

USO DO SCRATCH PARA INTRODUZIR PENSAMENTO COMPUTACIONAL NO CÁLCULO DE ÍNDICE DE MASSA CORPORAL

Kaiki A. de SOUZA¹; Sophia P. COELHO²; Fredy C. RODRIGUES³

RESUMO

Este estudo exploratório investigou como licenciandos em Matemática do IFSULDEMINAS – Campus Passos aplicam os quatro pilares do Pensamento Computacional ao desenvolver programas em Scratch para calcular o Índice de Massa Corporal (IMC). Quatro grupos tiveram seus projetos avaliados pelo Dr. Scratch, que atribuiu escores de 0 a 3 em sete dimensões: abstração, paralelismo, lógica, sincronização, fluxo de controle, interatividade e representação de dados. Os resultados mostraram proficiência avançada em lógica e representação de dados (média 3,0), níveis intermediários em fluxo de controle (1,0) e interatividade (2,0), e performance básica em abstração (0,5), paralelismo (0,25) e sincronização (0,5). Conclui-se que, apesar de dominarem algoritmos e reconhecimento de padrões, os futuros professores necessitam de intervenções pedagógicas que enfatizem a criação de blocos customizados e o uso de scripts paralelos para promover um desenvolvimento equilibrado dos quatro pilares do Pensamento Computacional.

Palavras-chave: Educação Matemática; Pensamento Computacional; Scratch; Índice de Massa Corporal; Matemática.

1. INTRODUÇÃO

Na formação de professores de Matemática, o desenvolvimento de competências que integrem pensamento lógico e domínio de ferramentas digitais é cada vez mais crucial. O Pensamento Computacional (PC) surge como uma abordagem capaz de articular essas dimensões, fornecendo um conjunto de estratégias mentais para formular, modelar e resolver problemas de forma sistemática. Fundamentado nos trabalhos de Wing (2006, 2014) e aprimorado por estudos como Brackmann et al. (2017), o PC vai além da simples programação: é uma atitude intelectual que todos os profissionais da educação devem cultivar para enfrentar a complexidade do século XXI.

O PC apoia-se em quatro pilares interdependentes. A decomposição permite fragmentar problemas amplos em subproblemas manejáveis; o reconhecimento de padrões identifica regularidades que viabilizam a reutilização de soluções; a abstração seleciona apenas as informações essenciais, ocultando detalhes supérfluos; e a formulação de algoritmos organiza passos lógicos e precisos para conduzir à resolução. Juntos, esses elementos fortalecem o raciocínio matemático, estimulam a autonomia investigativa e favorecem a construção de atividades didáticas mais dinâmicas

¹Bolsista PIBIC/CNPq, IFSULDEMINAS – Campus Passos e discente do curso de Bacharelado em Ciência da Computação. E-mail: kaiki.alvarenga@alunos.if sulde minas.edu.br

²Colaboradora voluntária e discente do curso técnico em Design Gráfico integrado ao ensino médio, IFSULDEMINAS – Campus Passos. E-mail: sophia.coelho@alunos.if sulde minas.edu.br.

³PhD Science Education - Math. Orientador do estudo. IFSULDEMINAS – Campus Passos. E-mail: fredy.rodrigues@if sulde minas.edu.br

e reflexivas Wing (2006, 2014).

Integrar o Pensamento Computacional ao Scratch torna conceitos abstratos acessíveis e práticos. Ao criar programas de cálculo de IMC, futuros professores exercitam decomposição, reconhecimento de padrões, abstração e algoritmos. Essa prática reforça o domínio de estruturas lógicas e a tradução de ideias matemáticas em blocos visuais. Além disso, prepara educadores para aplicar metodologias inovadoras que promovem raciocínio lógico e solução de problemas reais.

O objetivo geral desta pesquisa é analisar como e em que nível os alunos do curso de Licenciatura em Matemática do IFSULDEMINAS – Campus Passos aplicam os quatro pilares do Pensamento Computacional (decomposição, reconhecimento de padrões, abstração e algoritmos) ao elaborar, em Scratch, um programa para o cálculo do Índice de Massa Corporal (IMC).

2. MATERIAL E MÉTODOS

No presente estudo, 17 alunos do curso de Licenciatura em Matemática do IFSULDEMINAS – Campus Passos, já familiarizados com Scratch após um curso de oito horas, foram divididos em quatro equipes de quatro a cinco integrantes e desafiados a criar, no ambiente visual do Scratch (<https://scratch.mit.edu>), um programa⁴ em forma de animação ou jogo capaz de calcular o Índice de Massa Corporal a partir de peso e altura fornecidos pelo usuário. A atividade foi realizada em duas sessões de cinquenta minutos cada, no laboratório de informática, e, ao final de cada encontro, cada grupo exportou seu projeto no formato .sb3 e enviou o link ao pesquisador.

Para aferir o domínio das competências de Pensamento Computacional mobilizadas, todos os projetos passaram pelo Dr. Scratch (<https://www.drscratch.org>) em modo Vanilla, que atribui escores⁵ (notas) de zero a três em sete dimensões (abstração, paralelismo, lógica, sincronização, controle de fluxo, interatividade e representação de dados), indicando grau de proficiência e eventuais “maus hábitos” no uso de blocos. Em seguida, realizou-se análise qualitativa dos códigos para verificar como os pilares de decomposição, reconhecimento de padrões, abstração e algoritmos se materializam na solução do IMC. Ao combinar os escores quantitativos com a interpretação dos relatórios, foi possível não apenas medir o nível de cada dimensão, mas também compreender como

⁴ Scratch é um ambiente de programação visual desenvolvido pelo MIT Media Lab que permite criar animações, jogos e histórias interativas por meio de blocos de código conectáveis.

⁵ Escore 0 – Ausência (Não há qualquer evidência do uso do construto avaliado. Por exemplo, em Abstração, não existem blocos customizados; em Paralelismo, não há scripts paralelos; em Lógica, faltam testes condicionais, etc. Escore 1 – Presença incipiente. Há menções pontuais ao conceito, mas de forma pouco consistente. Exemplo: um único bloco customizado criado mas sem parâmetros (Abstração), ou um “broadcast” simples sem coordenação (Sincronização). Escore 2 – Aplicação consistente (nível intermediário). O recurso já é usado de modo correto e recorrente, porém sem alto grau de complexidade. Por exemplo, vários blocos customizados sem invocação de parâmetros complexos; condicionais aninhados, mas com poucos laços; algumas transmissões coordenadas, mas não em múltiplas threads. Escore 3 – Aplicação robusta e sofisticada. Demonstra domínio pleno: blocos customizados parametrizados e reutilizados em diferentes scripts; estruturas de controle (laços e condicionais) bem aninhadas; paralelismo organizado com transmissões “broadcast & wait” em vários níveis; interatividade avançada com validações e feedback dinâmico; uso extensivo de listas ou estruturas de dados (Fonte: [Dr. Scratch](#)).

esses saberes computacionais emergem na prática formativa dos futuros professores, dando suporte à questão: “Como e em que nível esses alunos aplicam os quatro pilares do Pensamento Computacional ao elaborar um programa em Scratch para calcular o IMC?”

3. RESULTADOS E DISCUSSÃO

Foram analisados 4 projetos no modo Vanilla do Dr. Scratch, gerando escores de 0 a 3 em cada uma das sete dimensões centrais. Com base na metodologia descrita e utilizando o Dr. Scratch foi analisado os seguintes códigos: <https://scratch.mit.edu/projects/1096207526> (**Grupo 1**); <https://scratch.mit.edu/projects/1095560256> (**Grupo 2**); <https://scratch.mit.edu/projects/218455850> (**Grupo 3**); <https://scratch.mit.edu/projects/1095557241> (**Grupo 4**).

A Tabela 1 apresenta os scores de pontuação obtidos na avaliação dos códigos.

Tabela 1 – Escores (0–3) por dimensão de Pensamento Computacional

Grupo	Abstração	Paralelismo	Lógica	Sincronização	Fluxo de Controle	Interatividade	Representação de Dados	Total
G1	0	0	3	0	1	2	3	9/21
G2	1	1	3	1	1	2	3	12/21
G3	0	0	3	1	2	2	3	11/21
G4	1	0	3	0	1	2	3	10/21

Os escores atribuídos pelo Dr. Scratch (0 = ausência, 1 = presença incipiente, 2 = aplicação consistente, 3 = aplicação robusta) permitiram mapear o desempenho dos quatro grupos em cada dimensão do Pensamento Computacional. A média de Abstração foi 0,5, situada entre ausência e presença incipiente, enquanto Paralelismo (0,25) e Sincronização (0,5) também se mantiveram em patamares básicos. Em contraponto, Representação de Dados recebeu média máxima (3,0), sinalizando aplicação robusta. No âmbito Algorítmico, Lógica atingiu 3,0, Fluxo de Controle obteve exatamente 1,25 (limiar entre presença incipiente e aplicação consistente) e Interatividade alcançou 2,0, indicando uso recorrente do recurso.

Ao analisar a Decomposição, nota-se que apenas o Grupo 2 obteve escore 1 em Paralelismo e Sincronização, evidenciando presença incipiente de scripts paralelos e “broadcast & wait”. Os Grupos 1, 3 e 4, com escores 0 nessas dimensões, revelaram ausência de fragmentação, centralizando todo o fluxo de cálculo em um único script. Esse padrão corrobora Wing (2014), para quem a divisão de um problema em subtarefas independentes ainda não foi assimilada pela maioria dos alunos.

No que tange ao Reconhecimento de Padrões, o escore 3 em Representação de Dados demonstra aplicação robusta de listas e variáveis para generalizar faixas de IMC e evitar repetições. Essa uniformidade entre os quatro grupos reflete a capacidade de modularizar padrões recorrentes, conforme defendido por Brackmann et al. (2017).

A Abstração, mensurada pela existência de blocos customizados, apresentou média de 0,5. Grupos 2 e 4 alcançaram escore 1, indicando presença incipiente de blocos sem parametrização

profunda, enquanto Grupos 1 e 3 obtiveram 0, ou seja, ausência total de unidades reutilizáveis. Segundo Wing (2006), essa limitação reduz a clareza e a escalabilidade dos projetos.

Por fim, o eixo Algoritmos revelou domínio na construção lógica (escore 3 em Lógica) e na integração de eventos interativos (escore 2 em Interatividade), com Fluxo de Controle pontuado em 1,0, mostrando uso simples e pontual de condicionais e laços. A combinação dessas medidas gera média avançada (2,0) para o pilar Algoritmos, confirmando a habilidade dos licenciandos em formular procedimentos corretos para o cálculo do IMC.

Em resposta à questão de pesquisa, conclui-se que os alunos aplicam os pilares Algoritmos e Reconhecimento de Padrões em níveis avançados e intermediários, mas permanecem em patamares básico a incipiente nos pilares Abstração e Decomposição. Tal desequilíbrio revela a necessidade de práticas pedagógicas que promovam de forma sistemática a criação de blocos customizados e o emprego de scripts paralelos, de modo a fortalecer o desenvolvimento equilibrado dos quatro pilares do Pensamento Computacional.

4. CONCLUSÃO

Os alunos demonstraram domínio avançado em Algoritmos e Reconhecimento de Padrões, mas apresentam proficiência básica em Abstração e Decomposição. Recomenda-se, portanto, atividades que enfatizem a criação de blocos customizados e o uso de scripts paralelos para equilibrar o desenvolvimento dos quatro pilares do Pensamento Computacional.

AGRADECIMENTOS

Agradecemos ao PIBIC do IFSULDEMINAS, Campus Passos pelo financiamento do estudo.

REFERÊNCIAS

BRACKMANN, C. C.; ANJANE, J. L.; FERNANDES, E.; LIMA, R. Pensamento computacional: definição e diretrizes educacionais. Revista Brasileira de Informática na Educação, Rio de Janeiro, v. 25, n. 2, p. 226–260, 2017.

MIT MEDIA LAB. Scratch. Disponível em: <https://scratch.mit.edu>. Acesso em: 20 jul. 2025.

SCRATCH TEAM. Dr. Scratch: ferramenta de avaliação de projetos Scratch. Disponível em: <https://www.drscratch.org>. Acesso em: 20 jul. 2025.

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

WING, J. M. Computational thinking – What and why? International Journal of Education and Development using ICT, v. 10, n. 2, p. 43–54, 2014.