ISSN: 2358-8829

O USO DO SCRATCH NO ENSINO DO PENSAMENTO COMPUTACIONAL: UMA EXPERIÊNCIA FORMATIVA NO PIBID.

Jussiara Élcia Gomes Rodrigues [1]
Claudia Marcela Rodrigues Santos [2]
Joseane Mirtis de Queiroz Pinheiro [3]
Diego Rodrigues de Almeida [4]

RESUMO

Este trabalho relata uma experiência pedagógica desenvolvida no âmbito do Programa Institucional de Bolsa de Iniciação à Docência (PIBID), objetivando a prática do Pensamento Computacional na educação básica por meio da plataforma Scratch. Fundamentada na perspectiva construcionista, que valoriza a construção de artefatos tangíveis como base para o aprendizado, e nos princípios da aprendizagem criativa, a atividade estruturou-se na construção dirigida de um jogo digital. A estratégia metodológica aplicada priorizou o desenvolvimento passo a passo, onde licenciandos e alunos codificaram conjuntamente as mecânicas do jogo. Essa abordagem estratégica permitiu explorar sistematicamente as áreas da programação, garantindo que conceitos fundamentais — como algoritmos, variáveis, eventos e coordenadas cartesianas — fossem compreendidos na prática, vinculando cada subtarefa do jogo a uma competência lógica específica. Diferentemente da exploração livre, a construção guiada assegurou que os estudantes transitassem por múltiplos níveis de abstração e decomposição de problemas sem a frustração comum à sintaxe de linguagens tradicionais. Durante o processo, o ambiente de programação em blocos facilitou a depuração de erros e a experimentação lógica, transformando o computador em uma ferramenta de "aumento cognitivo" e autonomia intelectual. Os resultados indicam que a criação de jogos, quando estruturada pedagogicamente, não apenas motiva e engaja os estudantes, mas efetiva a alfabetização digital, promovendo o raciocínio lógico necessário para a resolução de problemas em diversas áreas do conhecimento.

Palavras-chave: Pensamento Computacional, Scratch, Ensino de Programação, PIBID.

INTRODUÇÃO

No cenário educacional atual, a presença da tecnologia dentro das escolas e na vida dos alunos exige o desenvolvimento de competências cognitivas que ultrapassem o mero manuseio técnico ou o uso passivo dos dispositivos digitais. Nesse contexto, destaca-se o Pensamento Computacional como uma habilidade analítica fundamental a ser integrada à capacidade de raciocínio de crianças e jovens, articulando-se à leitura, à escrita e à aritmética. Formulado conceitualmente por Jeannette Wing em 2006, o Pensamento Computacional

[1]Graduanda do Curso de Computação do Instituto Federal de Pernambuco - IFPE, rodrigues.jussiarag@gmail.com.
[2]Graduanda do Curso de Computação do Instituto Federal de Pernambuco - IFPE, cmrs3@discente.ifpe.edu.br.
[3]Doutora em Educação Matemática pela Universidade Estadual Paulista "Júlio de Mesquita Filho" – UNESP – SP, mirtis.queiroz@unesp.br.
[4]Doutor em Ciência da Computação pela Universidade Federal de Campina Grande - UFCG - PB, diego.rodrigues@afogados.ifpe.edu.br.

ISSN: 2358-8829

abrange o processo de resolução de problemas, o projeto de sistemas e a compreensão do comportamento humano por meio dos conceitos basilares da Ciência da Computação.

O desenvolvimento dessa forma de pensamento apoia-se em quatro pilares estruturantes: a decomposição de problemas complexos em partes menores, o reconhecimento de padrões, a abstração e a construção de algoritmos de acordo com Brackmann, 2017. No Brasil, a urgência dessa formação foi chancelada pela Base Nacional Comum Curricular (BNCC), que estabeleceu a Cultura Digital e o Pensamento Computacional como competências transversais indispensáveis na Educação Básica.

Nas escolas públicas, porém, a efetivação dessas diretrizes esbarra em limitações de infraestrutura e na falta de letramento digital básico dos estudantes, que muitas vezes restringem o contato com tecnologia ao uso recreativo de celulares (PELISSER; BORGES, 2020).

Foi a partir desse cenário que o subprojeto de Computação executado pelas licenciandas em Computação do IFPE – Campus Afogados da Ingazeira, vinculado ao PIBID, experienciaram o que será tratado neste artigo, relatando as experiências vividas por meio das atividades realizadas com turmas do 4º ano de uma escola pública municipal, em Afogados da Ingazeira – PE. O objetivo geral foi analisar as contribuições de uma sequência didática de construção guiada de um jogo de tabuleiro do tipo quiz no Scratch para o desenvolvimento do PC.

A programação nos Anos Iniciais encontra sustentação no Construcionismo de Papert (2008) e na Aprendizagem Criativa de Resnick, que deu origem ao Scratch. Nessa perspectiva, o computador deixa de ser uma máquina de ensinar e torna-se ferramenta de expressão autoral: ao criarem jogos e animações, os estudantes aprendem conceitos lógicos e práticas de projeto, como testar e depurar, e passam a se ver como criadores de tecnologia (BRENNAN; RESNICK, 2012).

METODOLOGIA

Este trabalho trata-se de um relato de experiência de intervenção pedagógica, de abordagem qualitativa, desenvolvido no âmbito do PIBID pelo subprojeto de computação exercido pelas discentes de Licenciatura em Computação do IFPE – Campus Afogados da Ingazeira. O planejamento articulou a Aprendizagem Baseada em Projetos (ABP) com a Instrução Direta Guiada. A combinação justificou-se pela necessidade de introduzir conceitos abstratos de forma estruturada e progressiva, aproveitando a familiaridade dos alunos com

ISSN: 2358-8829

jogos digitais. As licenciandas atuaram como mediadoras e os estudantes, como construtores de seus próprios artefatos.

O estudo foi realizado em uma escola pública municipal, em Afogados da Ingazeira – PE, que atende cerca de 278 alunos na Educação Infantil e nos Anos Iniciais. Foram participantes do projeto duas turmas do 4º ano (4º A e 4º B), com alunos de 10 a 11 anos e de 30 a 35 estudantes cada, incluindo alunos atendidos pelo Atendimento Educacional Especializado (AEE) e alunos em consolidação da leitura e da escrita. Em diagnóstico inicial, observou-se que os estudantes não tinham contato anterior com informática básica na escola, restringindo-se ao uso de celulares no convívio familiar.

O projeto foi executado no laboratório de informática que tinha disponível de 19 computadores desktop, dos quais 3 apresentavam falhas constantes (desligamento repentino). Para garantir uma máquina por aluno, cada turma foi dividida em dois grupos de 15 a 17 estudantes. Enquanto um grupo programava no laboratório, o outro desenvolvia atividades desplugadas e projetos complementares na sala regular, alternando-se a dinâmica. Isso eliminou o trabalho em trios e permitiu atendimento individualizado pelas bolsistas.

As intervenções ocorreram de 02 de setembro a 16 de dezembro de 2025, em 15 semanas letivas, com um encontro semanal por turma. Cada aula teve 50 minutos e cada encontro reuniu duas aulas consecutivas (1 hora e 40 minutos) com cada turma. Assim, em cada dia de atuação foram ministradas quatro aulas: duas para o 4º A e duas para o 4º B. A sequência foi organizada em nove oficinas temáticas, e 7 delas tiveram aplicação de 2 semanas para contemplar a turma em sua totalidade.

Oficina Scratch		
Data	**Nº da Aula**	**Oficina**
02/09/2025	01	Introdução: O que é jogo? Como construí-lo?
09/09/2025	02	Esquematizando o Cenário
16/09/2025	02	Esquematizando o Cenário
23/09/2025	03	Rolando Dados
30/09/2025	03	Rolando Dados
07/10/2025	04	Mapeando o Tabuleiro
14/10/2025	04	Mapeando o Tabuleiro
21/10/2025	05	Personagem 1
28/10/2025	05	Personagem 1
04/11/2025	06	Personagem 2
11/11/2025	06	Personagem 2
18/11/2025	07	Cenários Complementares

ISSN: 2358-8829

25/11/2025	07	Cenários Complementares
02/12/2025	08	Personagem 3
09/12/2025	08	Personagem 3
16/12/2025	09	Apresentação dos Jogos e Encerramento

Tabela 1 - Distribuição de Aulas

O produto final foi um jogo de tabuleiro interativo do tipo quiz, com dados e movimentação por coordenadas, construído no Scratch (versão offline instalada nas máquinas). A progressão das oficinas foi a seguinte:

- **Oficina 1 – Introdução: O que é jogo? Como construí-lo?:** apresentação da proposta e dos objetivos, e planejamento do jogo no papel, com ideia, regras e tema.
- **Oficina 2 – Esquematizando o cenário:** busca, seleção e salvamento de imagens, noções de organização de pastas e uso dos recursos "Carregar Cenário" e "Selecionar Ator" para montar o tabuleiro.
- **Oficina 3 – Rolando dados:** inserção dos dados como atores e das seis faces com "Carregar Fantasia", e simulação do sorteio com os blocos "*repita 10 vezes*", "*número aleatório entre 1 e 6*" e "*transmita mensagem*".
- **Oficina 4 – Mapeando o Tabuleiro:** introdução aos eixos X e Y do palco e criação das listas "posicaoX" e "posicaoY" com as coordenadas de cada casa, trabalhando a ideia de indexação.
- **Oficina 5 – Personagem 1:** Adição do personagem ao tabuleiro e integração da movimentação usando de variável de posição, condicionais (*se... então*), repetição e movimento suave (*deslize por... segs até x/y*), com alternância de turnos por mensagens.
- **Oficina 6 – Replicação para o personagem 2:** ajuste da orientação dos personagens e desafio de replicar e adaptar o código do jogador 1 para o jogador 2.
- **Oficina 7 – Cenários complementares:** inserção de cenários de transição e integração do quiz ao fluxo do jogo.
- **Oficina 8 – Personagem 3:** Adição do personagem ao tabuleiro, adição da lista de perguntas e integração de comandos de verificação.
- **Oficina 9 – Apresentação e encerramento:** socialização dos jogos finalizados entre as turmas.

Os dados foram obtidos através da observação diária das licenciandas, com registros em relatórios de trabalho; análise dos arquivos de projeto salvos ao final de cada aula, verificando a organização do código e uso correto de variáveis, repetições, eventos e

ISSN: 2358-8829

condicionais; e registro qualitativo da autonomia, através de progressão da turma, comportamento e colaboração entre pares. A análise foi interpretativa, com base nas dimensões de Brennan e Resnick (2012).

RESULTADOS E DISCUSSÃO

A trajetória do subprojeto começou longe do computador. Na Aula 01, as turmas do 4º ano tiveram que relembrar o que foi trabalhado com eles anteriormente sobre pensamento computacional e debater o que é um jogo e como fazê-lo tendo a notícia que ao final do percurso teriam construído dois jogos, um desplugado, feito na sala de aula, e outro plugado, programado no laboratório com o Scratch (e o qual é relatado neste trabalho). Antes de qualquer bloco, o que se pediu aos alunos foi imaginar: registraram por escrito o tema, as regras, os personagens e os objetivos finais de cada jogo. Esse começo no papel tem um sentido mais profundo do que parece. Para Papert (2008), aprende-se melhor quando se constrói algo que importa a quem constrói, e foi isso que a primeira aula ofereceu: antes de aprender uma ferramenta, os estudantes tornaram-se autores de uma ideia. Ao dividir o projeto em etapas e registrá-lo por escrito, a turma também iniciou, ainda sem nomear, o exercício de decomposição que Brackmann (2017) aponta como um dos pilares do Pensamento Computacional.

Na Aula 02, os alunos conheceram o destino da viagem: viram o jogo finalizado e entenderam aonde chegariam. A partir daí, começaram a dar forma ao seu próprio mundo, buscando, selecionando e salvando as imagens do cenário inicial, do cenário final e do tabuleiro, e usando os recursos “Carregar Cenário” e “Selecionar Ator”. Noções de arquivos, pastas e nomes de arquivo, já trabalhadas em aulas anteriores, ganharam um sentido prático novo, porque era preciso saber guardar e reencontrar o projeto para continuá-lo na aula seguinte. Para as turmas, o cenário e o tabuleiro deixaram de ser enfeites e passaram a ser a base visual e de jogabilidade de tudo o que viria depois.

Foi na Aula 03 que a programação de fato começou. Pela primeira vez os alunos usaram blocos de código, e o fizeram para dar vida a algo concreto: dois dados que giram quando clicados. Inseriram os dados como atores, adicionaram as seis faces numeradas e combinaram a repetição, *repita 10 vezes*, com o sorteio, *número aleatório entre 1 e 6*, para que o dado parecesse rolar até parar em um valor. O bloco *transmita* completou a cena, ao permitir que os atores se comunicassem e definissem de quem era a vez e quantas casas andar. Em termos do referencial de Brennan e Resnick (2012), mobilizaram-se ali laços, operadores e eventos, mas é provável que, para as crianças, o que ficou fosse mais simples e mais forte: o código que

ISSN: 2358-8829

escreveram fazia algo acontecer na tela. E isso não podia ser mais genuíno que um simples rolar de dados conseguiu arrancar inúmeros sorrisos e admiração pelo pouco construído.

A Aula 04 trouxe um desafio mais abstrato: para mover os personagens com precisão pelo tabuleiro, era preciso descobrir que cada casa tem um "endereço" no palco do Scratch, definido pelas coordenadas X e Y. Os alunos criaram as listas e as preencheram, casa por casa, com o bloco *adicione item à lista*, entendendo ao mesmo tempo a ideia de indexação, isto é, que cada item guardado tem uma posição própria. A matemática do plano cartesiano, que costuma parecer distante nessa idade, passou a ter utilidade imediata: sem os endereços certos, o personagem não chegaria à casa certa. Esse movimento de representar o tabuleiro por números é um exemplo claro de abstração, que Wing (2006) considera o cerne do pensamento computacional, e dialoga com o que Pelisser e Borges (2020) descrevem sobre o uso do PC para dar forma a noções matemáticas. Por ser algo mais distante do que estavam vivenciando, foi uma das partes que exigiu deles mais observação e atenção.

Se há uma aula em que o projeto mostrou seu lado mais exigente, é a Aula 05. Nela, os alunos reuniram tudo o que tinham construído para programar o movimento do primeiro personagem: a variável que guarda a posição atual, as listas de coordenadas, a condicional *se... então* e a repetição *repita... vezes*, de modo que o avanço respeitasse o número tirado no dado e o limite final do tabuleiro, com o deslize suave do bloco *deslize por... segs* e a troca de turno por *transmita*. O jogo começava a tomar forma, mas os blocos de código ficavam maiores e mais detalhados, e os alunos sentiram a dificuldade. O que se viu ali foi o momento em que a instrução guiada fez diferença: com o auxílio das professoras, a dificuldade foi vencida, e a turma saiu da aula com o primeiro personagem em movimento.

A Aula 06 foi, para as docentes, a mais surpreendente. Depois de ajustar as animações para que os personagens andassem no mesmo sentido do tabuleiro e de acompanhar a explicação passo a passo, os alunos foram desafiados a programar sozinhos o segundo personagem. Alguns tentaram lembrar de memória o caminho feito na aula anterior, outros foram olhar o código do personagem 1 e replicá-lo no personagem 2, cada um encontrando seu jeito de resolver o problema. Em termos computacionais, o que ocorreu foi o reconhecimento de um padrão e o reaproveitamento de um código que já funcionava, prática que Brennan e Resnick (2012) chamam de reutilizar e remixar e que Resnick e Rosenbaum (2013) associam ao ato de experimentar, montar e adaptar. Para as docentes, o episódio mostrou que o pensamento computacional trabalhado com a turma ao longo do ano já dava frutos, e que os alunos, mais do que seguir instruções, começavam a pensar por conta própria.

ISSN: 2358-8829

As Aulas 07 e 08 fecharam o percurso com a criação do personagem motor e alguns testes do jogo completo. Os alunos verificaram o funcionamento de tudo o que haviam programado, identificaram erros e fizeram os ajustes necessários até que o jogo funcionasse. Fosse um boneco andando para trás, uma pergunta que não setava sua resposta corretamente, ou um personagem que continuava andando pela tela mesmo tendo alcançado o fim do jogo. Nessa etapa, a depuração, que Brennan e Resnick (2012) descrevem como a prática de testar e corrigir o que não funciona como esperado, deixou de ser um conceito e virou vivência: os estudantes passaram pelo ciclo completo de desenvolvimento de um projeto digital. O ponto alto foi a satisfação visível ao ver o jogo funcionando: as docentes observaram entusiasmo e sentimento de conquista ao perceberem que o jogo feito por eles estava completo e funcional. É a perspectiva de expressar-se, na formulação dos mesmos autores, em que o aluno deixa de apenas consumir tecnologia e passa a criá-la.

Esse resultado não veio sem obstáculos. O laboratório dispunha de poucos computadores, e algumas máquinas desligavam sozinhas, o que apagava o progresso do aluno na aula e era uma preocupação constante. A divisão das turmas em dois grupos, em que metade programava enquanto a outra desenvolvia atividades desplugadas em sala, garantiu uma proporção adequada de alunos por máquina, e as professoras ajudaram a reconstruir o código perdido. A experiência acompanha o que Brackmann (2017) e Pelisser e Borges (2020) defendem: as limitações da escola pública não precisam impedir o ensino de computação quando há planejamento criativo e articulação entre atividades plugadas e desplugadas.

Houve também desafios pedagógicos, como manter a atenção e lidar com ritmos de aprendizagem diferentes. Para engajar as turmas, as docentes adotaram um sistema de recompensas, com brindes e sorteios para quem demonstrava bom comportamento e foco. Mas o que mais sustentou o ritmo foi a colaboração entre os próprios alunos: quem aprendia mais rápido ajudava os colegas com mais dificuldade, o que otimizou o tempo de regência e o andamento das atividades. Esse movimento reflete a perspectiva de conectar, em que criar com e para os outros faz parte da aprendizagem (BRENNAN; RESNICK, 2012), e sugere que o projeto em si, e não só os brindes, ajudou a manter as turmas envolvidas.

Ao final, todos os alunos concluíram a montagem do seu jogo de perguntas e respostas, o que indica a eficácia da metodologia neste contexto. O percurso deixou aprendizagens técnicas, como o uso de variáveis, estruturas de controle, sequenciamento lógico, repetição de padrões e depuração, e também comportamentais, como o raciocínio lógico, a autonomia, o domínio da plataforma e a colaboração. Para os alunos, foi um primeiro contato estruturado

ISSN: 2358-8829

com a programação, que desmistificou a tecnologia e os deslocou da condição de meros consumidores para a de criadores digitais, ideia que dialoga com a visão de Valente (2014) do computador como ferramenta para o aluno expressar ideias e construir conhecimento. A leitura e a digitação, indispensáveis para escrever o código e cadastrar as perguntas do jogo, foram exercitadas de forma natural, integrando o letramento digital à aprendizagem.

Para as licenciandas, a trajetória foi igualmente formadora. A regência, a gestão das turmas e o planejamento das aulas exigiram adaptação diante de imprevistos técnicos e de ritmos diferentes entre os alunos. A colaboração foi mútua: enquanto as vivências práticas reforçaram o conhecimento dos alunos, elas confirmaram a escolha profissional das bolsistas, o que valida a importância da iniciação à docência proporcionada pelo PIBID.

CONSIDERAÇÕES FINAIS

O projeto aponta que a construção guiada de um jogo de tabuleiro do tipo quiz no 4º ano pode favorecer a alfabetização digital e a aprendizagem de conceitos lógicos fundamentais. A passagem do planejamento no papel para a codificação no Scratch permitiu que os estudantes, sem experiência prévia com informática básica, avançassem do papel de consumidores para o de criadores de mídias interativas.

A sequência mobilizou, em diferentes momentos, os quatro pilares do Pensamento Computacional: a decomposição do jogo em etapas, o reconhecimento de padrões na replicação do código, a abstração na representação do tabuleiro por coordenadas e o raciocínio algorítmico na movimentação e nas regras. A etapa de testes fortaleceu a autonomia e a depuração de erros, e a heterogeneidade das turmas favoreceu a tutoria entre pares. A divisão das turmas mostrou-se eficaz neste contexto para lidar com as limitações do laboratório, e a regência qualificou a mediação, o planejamento e a gestão de sala das licenciandas.

Pode-se concluir que, apesar das dificuldades encontradas no sistema público de ensino, existem meios e medidas que podem ser adotados para melhorar a aplicação do ensino de informática. Um planejamento voltado ao contexto vivido e adaptado às dificuldades e ao nível da turma mostra que uma prática guiada pode ter sucesso e render ainda mais frutos quando pautada nos preceitos do Pensamento Computacional, que trabalha o raciocínio do aluno de forma sequencial. Isso traz ganhos para os alunos e experiência para quem aplica.

ISSN: 2358-8829

REFERÊNCIAS

BRACKMANN, Christian Puhlmann. Desenvolvimento do pensamento computacional através de atividades desplugadas na educação básica. 2017. Tese (Doutorado em Informática na Educação) – Universidade Federal do Rio Grande do Sul, Porto Alegre, 2017.

BRASIL. Ministério da Educação. Base Nacional Comum Curricular. Brasília: MEC, 2017. Disponível em: https://basenacionalcomum.mec.gov.br/. Acesso em: [data].

BRASIL. Conselho Nacional de Educação. Câmara de Educação Básica. Resolução CNE/CEB nº 1, de 4 de outubro de 2022. Normas sobre Computação na Educação Básica – Complemento à BNCC. Brasília: CNE, 2022.

BRENNAN, Karen; RESNICK, Mitchel. New frameworks for studying and assessing the development of computational thinking. In: ANNUAL MEETING OF THE AMERICAN EDUCATIONAL RESEARCH ASSOCIATION, 2012, Vancouver. Proceedings [...]. Vancouver: AERA, 2012. p. 1-25.

PAPERT, Seymour. A máquina das crianças: repensando a escola na era da informática. Porto Alegre: Artmed, 2008.

PELISSER, Abner J.; BORGES, Vanessa dos Anjos. Aplicação do Pensamento Computacional no ensino básico para o desenvolvimento de crianças e jovens: uma análise sistemática. In: JORNADA CIENTÍFICA E TECNOLÓGICA DA FATEC DE BOTUCATU, 9., 2020, Botucatu. Anais [...]. Botucatu: Fatec, 2020. p. 1-6.

RESNICK, Mitchel; ROSENBAUM, Eric. Designing for tinkerability. In: HONEY, Margaret; KANTER, David E. (org.). Design, make, play: growing the next generation of STEM innovators. New York: Routledge, 2013. p. 163-181.

VALENTE, José Armando. O computador na sociedade do conhecimento. Campinas: UNICAMP/NIED, 2014.

WING, Jeannette M. Computational thinking. Communications of the ACM, New York, v. 49, n. 3, p. 33-35, 2006.