

DESENVOLVENDO O PENSAMENTO COMPUTACIONAL NO ENSINO MÉDIO INTEGRADO PARTIR DA PROGRAMAÇÃO SCRATCH

Gildevan Santos de Araújo¹
Caroline Carvalho Nunes de Souza²
Demson Oliveira Souza³
Márcia Jussara Hepp Rehfeldt⁴
Maria Claudete Schorr⁵

RESUMO

O presente relato de experiência visa apresentar resultados provenientes da aplicação de uma sequência didática usando o ambiente on-line de programação Scratch, com alunos de uma turma do 1º ano do Ensino Médio Técnico Integrado em Informática. O objetivo é descrever os avanços e desafios dessa aplicação, que visou fomentar o Pensamento Computacional na referida turma a partir da abordagem da Computação Plugada. Os aportes teóricos deste trabalho baseiam-se nas concepções de Wing (2008, 2017), Brennan e Resnick (2012), Marji (2014), Moreno-León et al. (2015) e Resnick (2018), assim como, no Complemento à Base Nacional Comum Curricular (BNCC) – Computação de 2022, que referencia a habilidade do Pensamento Computacional como eixo fundamental do Ensino da Computação na Educação Básica. A abordagem metodológica do trabalho é qualitativa e exploratória, e os dados foram coletados por meio de observação participante, questionários e registros de atividades, sem deixar de considerar os aspectos éticos decorrentes do processo investigativo (Gil, 2008). Os resultados indicam, que os estudantes, em seu primeiro ano de curso, apesar de praticarem técnicas iniciais de programação, não as relacionam ao contexto do Pensamento Computacional. Além disso, o fato da linguagem de programação Scratch apresentar uma abordagem visual com menor complexidade, possibilitou que a maioria dos alunos se sentissem engajados e motivados ao desenvolverem seus projetos de programação.

Palavras-chave: Programação Visual, Scratch, Computação Plugada, Educação Básica, BNCC Computação.

INTRODUÇÃO

O avanço contínuo das tecnologias digitais consolidou o Pensamento Computacional (PC) como uma competência fundamental a ser desenvolvida no contexto da Educação Básica. O cenário educacional atual tem sido moldado por novas diretrizes que reconhecem essa necessidade, destacando-se a recente homologação das Normas sobre Computação na Educação

¹ Graduando do Curso de Licenciatura em Computação do Instituto Federal de Educação, Ciência e Tecnologias da Bahia - IFBA, campus Jacobina, gildevan.s.a@gmail.com;

² Graduanda do Curso de Licenciatura em Computação do Instituto Federal de Educação, Ciência e Tecnologias da Bahia - IFBA, campus Jacobina, carol.lcifba@gmail.com;

³ Doutorando em Ensino de Ciências Exatas pelo Programa de Pós-graduação Profissional em Ensino de Ciências Exatas da Univates - Lajeado - RS, demson.souza@universo.univates.br;

⁴ Doutora em Informática na Educação pela Universidade Federal do Rio Grande do Sul - RS, Professora do Programa de Pós-Graduação em Ensino de Ciências Exatas da Univates, mreinfeldt@univates.br;

⁵ Doutora em Informática na Educação pela Universidade Federal do Rio Grande do Sul - RS, Professora do Programa de Pós-Graduação em Ensino de Ciências Exatas da Univates, mclaudetesw@univates.br.

Básica, que servem como complemento à Base Nacional Comum Curricular (BNCC). Este documento referenda o Pensamento Computacional como um eixo fundamental do Ensino da Computação na Educação Básica, destacando a importância de promover habilidades cognitivas como decomposição, abstração, reconhecimento de padrões e a criação de algoritmos, todas essenciais para a resolução de problemas complexos no século XXI.

Nesse contexto, surgem desafios sobre como mediar o ensino desses conceitos seguindo essas novas diretrizes. Nesse sentido, ambientes de programação visual como o Scratch, aparecem como estratégia mediadora promissora, facilitando o engajamento e a formação de estudantes na compreensão desses conceitos.

O presente artigo configura-se como um relato de experiência, nele serão apresentados os resultados provenientes da aplicação de uma sequência didática que utilizou o ambiente on-line de programação em blocos Scratch. Esta sequência foi aplicada a alunos de uma turma do 1º ano do Ensino Médio Técnico Integrado em Informática. O objetivo do trabalho é descrever os avanços e desafios observados durante essa aplicação, que teve como meta central fomentar o Pensamento Computacional na referida turma a partir da abordagem da Computação Plugada.

METODOLOGIA

A abordagem metodológica do trabalho é qualitativa e exploratória. Os dados foram coletados por meio de observação participante, questionários e registros de atividades. Esta coleta considerou os aspectos éticos decorrentes do processo investigativo, conforme indicado por Gil (2008). A metodologia foi ainda embasada na análise dos referenciais legais e acadêmicos como a BNCC - Computação sobre o ensino de Pensamento Computacional na Educação Básica.

Destaca-se que o curso foi realizado com uma turma do 1º ano do Ensino Médio Técnico em Informática do Instituto Federal da Bahia – Campus Jacobina. Os estudantes encontravam-se no período final do ano letivo e a proposta didática visou desenvolver práticas introdutórias de programação em linguagem visual Scratch com os mesmos, bem como, mobilizando suas habilidades referentes ao Pensamento Computacional. Além disso, buscou-se identificar se já haviam tido experiências prévias com esse conceito e com o uso da referida linguagem de programação.

Como proposta avaliativa do curso, os dezoito alunos da turma envolvida foram convidados a produzir ou remixar um jogo no Scratch. Neste caso, como materiais utilizados para o desenvolvimento do curso, foram utilizados computadores conectados à internet no

laboratório de informática, quadro branco, projetor e notebook. No que se refere a ferramentas de apoio didático, pode-se contar com o ambiente on-line Scratch para desenvolvimento dos projetos, assim como, google forms para coletar informações nos momentos inicial e final do curso. Além disso, o a ferramenta on-line Plickers⁶ foi usada como uma atividade diagnóstica em forma de quiz.

A formação ocorreu em 6 encontros, sendo 2 semanais de 1 hora e 40 minutos, totalizando um período de 10 horas de curso durante o mês de dezembro de 2024. O processo foi previamente planejado por meio de uma sequência didática e para sintetizar a prática desenvolvida apresta-se Quadro 1 abaixo.

Quadro 1- Etapas de desenvolvimento do curso

Encontro	Conteúdos/Atividades	Estratégias Didático-Pedagógicas
1º Encontro – Introdução ao curso e ao Pensamento Computacional	- Apresentação dos objetivos do curso e do Scratch. - Aplicação de questionário diagnóstico inicial a respeito do tema. - Discussão sobre o que é Pensamento Computacional e suas aplicações.	- Exposição dialogada e roda de conversa com apoio visual. - Uso do google <i>forms</i> para aplicação do questionário.
2º Encontro – Um pouco sobre o desenvolvimento de jogos e os primeiros passos no Scratch	- Apresentação do ambiente Scratch. - Introdução aos elementos de um jogo como: Narrativa, regras, objetivos, <i>sprites</i> (personagens), cenário, mecânica, física, sons, <i>feedback</i> e etc.	- Apresentação do ambiente on-line Scratch. - Apresentação dos elementos presentes nos jogos utilizando um slide e fazendo algumas demonstrações no Scratch.
3º Encontro – Desenvolvendo um jogo modelo	- Desenvolvimento de um jogo do tipo labirinto (<i>pac-man</i>) de maneira a explorar os recursos disponíveis no scratch, além de poder aproveitar elementos fora da plataforma, importando <i>sprites</i> e outros elementos.	- Aula prática mediada para construção do jogo.
4º Encontro – Construção ou	- A turma será dividida em trios para construção ou	- Aprendizagem baseada em projeto (ABP).

⁶ O Plickers é uma ferramenta digital usada principalmente em ambientes educacionais para realizar avaliações rápidas e interativas, sem a necessidade de que os alunos usem celulares ou computadores. Pode ser acessado no seguinte endereço <https://www.plickers.com/>.

remixagem de um jogo.	remixagem de um jogo aplicando os conceitos trabalhados.	- Acompanhamento e suporte nas dúvidas que forem surgindo.
5º Encontro – Continuação do desenvolvimento do jogo.	- Os trios continuarão o desenvolvimento do jogo.	- Acompanhamento e suporte nas dúvidas que forem surgindo.
6º Encontro – Apresentação e avaliação.	- Apresentação dos jogos produzidos. - Discussão sobre as aprendizagens e desafios. - Aplicação de um questionário avaliativo do curso.	- Apresentação e discussão sobre cada projeto. - Aplicação de formulário pelo google forms para avaliar como foi o curso na visão dos alunos.

Fonte: Elaborado pelos autores (2025).

No Quadro 2, que apresenta-se a seguir, estão descritas as habilidades da BNCC – Computação relacionadas ao eixo Pensamento Computacional, contemplando tanto os anos finais do Ensino Fundamental quanto o Ensino Médio, que serviram de referência para o planejamento e desenvolvimento das atividades do curso.

Quadro 2 – Habilidades BNCC Computação relacionadas

Habilidade	Ações desenvolvidas	Atividades relacionadas
“(EF69CO02) Elaborar algoritmos que envolvam instruções sequenciais, de repetição e de seleção usando uma linguagem de programação.” (Brasil, 2022, p. 56)	- Construção do jogo pac-man como exemplificação de desenvolvimento de projetos Scratch, servindo como referência para o projeto final dos alunos.	- Mediação do proponente do curso na construção do jogo pac-man. - Desenvolvimento do jogo em equipe;
(EF69CO03) Descrever com precisão a solução de um problema, construindo o programa que implementa a solução descrita. (Brasil, 2022, p. 56)	- Desenvolvimento do projeto organizado pela equipe. - Apresentação do trabalho final.	- Orientação do proponente as equipes. - Socialização do trabalho que foi produzido.
“(EM13CO01) Explorar e construir a solução de problemas por meio da reutilização de partes de soluções existentes.” (Brasil, 2022, p. 62)	- Remixagens de códigos já produzido anteriormente. - Busca e aplicação de <i>sprites</i> (personagens) disponíveis na internet.	- Reaproveitamento de códigos no desenvolvimento do trabalho final.
(EM13CO02) Explorar e construir a solução de problemas por meio de	- Planejamento e desenvolvimento do projeto	- Correção de falhas durante o processo de desenvolvimento.

refinamentos, utilizando diversos níveis de abstração desde a especificação até a implementação. (Brasil, 2022, p. 62)	- Ajustes no projeto final para corrigir falhas no desenvolvimento.	
--	---	--

Fonte: Elaborado pelos autores (2025).

Com relação ao processo avaliativo, além das percepções geradas durante as aulas, utilizou-se a ferramenta Dr. Scratch⁷, desenvolvida por Moreno-León, Robles e Román-González (2015), com a proposta de analisar automaticamente os níveis de habilidades do pensamento computacional (PC) resultantes no desenvolvimento de cada jogo produzido pelos estudantes. Esta ferramenta web gratuita permite aos educadores e alunos avaliar projetos desenvolvidos no ambiente Scratch, identificando o grau de desenvolvimento das competências do pensamento computacional de forma objetiva e sistematizada.

O Dr. Scratch analisa sete dimensões fundamentais do pensamento computacional presentes nos projetos: abstração e decomposição de problemas, pensamento lógico, sincronização, paralelismo, controle de fluxo algorítmico, interatividade com o usuário e representação de dados. Para cada uma dessas dimensões, a ferramenta atribui uma pontuação em três níveis hierárquicos de proficiência: básico (1 ponto), desenvolvimento (2 pontos) e proficiente (3 pontos). A soma das pontuações de todas as dimensões resulta em uma classificação geral do projeto, que pode variar de 0 a 21 pontos.

Essa abordagem de avaliação automática possibilitou uma visão sobre o que cada grupo conseguiu produzir, levando em conta os níveis de habilidades do PC desenvolvidas em cada projeto. Além disso, o Dr. Scratch identifica eventuais "*code smells*" (maus hábitos de programação), como uso de nomes não significativos para variáveis, repetição desnecessária de código, blocos de código que nunca são executados e inicialização incorreta de atributos, fornecendo feedback específico para melhoria.

REFERENCIAL TEÓRICO

Segundo concepções de Wing (2006; 2008; 2017), o Pensamento Computacional deve ser visto não apenas como uma habilidade para cientistas da computação, mas como um

⁷ Ferramenta online gratuita desenvolvida para avaliar automaticamente projetos criados no Scratch, com o objetivo de medir as habilidades de Pensamento Computacional (PC) demonstradas pelos alunos. Pode ser acessado no endereço <https://www.drscratch.org/>.

conjunto de habilidades cognitivas fundamentais para todos os cidadãos. A mesma autora define essa habilidade como um processo de pensamento envolvido na resolução de problemas e na expressão de suas soluções de forma que um computador (humano ou máquina) possa executá-las. Além disso, ela descreve que o PC envolve o uso de conceitos computacionais essenciais como algoritmos (criação de passos para resolver um problema), abstração (focar no essencial e ignorar detalhes irrelevantes) e decomposição (dividir um problema complexo em partes menores).

Uma outra definição proposta por Karen Brennan e Mitchel Resnick (2012) transpõem a definição de Wing para um contexto de aprendizagem prática, especificamente com o Scratch. Eles propõem um "*framework*" que associa o desenvolvimento do Pensamento Computacional a experiências de aprendizagem que são colaborativas e criativas. Ou seja, não basta apenas "aprender a programar", mas usar a programação como um meio para se expressar, criar projetos e colaborar.

Com relação ao Scratch, Majed Marji (2014) justificar a escolha desta linguagem de programação e enfatiza a importância de sua manipulação em blocos (visual) para facilitar o uso por usuários sem conhecimento avançado de programação. Isso significa reduzir a complexidade da sintaxe (escrever o código corretamente), permitindo que os alunos foquem na lógica, na estruturação de ideias e na resolução de problemas, facilitando a introdução aos conceitos do PC.

Com base nas ideias de Moreno-León *et al.* (2015), torna-se relevante desenvolver um processo de avaliação dos códigos produzidos no Scratch, para isso, esses mesmos autores desenvolveram a ferramenta "Dr. Scratch". A partir desta tecnologia é possível analisar projetos de programação visual (como os feitos no Scratch) visando avaliar e fomentar o desenvolvimento das habilidades do Pensamento Computacional. O trabalho deles valida o Scratch como um ambiente acessível e, ao mesmo tempo, robusto o suficiente para desenvolver e medir competências computacionais em alunos da educação básica.

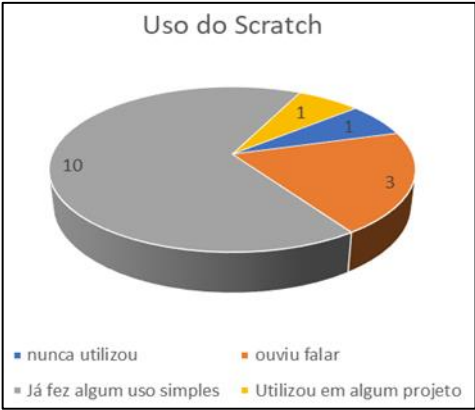
No contexto brasileiro, surge o Complemento à BNCC-Computação, que representa o documento que norteia o processo de inclusão do Pensamento Computacional como um eixo fundamental da Educação Básica (Brasil, 2022).

RESULTADOS E DISCUSSÃO

De início vamos descrever alguns resultados baseados nos questionários respondidos no início e final do curso. No caso do questionário inicial foi respondido por um total de quinze

alunos, e dentre as perguntas, uma delas é se eles já tinham tido acesso a alguma linguagem de programação. Como os alunos eram do ensino médio técnico em informática, a resposta de todos foi, sim. No que se refere ao uso do Scratch, quatro alunos nunca haviam utilizado, dez destes alunos já haviam feito uso e apenas um utilizou em algum momento para o desenvolvimento de algum projeto. Constatou-se nas repostas iniciais, que todos já tinham utilizado alguma linguagem de programação ou a maioria já ter desenvolvido algum trabalho com o Scratch, entretanto, em nenhum dos casos haviam desenvolvido alguma atividade relacionada de forma específica ao tema pensamento computacional. Um destes dados pode ser visualizado no gráfico da Figura 1.

Figura 1 - Representação gráfica parcial dos dados iniciais

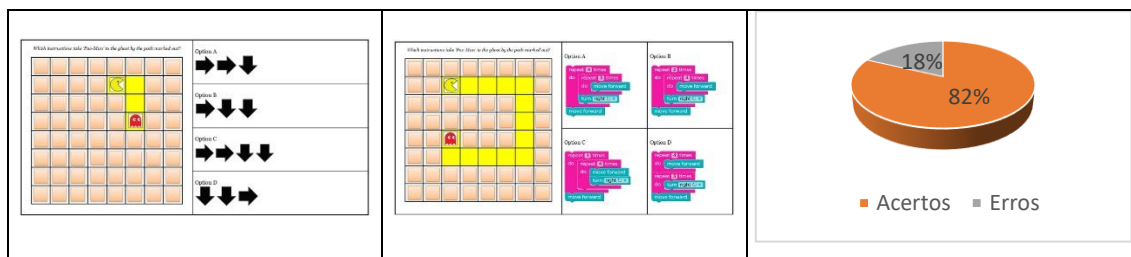


Fonte: Elaborado pelos autores (2025).

Além deste questionário, tivemos uma atividade no Plickers, em formato de quiz, que serviu para analisar como estava o raciocínio lógico dos alunos. A atividade consistiu na construção e aplicação de 10 questões de múltipla escolha, possibilitando o aluno analisar e definir qual a resposta correta. As questões foram variadas e aumentando a dificuldade no decorrer da atividade, como demonstra a Figura 2. Resultando em um percentual médio de acerto de 82% na turma, o que indica um bom desempenho geral e compreensão satisfatória dos conceitos trabalhados.

Figura 2 – Exemplo de atividade e média da turma

Exemplos de questões aplicadas	Média da turma
--------------------------------	----------------

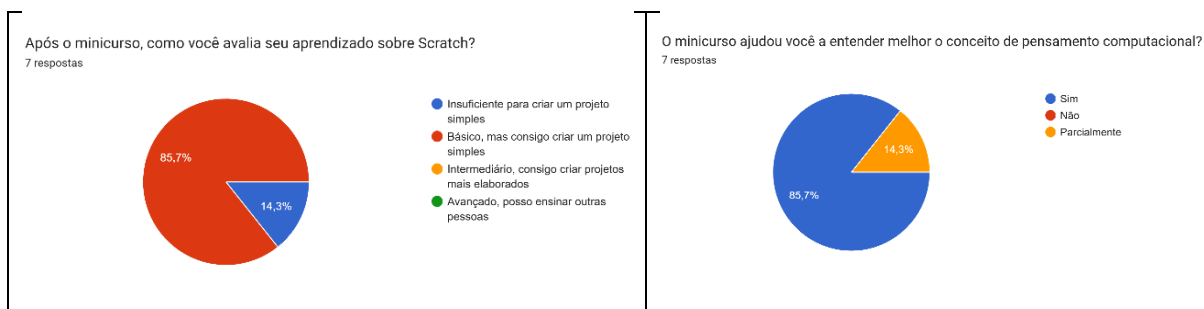


Fonte: Elaborado pelos autores (2025).

Com base nessas informações iniciais, é possível perceber que, mesmo após dois anos da aprovação do complemento de Computação na BNCC, e considerando uma formação voltada à área de Informática, ainda pode ser observado uma presença limitada de discussões sobre Pensamento Computacional. Nesse sentido, o curso proporcionou aos participantes o primeiro contato efetivo com essa temática, favorecendo o reconhecimento de sua importância, bem como a compreensão de como uma linguagem de programação em blocos, como o Scratch, pode auxiliar nesse processo de aprendizagem.

Continuando nossa análise, passamos para o questionário final, com a proposta de avaliarmos o resultado do curso. Nesse caso, apenas sete alunos responderam este formulário, ainda assim, os dados permitiram traçar uma perspectiva geral dos resultados, conforme demonstrado na Figura 3. No que se refere à autoavaliação dos alunos sobre o uso do Scratch, seis participantes (85,7%) declararam sentir-se aptos a produzir um projeto simples, enquanto um aluno (14,3%) considerou sua capacidade ainda insuficiente. Em relação à compreensão dos conceitos de Pensamento Computacional, seis alunos (85,7%) afirmaram ter compreendido os conceitos trabalhados, ao passo que um participante (14,3%) relatou ter alcançado apenas uma compreensão parcial.

Figura 3 – Avaliação final do curso

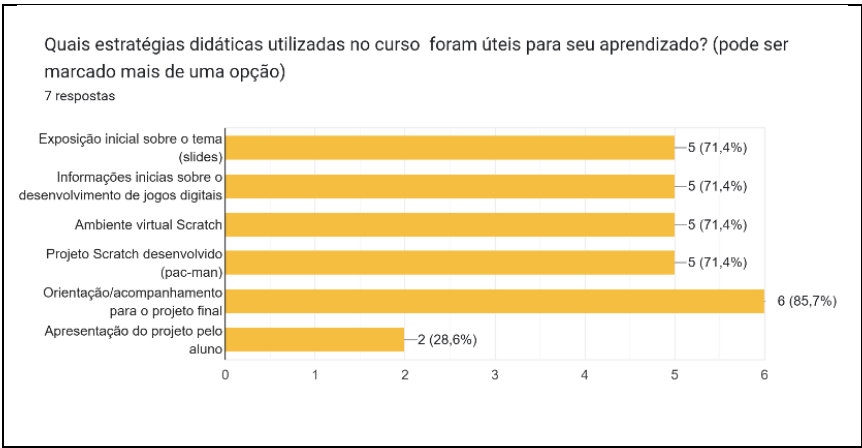


Fonte: Elaborado pelos autores (2025).

Com base nos dados e percepções geradas em sala de aula, acredita-se que o curso proporcionou uma boa compreensão dos conceitos de Pensamento Computacional (PC). O

desenvolvimento de projetos utilizando a ferramenta digital Scratch mostrou-se uma estratégia promissora para o aumento do interesse e do engajamento dos alunos. No projeto final, os participantes tiveram liberdade de escolha entre criar um jogo do zero ou realizar o remix de um já existente, adotando uma proposta alinhada à metodologia do “aprender fazendo”. Essa abordagem se aproxima do que defende Papert (1980), ao afirmar que as pessoas aprendem de forma mais eficaz quando estão ativamente envolvidas na construção de algo que possua significado pessoal. Outro aspecto relevante desse processo construtivo, conforme destacado por Brennan e Resnick (2012), é a mediação docente, entendida como a assistência pedagógica que orienta e potencializa a aprendizagem. Isso se torna mais interessante quando os próprios alunos reconhecem isso, para tanto que esse é um dos pontos que eles melhor avaliaram como estratégia didática, como apresenta Figura 4 a seguir.

Figura 4 – Avaliação das estratégias do curso



Fonte: Elaborado pelos autores (2025).

No quadro 3, apresentamos os resultados conceituais gerados pela avaliação do Dr.Scratch (básico, em desenvolvimento ou proficiente) referentes aos níveis de cada habilidade do PC mobilizadas nos 4 projetos produzidos como trabalho final no Scratch.

Quadro 3 – Resultados com níveis de habilidades do PC com base na avaliação do Dr.Scratch

Análise das Habilidades desenvolvidas nos Projetos				
Conceito Analisados	Projeto 1	Projeto 2	Projeto 3	Projeto 4
Abstração	1	1	1	1
Paralelismo	3	3	1	1
Lógica	1	3	1	1
Sincronização	3	3	0	2
Controle de fluxo	2	2	2	2

Interatividade com o usuário	2	2	2	2
Representação de dados	3	3	1	3
Pontuação (Máximo 21)	15	17	8	12
Nível Alcançado em PC	Proficiente	Proficiente	Desenvolvimento	Desenvolvimento

Fonte: Elaborado pelos autores com base na rubrica de Moreno-león *et al.* (2025).

Com base na rubrica elaborada por Moreno-León *et al.* (2015), foi possível organizar um quadro com os resultados das avaliações das programações dos jogos desenvolvidos pelos onze alunos participantes, distribuídos em trios e uma dupla, como atividade final do curso. A rubrica contemplou sete habilidades do Pensamento Computacional, atribuídas em índices numéricos (Básico = 1, Em Desenvolvimento = 2 e Proficiente = 3) permitindo quantificar o desempenho de cada grupo em cada dimensão analisada. Conforme descrito pelos autores, a soma dos índices na linha “pontuação” indica o nível geral de proficiência alcançado no projeto, seguindo a seguinte escala - 0: Habilidade não verificada; 1–7: Nível Básico; 8–14: Em Desenvolvimento; 15–21: Proficiente. Neste trabalho, optamos por analisar apenas os resultados provenientes da rubrica adaptada de Moreno-León *et al.* (2015), com foco em apresentar uma visão geral e comparativa do desempenho dos alunos a partir das dimensões avaliadas.

De modo geral, observa-se que os Projetos 1 e 2 atingiram o nível Proficiente, indicando domínio das habilidades de pensamento computacional. Já os Projetos 3 e 4 encontram-se em fase de desenvolvimento, com avanços parciais em controle de fluxo e interatividade. As dimensões mais consolidadas foram “Controle de fluxo”, “Interatividade com o usuário” e “Representação de dados”, enquanto “Abstração” e “Lógica” se mostraram como pontos a serem reforçados.

A “abstração” foi a habilidade que se manteve padrão entre os menores conceitos e atribuída a todos os projetos, embora citado por Wing (2008) como o conceito mais significativo do pensamento computacional, é o que oferece maior complexidade em ser avaliado. Naturalmente, a abstração se torna presente em todos os projetos, pois ao programar essa capacidade é peculiar ao processo. Embora exista parâmetros como os percorridos por Poloni (2018), baseados em estudos de psicologia da aprendizagem, como os de Piaget, não procuramos estabelecer essa relação entre os saberes, mas cientes desta alternativa para complementar a análise e avaliação do progresso dos cursistas.

Gostaríamos de destacar que o processo avaliativo não se limitou ao uso da rubrica. Embora ela tenha servido como um importante instrumento para mensurar o desenvolvimento

técnico dos participantes, nossa avaliação buscou ir além da pontuação obtida em cada dimensão. O acompanhamento contínuo do progresso individual e coletivo ao longo do curso permitiu uma compreensão mais ampla e qualitativa da evolução dos participantes.

Esse acompanhamento possibilitou observar não apenas a aplicação de conceitos de programação, mas também aspectos como o engajamento, a colaboração entre as duplas, a autonomia na resolução de problemas e a capacidade de refletir sobre as próprias produções. Dessa forma, o processo avaliativo assumiu um caráter formativo, valorizando o percurso de aprendizagem e não apenas o resultado final.

Assim, compreendemos que a combinação entre instrumentos objetivos, como rubricas e ferramentas de análise automática, e a observação pedagógica atenta dos educadores oferece uma visão mais humanizada da aprendizagem, respeitando as particularidades e os diferentes ritmos de desenvolvimento dos estudantes. Conforme destaca Perrenoud (1999), a avaliação formativa deve ser entendida como um processo de orientação do percurso educativo de cada aluno e contribuindo para a construção de saberes mais significativos.

CONSIDERAÇÕES FINAIS

Os resultados obtidos ao longo do curso evidenciam um progresso significativo no desenvolvimento das habilidades de Pensamento Computacional (PC) dos participantes. A experiência prática com a programação de jogos no Scratch, aliada ao engajamento individual e colaborativo, bem como à mediação pedagógica constante, mostrou-se fundamental para favorecer a construção dos projetos e, conseqüentemente, o avanço nas competências relacionadas ao PC.

Percebemos que o uso de rubricas contribui para uma dimensão educacional relevante, pois permite acompanhar e mapear o progresso dos estudantes. No entanto, compreendemos que as rubricas, embora úteis, não são instrumentos suficientes por si só, exigindo o apoio de outras formas de avaliação. Nesse sentido, estudos como os de Brennan e Resnick (2012) destacam a importância de compreender o raciocínio do programador, solicitando explicações sobre as decisões tomadas no código, o que amplia a análise além do simples produto final.

Reconhecemos, portanto, que o processo de aprendizagem em programação envolve um nível de subjetividade que ultrapassa o alcance de métricas automáticas ou pontuações numéricas. Essa constatação reforça a necessidade de uma abordagem mais pedagógica, humana e mediadora, que valorize a reflexão, a criatividade e o sentido atribuído pelos estudantes às suas produções. Assim, diante dos desafios que permeiam o ensino da computação

nas escolas públicas brasileiras, esta experiência contribui como um relato que evidencia caminhos possíveis para integrar o ensino do Pensamento Computacional de forma significativa e contextualizada.

REFERÊNCIAS

BRASIL. Ministério da Educação. **Normas sobre Computação na Educação Básica – Complemento à Base Nacional Comum Curricular (BNCC) – Computação.** Brasília: MEC, 2022.

BRENNAN, K.; RESNICK, M. 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, 2012.

GIL, A. C. **Métodos e técnicas de pesquisa social.** 6. ed. Editora Atlas SA, 2008.

MORENO-LEÓN, Jesús; ROBLES, Gregorio. **Analyze your Scratch projects with Dr. Scratch and assess your Computational Thinking skills.** 2015. Disponível em: <<https://jemole.me/replication/2015scratch/InferCT.pdf>> Acesso em: 19 out. 2025.

PAPERT, S. **Mindstorms: Children, Computers, and Powerful Ideas.** New York: Basic Books, 1980.

PERRENOUD, P. **Avaliação: da excelência à regulação das aprendizagens — entre duas lógicas.** Porto Alegre: Artmed, 1999.

POLONI, L. **Aprendizagem de programação mediada por uma linguagem visual: possibilidade de desenvolvimento do pensamento computacional.** Dissertação (Mestrado) - Universidade de Caxias do Sul. Caxias do Sul- RS, 2018.

RESNICK, M. **Lifelong Kindergarten: Cultivating Creativity through Projects, Passion, Peers, and Play.** Cambridge, MA: MIT Press, 2018.

WING, J. M. Computational thinking. **Communications of the ACM**, v. 49, n. 3, p. 33–35, 2006.

WING, J. M. Computational thinking and thinking about computing. **Philosophical Transactions of the Royal Society A**, v. 366, n. 1881, p. 3717–3725, 2008.

WING, J. M. The professionalization of computational thinking. **Communications of the ACM**, v. 60, n. 11, p. 28–30, 2017.