagem Baseada em Projetos. As alunas, organizadas em equipes de até cinco integrantes, desenvolvem soluções tecnológicas para problemas reais de suas comunidades, alinhados aos Objetivos de Desenvolvimento Sustentável (ODS) da ONU.

Os projetos articulam desafios locais a temas da Agenda 2030, como saúde e bem-estar, educação de qualidade, igualdade de gênero, redução das desigualdades e sustentabilidade ambiental, promovendo uma aprendizagem significativa, socialmente contextualizada e voltada ao protagonismo feminino por meio da tecnologia. As etapas da competição são apresentadas na Figura 1.

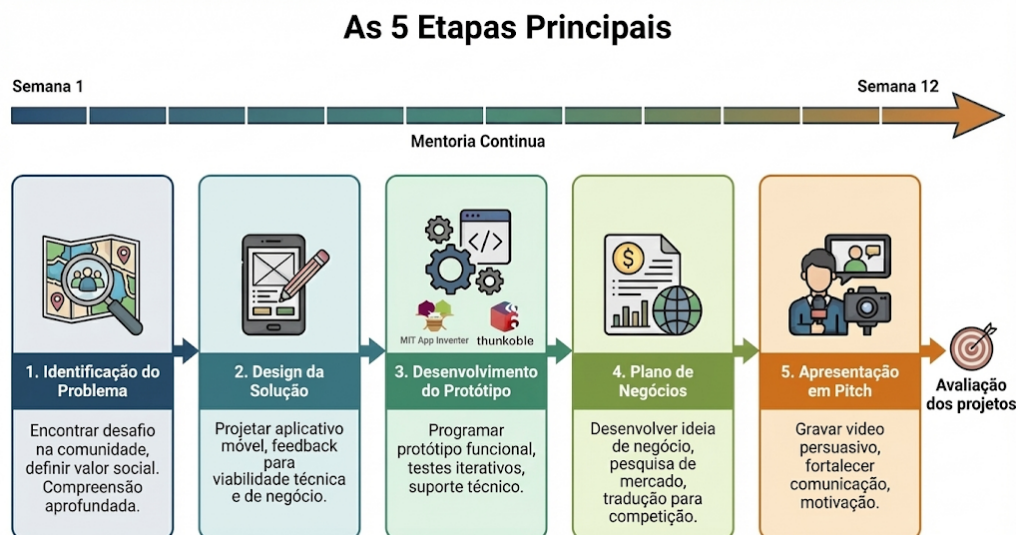


Figura 1. Etapas da competição de tecnologia

A participação é voluntária e não exige conhecimento prévio em programação. Durante todo o processo, cada equipe recebe acompanhamento contínuo de mentores das áreas de Computação e Tecnologia, cuja atuação consistiu em orientar o desenvolvimento dos projetos, esclarecer dúvidas e oferecer apoio técnico e metodológico, sem fornecer soluções prontas, preservando o protagonismo das alunas.

A competição ocorreu em formato remoto, combinando encontros síncronos semanais e atividades assíncronas. Ao final, os projetos foram avaliados por jurados externos, considerando critérios técnicos, inovação, impacto social e apresentação. As notas, atribuídas em escala de 0 a 5, foram agregadas por média aritmética e normalizadas, possibilitando a comparação entre categorias etárias e a construção do ranking final.

Embora os projetos abordassem diferentes problemas sociais, a competição adotava critérios comuns de avaliação técnica e funcional, de modo a minimizar diferenças muito acentuadas de escopo entre as soluções desenvolvidas.

4.2. Caracterização dos Projetos Analisados

Foram analisados 22 projetos, distribuídos em três categorias etárias: Iniciante (8–12 anos, 6 projetos), Júnior (13–15 anos, 7 projetos) e Sênior (16–18 anos, 9 projetos). Todos os projetos foram desenvolvidos nas plataformas MIT App Inventor e Thunkable, baseadas em programação em blocos. As alunas não possuíam experiência prévia em programação, caracterizando um contexto de aprendizagem inicial apropriado para investigar a emergência do Pensamento Computacional.

4.3. Coleta e Protocolo de Análise dos Dados

Os dados foram coletados por meio de inspeção visual sistemática dos códigos em blocos, acessados a partir do repositório da competição. Para assegurar rigor metodológico e replicabilidade, foi desenvolvido um protocolo padronizado de análise, ilustrado na Figura 2.

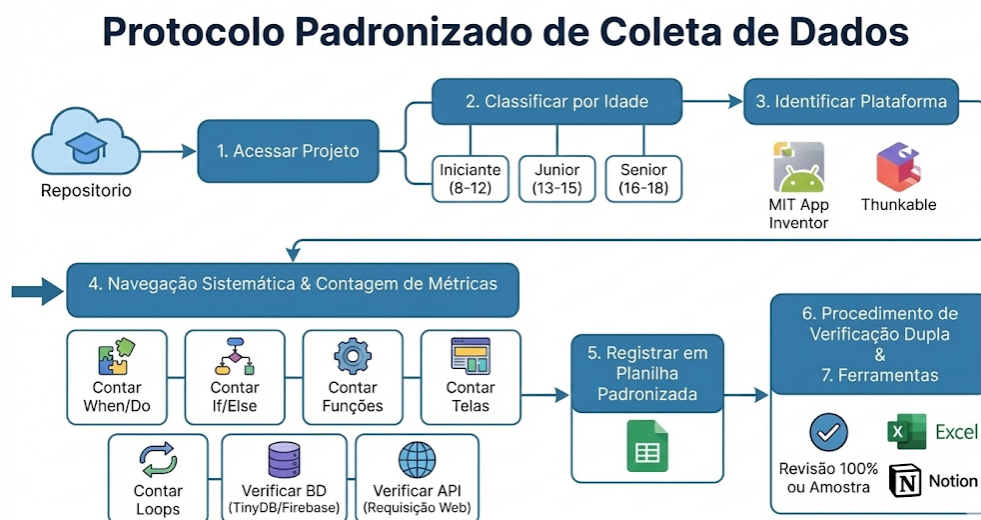


Figura 2. Método da análise dos dados

O protocolo envolveu: identificação do projeto e da categoria etária; navegação sistemática pelas telas; mapeamento dos blocos por tipo; e registro em planilha estruturada. A classificação e a contagem das estruturas foram realizadas com dupla checagem, e divergências foram resolvidas por consenso.

Para a análise do Pensamento Computacional, os blocos *when/do* foram classificados como tratadores de eventos, enquanto as estruturas internas foram analisadas separadamente. Blocos *if/else* foram registrados como evidências de algoritmos de seleção, e estruturas iterativas (*forever*, *repeat*, *for each*) como evidências de algoritmos de repetição e reconhecimento de padrões.

4.4. Procedimentos de Análise Quantitativa e Qualitativa, Validade e Confiabilidade

A análise quantitativa consistiu na contagem dos diferentes tipos de blocos de programação, com tratamento por estatísticas descritivas. As notas atribuídas pelos jurados foram utilizadas como indicadores externos de qualidade, após normalização para permitir comparações entre categorias etárias.

A análise qualitativa baseou-se nos quatro pilares do Pensamento Computacional, interpretando como as equipes aplicaram decomposição, reconhecimento de padrões, abstração e algoritmos nos códigos desenvolvidos. A síntese dos métodos de avaliação é apresentada na Figura 3, e o mapeamento entre os pilares e os elementos observáveis no código, na Figura 4.

Não foi calculada métrica estatística formal de concordância interavaliadores, uma



Figura 3. Métodos da Avaliação de Aprendizagem do PC

 Pilar de PC	 Estrutura de de Código	 Indicador Observável
 Decomposição	Funções/Procedimentos	Criação de blocos reutilizáveis nomeados; divisão de código em componentes lógicos
 Algoritmos	If/Else, Loops	Lógica condicional para casos distintos; iteração sobre coleções de dados
 Padrões	Repetição de estruturas	Reuso de padrões em contextos distintos; estruturas análogas em diferentes telas
 Abstração	BD, APIs	Uso de serviços abstratos; integração com sistemas externos

Figura 4. Mapeamento entre os pilares do Pensamento Computacional

vez que a análise se concentrou em contagens objetivas de estruturas de código, com divergências resolvidas por consenso. A validade e a confiabilidade foram fortalecidas pelo uso de protocolo padronizado, dupla checagem e triangulação entre análises quantitativa e qualitativa. O protocolo completo e a planilha de análise foram disponibilizados como material suplementar em repositório público (Zenodo), com DOI 10.5281/zenodo.18839650.

4.5. Aspectos éticos

A pesquisa foi conduzida observando os princípios éticos aplicáveis às pesquisas em contexto educacional. Foram adotados procedimentos para assegurar a participação voluntária das alunas, a autorização dos responsáveis legais, quando aplicável, a confidencialidade das informações, a anonimização dos dados e a preservação da identidade das alunas e das equipes em todas as etapas da pesquisa.

5. Resultados e Discussões

Foram analisados 22 projetos desenvolvidos pelas alunas da TechSchool, distribuídos nas categorias etárias: Iniciante (6 projetos), Júnior (7 projetos) e Sênior (9 projetos). A análise adotou uma abordagem de métodos mistos, combinando a contagem sistemática de blocos de programação com a inspeção qualitativa da sofisticação lógica das estruturas desenvolvidas.

A Figura 5 apresenta a distribuição dos blocos associados aos elementos do Pensamento Computacional. Nos projetos do nível Iniciante, observou-se elevada frequência de blocos *when/do*, caracterizando uma programação fortemente orientada a eventos. A análise qualitativa indicou que as alunas mais jovens construíram a interatividade por meio de múltiplos gatilhos independentes, utilizando predominantemente seleções simples (*if*) e laços contínuos (*forever*) para o controle de estados básicos da interface.

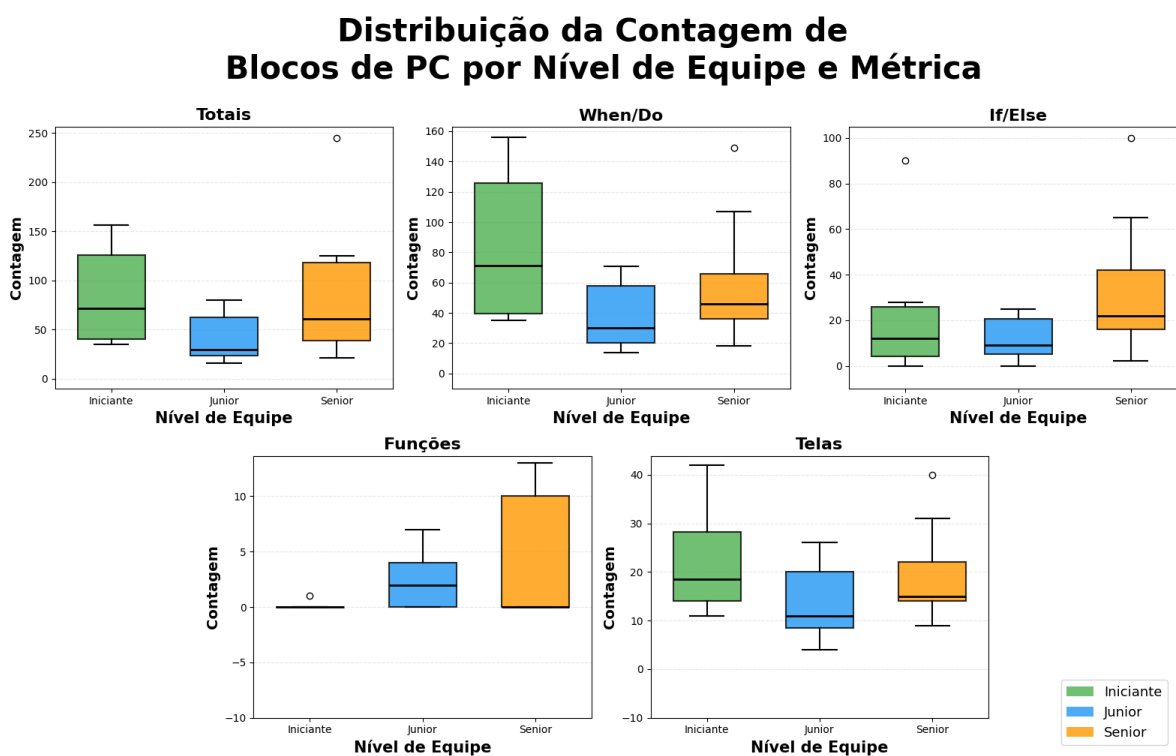


Figura 5. Avaliação de aprendizagem do Pensamento Computacional

No nível Júnior, observou-se uma transição na decomposição dos problemas, com menor quantidade total de blocos e o surgimento do uso de funções, indicando avanços iniciais em abstração e modularização do código. Já no nível Sênior, os projetos apresentaram maior domínio de estruturas complexas, com uso frequente de condicionais compostas (*if/else*), funções e estruturas de iteração sobre dados (*for each*), evidenciando maior abstração e organização algorítmica.

A Figura 6 mostra a utilização de recursos avançados de abstração. Enquanto nenhum projeto do nível Iniciante utilizou banco de dados ou APIs, nos níveis Júnior e Sênior observou-se aumento progressivo desses recursos, especialmente no nível Sênior, indicando o desenvolvimento gradual da capacidade de abstração e a construção de soluções mais robustas.

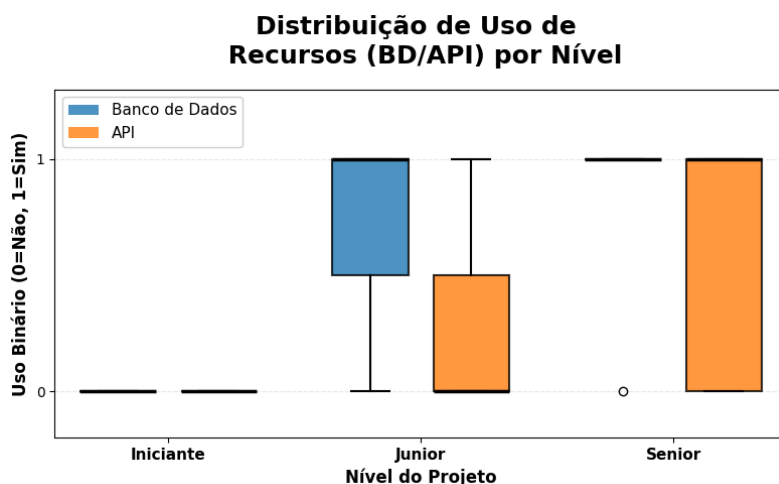


Figura 6. Distribuição do uso de recursos avançados de abstração (BD/API)

As notas normalizadas atribuídas pelos jurados (Figura 7) revelam que os projetos do nível Iniciante obtiveram avaliações elevadas, apesar da menor complexidade técnica. Nos níveis Júnior e Sênior, embora os projetos incorporassem recursos mais avançados, as médias não foram substancialmente superiores. Esses resultados indicam que, no contexto analisado, fatores como criatividade, completude funcional e adequação ao problema social estiveram associados a avaliações positivas em todas as faixas etárias.

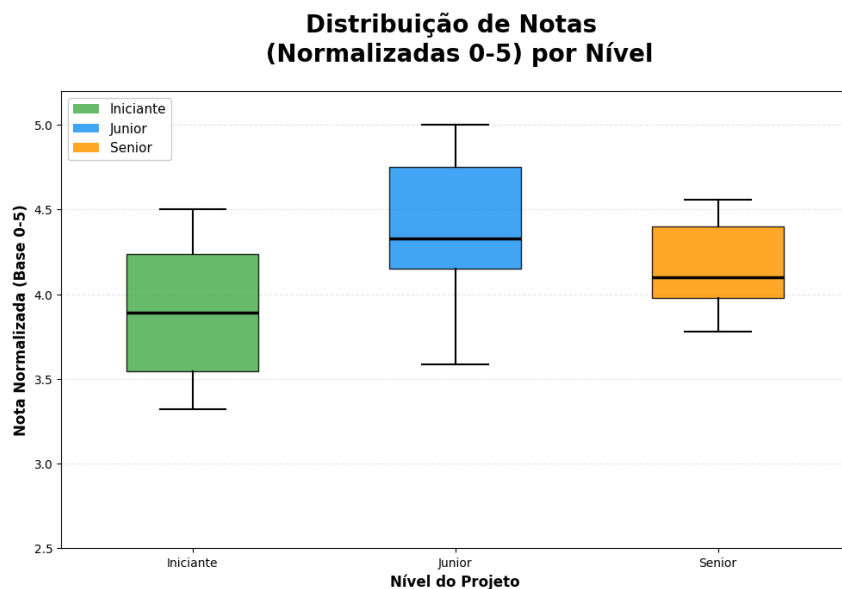


Figura 7. Boxplot comparando as notas finais dos projetos das alunas

A análise qualitativa confirmou a presença dos quatro pilares do Pensamento Computacional em todos os níveis. A decomposição manifestou-se na organização modular dos projetos; o reconhecimento de padrões, na reutilização de estruturas lógicas; a abstração, no uso de funções, banco de dados e APIs; e os algoritmos, na construção de sequências lógicas progressivamente mais sofisticadas, especialmente nos projetos do nível Sênior.

De forma geral, os resultados quantitativos, qualitativos e de avaliação externa permitiram responder às questões de pesquisa propostas. Observou-se que todas as equipes aplicaram efetivamente os pilares do Pensamento Computacional, mesmo sem experiência prévia em programação. Embora os projetos dos níveis mais avançados apresentem maior complexidade técnica, a qualidade final das soluções não se mostrou diretamente proporcional à sofisticação do código, indicando que diferentes estratégias cognitivas podem resultar em soluções eficazes em distintas faixas etárias.

6. Conclusões

Este estudo investigou a aprendizagem do Pensamento Computacional em projetos desenvolvidos por alunas do ensino básico alunas de uma competição tecnológica, com foco na análise da influência da faixa etária e do contexto competitivo no desenvolvimento dessas habilidades. Por meio de um estudo de caso com métodos mistos, foi possível identificar evidências consistentes dos quatro pilares do Pensamento Computacional, decomposição, reconhecimento de padrões, abstração e algoritmos, mesmo em um cenário de aprendizagem inicial e em um período relativamente curto de formação.

Os resultados indicam que todas as equipes, independentemente da faixa etária, foram capazes de desenvolver o Pensamento Computacional na construção de soluções funcionais, criativas e socialmente relevantes. Observou-se que a idade influencia a complexidade técnica das soluções, refletida no maior uso de funções, banco de dados e APIs nos níveis mais avançados. Entretanto, a qualidade global dos projetos, avaliada por critérios externos, não apresentou diferenças proporcionais à sofisticação técnica, evidenciando que criatividade, motivação e alinhamento com problemas reais podem compensar limitações técnicas iniciais.

Outro resultado relevante foi a identificação de evidências associadas ao desenvolvimento do Pensamento Computacional entre as alunas. Mesmo sem experiência prévia em programação, as alunas conseguiram estruturar os códigos, indicando que iniciativas dessa natureza podem contribuir significativamente para a inclusão e permanência de meninas nas áreas de Computação e Tecnologia desde a educação básica.

Do ponto de vista da Informática na Educação, este estudo contribui ao apresentar uma estratégia de avaliação do Pensamento Computacional baseada na análise de artefatos produzidos em um contexto não formal de aprendizagem. Além disso, os resultados reforçam o potencial de competições tecnológicas fundamentadas na Aprendizagem Baseada em Projetos para promover experiências de aprendizagem em programação e ampliar a participação de meninas na Computação.

Como limitações do estudo, destaca-se o número restrito de projetos analisados e o foco em um único contexto competitivo, o que sugere cautela na generalização dos resultados. Embora todos os grupos tenham recebido mentoria semelhante, é possível que diferenças na interação entre mentor e equipe tenham influenciado o desenvolvimento dos projetos. Além disso, como os projetos foram desenvolvidos em equipes, os resultados refletem o desempenho coletivo dos grupos, não sendo possível isolar a contribuição individual de cada integrante para o desenvolvimento do Pensamento Computacional. Como trabalhos futuros, propõe-se a ampliação da amostra, a comparação com contextos não competitivos e o acompanhamento longitudinal das alunas, a fim de compreender os impactos de longo prazo dessas iniciativas no desenvolvimento do Pensamento Computacional.

onal e na trajetória acadêmica das alunas.

Declaração sobre uso de Inteligência Artificial

Seguindo as orientações da SBC, declaramos que ferramentas de Inteligência Artificial (IA) foram utilizadas somente como apoio editorial, sem interferir no conteúdo científico do trabalho. O Gemini Colab foi usado para auxiliar na elaboração de figuras ilustrativas. As escolhas metodológicas, a análise dos dados e a interpretação dos resultados ficaram inteiramente a cargo dos autores. Nenhuma ferramenta de IA participou da construção do conteúdo científico, das análises ou das conclusões apresentadas.

7. Agradecimentos

Às agências brasileiras de fomento à pesquisa pelo apoio ao desenvolvimento deste trabalho, em especial à Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES), ao Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) e à REBECA – Rede de Ensino Brasileira para Engenharias e Ciências Exatas. Este trabalho recebeu apoio do CNPq por meio do projeto Processo nº 440425/2024-7 e da Bolsa de Pós-Doutorado (Processo nº 195424/2025-5).

Referências

- Alves, Socorro Vânia Lourenço e Alves, E. e. B. P. B. (2019). Programação e aprendizagem baseada em projetos como estratégias no ensino de pensamento computacional para crianças e adolescentes. In *Anais dos Workshops do VIII Congresso Brasileiro de Informática na Educação (CBIE 2019)*, pages 749–758, Porto Alegre. SBC.
- Araújo, Elias Dias de e Teixeira, D. J. e. M. M. V. (2024). Análise da relação entre fundamentos de programação e conceitos do pensamento computacional em portfólios de estudantes de um curso ead de programação. In *Anais do Simpósio Brasileiro de Informática na Educação (SBIE)*, pages 2037–2050. Sociedade Brasileira de Computação.
- Araújo, Mariana Tonini e Tonini, A. M. (2019). A participação das mulheres nas áreas de STEM (Science, Technology, Engineering and Mathematics). *Revista de Ensino de Engenharia*, 38(3).
- Brackmann, C. P. (2017). *Desenvolvimento do pensamento computacional através de atividades desplugadas na educação básica*. PhD thesis, Universidade Federal do Rio Grande do Sul, Porto Alegre. Disponível em: <https://lume.ufrgs.br/handle/10183/172208>. Acesso em: 05 mai. 2026.
- Brasil. Ministério da Educação (2018). Base nacional comum curricular. Disponível em: <http://basenacionalcomum.mec.gov.br/>. Acesso em: 23 mai. 2026.
- Brennan, Karen e Resnick, M. (2012). New frameworks for studying and assessing the development of computational thinking. In *Proceedings of the 2012 Annual Meeting of the American Educational Research Association (AERA)*, Vancouver, Canada. Acesso em: 07 mai. 2026.
- CNN Brasil (2023). Participação feminina na ciência brasileira cresce 29% em 20 anos, diz relatório.

- Computer Science Teachers Association (2016). K-12 computer science standards. Disponível em: <https://www.csteachers.org/page/standards>. Acesso em: 03 mai. 2026.
- França, J. B. S. e. o. (2021). Desenvolvendo o pensamento computacional de meninas através de histórias. In *Anais do Simpósio Brasileiro de Informática na Educação (SBIE)*. Sociedade Brasileira de Computação.
- França, Rozelma Soares de e Tedesco, P. (2015). Ensinando computação e promovendo o pensamento computacional com o uso do scratch. In *Anais do Workshop de Informática na Escola (WIE)*. Sociedade Brasileira de Computação.
- Grover, Shuchi e Pea, R. (2013). Computational thinking in K-12: A review of the state of the field. *Educational Researcher*, 42(1):38-43.
- Grover, Shuchi e Pea, R. (2018). Computational thinking: A competency whose time has come. In Sentance, Sue e Waite, J. e. H. K., editor, *Computer Science Education: Perspectives on Teaching and Learning*, pages 19-38. Bloomsbury Academic.
- Gürer, D. (2002). Women in computing history. *ACM SIGCSE Bulletin*, 34(2):116-120.
- Hussin, A. (2018). Education 4.0 made simple: Ideas for teaching. *International Journal of Education and Literacy Studies*, 6(3):92-98.
- International Society for Technology in Education (2016). Iste standards for students. Disponível em: <https://www.iste.org/standards/iste-standards-for-students>. Acesso em: 23 mai. 2026.
- Misa, T. J. (2021). Dynamics of gender bias in computing: a new dataset significantly revises both scholarly assessment and popular understanding about gender bias in computing. *Communications of the ACM*, 64(6):76-83.
- Moreno-León, Jesús e Robles, G. e. R.-G. M. (2015). Dr. Scratch: Automatic analysis of scratch projects to assess and foster computational thinking. *RED. Revista de Educación a Distancia*, 46:1-23. Disponível em: <https://revistas.um.es/red/article/view/240111>. Acesso em: 05 mai. 2026.
- Papert, S. (1971). Teaching children thinking. Technical Report 247, MIT Artificial Intelligence Laboratory, Cambridge, MA. Disponível em: <https://dspace.mit.edu/handle/1721.1/5835>. Acesso em: 07 mai. 2026.
- Raabe, André e Viana, C. e. C. L. (2020). Ct puzzle test: Em direção a uma avaliação interativa do pensamento computacional. In *Anais do XXXI Simpósio Brasileiro de Informática na Educação (SBIE 2020)*, pages 1683-1692. Sociedade Brasileira de Computação.
- Román-González, M. (2015). Computational thinking test: Design guidelines and content validation. In *Proceedings of the 7th International Conference on Education and New Learning Technologies (EDULEARN15)*, pages 2436-2444, Barcelona, Spain. IATED. Disponível em: <https://library.iated.org/view/ROMANGONZALEZ2015COM>. Acesso em: 10 mai. 2026.
- Santos, A. P. S. e Medeiros, F. P. e. P. A. e. R. J. J. G. e. M. R. F. (2019). Uso de projetos em salas de aula dos Institutos Federais: uma análise sob a ótica da aprendizagem

- baseada em projetos e das competências do século 21. *Revista Principia: Divulgação Científica e Tecnológica do IFPB*, 1(44):113–121.
- Santos, Nayara Dias dos e Marczak, S. (2023). Fatores de atração, evasão e permanência de mulheres nas áreas da computação. In *Anais do XVII Women in Information Technology (WIT)*, pages 136–147, João Pessoa, PB, Brasil. Sociedade Brasileira de Computação.
- Seiter, Laina e Foreman, W. (2013). Assessing computational thinking in the early grades: the progression of early computational thinking model. In *Proceedings of the 2013 ACM Conference on International Computing Education Research (ICER)*, pages 123–130.
- Sociedade Brasileira de Computação (2019). Diretrizes para ensino de computação na educação básica. Disponível em: <https://www.sbc.org.br/educacao/diretrizes-para-ensino-de-computacao-na-educacao-basica>. Acesso em: 05 mai. 2026.
- Sousa, L. d. L. e. o. (2020). Programação em blocos aplicada no ensino do pensamento computacional: Um mapeamento sistemático da literatura. In *Anais do Simpósio Brasileiro de Informática na Educação (SBIE)*. Sociedade Brasileira de Computação.
- Viegas, Renata e Oliveira, A. e. S. L. e. M. M. M. (2025). Educação em computação sem padrões limitantes: Caminhos para diversidade de gênero no ensino superior. In *Anais do Simpósio Brasileiro de Educação em Computação (EDUCOMP) – Mesas Temáticas*, pages 135–139, Juiz de Fora, MG, Brasil. Sociedade Brasileira de Computação.
- Wang, Lucy Lu e Stanovsky, G. e. W. L. e. E. O. (2021). Gender trends in computer science authorship: under optimistic projection models, gender parity is forecast to be reached after 2100. *Communications of the ACM*, 64(3):78–84.
- Wing, J. M. (2006). Computational thinking. *Communications of the ACM*, 49(3):33–35.
- Wing, J. M. (2011). Research notebook: Computational thinking—what and why? *The Link Magazine*, pages 20–23.
- Yin, R. K. (2015). *Estudo de Caso: Planejamento e Métodos*. Bookman, Porto Alegre, 5 edition.