


Pensamento computacional em movimento: narrativas digitais na construção das relações funcionais com software Scratch¹

Computational thinking in motion: digital narratives in the construction of functional relationships with scratch software

Fabiano Ferraz Trancoso ^[a] 

Ilhéus, BA, Brasil

Secretaria da Educação do Estado da Bahia

Flaviana dos Santos Silva ^[b] 

Ilhéus, BA, Brasil

Universidade Estadual de Santa Cruz (UESC), Departamento de Ciências Exatas (DCEX)

Jurema Lindote Botelho Peixoto ^[c] 

Ilhéus, BA, Brasil

Universidade Estadual de Santa Cruz (UESC)

Como citar: TRANCOSO, F. Ferraz; SANTOS SILVA, Flaviana dos; PEIXOTO, Jurema L. Botelho. Pensamento computacional em movimento. *Revista Diálogo Educacional*, Curitiba: Editora PUCPRESS, v. 23, n. 77, p. 746-759, abr./jun. 2023. DOI: doi.org/10.7213/1981-416X.23.077.DS06

Resumo

Este artigo pretende apresentar um recorte dos resultados de uma pesquisa de mestrado que teve como objetivo investigar as possíveis implicações do pensamento computacional no estudo das relações funcionais para promover a construção do conceito de função afim por meio do software Scratch. A intervenção de ensino ocorreu com estudantes

¹ Artigo é um recorte da dissertação de Mestrado. TRANCOSO, F. F. *Implicações do pensamento computacional no desenvolvimento das relações funcionais com o software scratch: o caso da função afim*. 2019. 159p. Dissertação (Mestrado em Educação Matemática) – Universidade Estadual de Santa Cruz, Ilhéus, 2019.

^[a] Mestre em Educação Matemática, e-mail: fabianoft@yahoo.com.br

^[b] Doutora em Educação, e-mail: fssilva@uesc.br

^[c] Doutora em Difusão do Conhecimento, e-mail: jurema@uesc.br

do 1º ano do ensino médio do curso técnico em Administração de uma escola no município de Ilhéus/BA. Para tanto, adotou-se uma abordagem qualitativa de caráter quase experimental na qual se considerou os aportes teóricos do pensamento computacional e a utilização de metodologias ativas. Os resultados mostraram que a utilização do software Scratch a partir da produção de narrativas digitais contribuiu para o desenvolvimento do pensamento computacional, a compreensão do conceito de variável, a relação de dependência entre as variáveis e a lei de formação da função afim. Além disso, o uso do software trouxe motivação para envolver os estudantes no ensino matemática com autonomia.

Palavras-chave: Pensamento Computacional. Metodologias Ativas. Narrativas Digitais. Relações Funcionais.

Abstract

This article intends to present an extract of the results of a master's research that aimed to investigate the possible implications of computational thinking in the study of functional relations to promote the construction of the concept of affine function through the Scratch software. The teaching intervention took place with students in the 1st year of high school in the technical course in Administration at a school in the municipality of Ilhéus/BA. Therefore, a qualitative approach of an almost experimental character was adopted, in which the theoretical contributions of Computational Thinking and the use of Active Methodologies were considered. The results showed that the use of the Scratch software from the production of digital narratives contributed to the development of computational thinking, the understanding of the variable concept, the dependency relationship between the variables and the formation law of the affine function. In addition, the use of the software brought motivation to involve students in teaching mathematics with autonomy.

Keywords: Computational Thinking. Active Methodologies. Digital Narratives. Functional Relationships.

Resumen

Este artículo tiene como objetivo presentar un extracto de los resultados de una investigación de maestría que tuvo como objetivo investigar las posibles implicaciones del pensamiento computacional en el estudio de las relaciones funcionales para promover la construcción del concepto de función afín a través del software Scratch. La intervención pedagógica se realizó con alumnos del 1º año de la enseñanza media del curso técnico en Administración de una escuela del municipio de Ilhéus. Por ello, se adoptó un enfoque cualitativo de carácter casi experimental, en el que se consideraron los aportes teóricos del Pensamiento Computacional y el uso de Metodologías Activas. Los resultados mostraron que el uso del software Scratch a partir de la producción de narrativas digitales contribuyó al desarrollo del pensamiento computacional, la comprensión del concepto de variable, la relación de dependencia entre las variables y la ley de formación de la función afín. Además, el uso del software trajo motivación para involucrar a los estudiantes en la enseñanza de las matemáticas con autonomía.

Palabras clave: Pensamiento Computacional. Metodologías Activas. Narrativas digitales. Relaciones Funcionales.

Introdução

O estudo de funções faz parte do dia a dia e está presente na realização das atividades mais elementares, como as de relações de consumo, de relações financeiras e em muitas outras situações em que se precisa relacionar duas ou mais grandezas.

Isso confere ao estudo de funções o caráter de aplicabilidade tão questionado pelos alunos: “Afinal de contas, onde eu vou usar isso?”. Qual o professor de Matemática nunca ouviu essa pergunta? Ela normalmente é feita porque os alunos não conseguem reconhecer uma conexão entre o que é estudado na escola e as situações cotidianas que podem ser modeladas por funções a que todos nós somos expostos diariamente. Tais dificuldades podem ser decorrentes das práticas utilizadas na sala de aula que na maioria das vezes seguem tradicionalmente a ordenação dos conteúdos sugerida pelos livros didáticos. Esta ordenação costuma tratar os conteúdos de forma isolada, trazendo poucas situações que fazem referência às suas aplicações. Isso reduz os alunos a meros repetidores que não conseguem compreender o que estão fazendo nem por que estão fazendo.

Ao contrário dessa postura, os Parâmetros Curriculares Nacionais para o Ensino Médio (PCN+) (BRASIL, 2002) sugerem que o ensino de funções pode ser iniciado a partir de situações contextualizadas. Sendo assim, os “problemas de aplicação devem ser motivo e contextos para o estudante aprender funções. A riqueza de situações envolvendo funções permite que seu ensino se estruture permeado de exemplos do cotidiano” (BRASIL, 2002, p. 118).

Desse modo, pode-se ir além do livro didático, colocando o aluno em uma posição mais ativa na construção de seu conhecimento, possibilitando a experimentação e estimulando a criatividade. Tais situações podem ser mediadas por ambientes digitais que serviriam de suporte no processo educacional, como o uso de planilhas eletrônicas, softwares de geometria dinâmica e por meio de softwares de programação.

Nesse interim, a utilização do Scratch como ferramenta usada para auxiliar os professores no ensino de diferentes componentes curriculares, sobretudo no ensino da matemática, pode se transformar em uma oportunidade de melhorar as relações de ensino e aprendizagem. Torna-se uma alternativa para trabalhar com a resolução de situações-problema, permitindo que os alunos construam seus conhecimentos a partir de contextos oriundos de sua própria realidade.

Sendo assim, as implicações do pensamento computacional estão diretamente articuladas ao estudo das relações funcionais. Isso porque o pensamento computacional se utiliza de princípios da ciência da computação, tais como coleta e análise de dados, a forma que esses dados são representados, a decomposição de problemas em problemas menores, o uso de algoritmos, principalmente o raciocínio lógico, entre outros.

As situações-problema poderão ser resolvidas através de metodologias ativas, que é um processo amplo cuja característica fundamental é a inserção do estudante como principal responsável pela sua aprendizagem, pois o coloca em um papel de protagonista dessa aprendizagem.

A programação permite ao estudante investigar e refletir sobre suas investigações, o que supõe que a ideia de função seja compreendida a partir da “associação aos estudos das expressões analíticas” que vão surgindo na elaboração e estruturação de algoritmos a partir dos comandos no software (PONTE, 1990, p. 4). Desse modo, o estudante se transforma em protagonista na construção de seu conhecimento, pois precisa enfrentar situações desafiadoras e tentar vencer possíveis dificuldades.

Diante do exposto, este artigo mostra os resultados de uma pesquisa de mestrado realizada em 2018, em uma escola do sul da Bahia, que teve como objetivo geral investigar as possíveis implicações do pensamento computacional no estudo das relações funcionais, em uma intervenção de ensino, para promover a construção do

conceito de função afim por meio do software Scratch¹ pelos estudantes do 1º ano do ensino médio. Para o qual foram definidos os seguintes objetivos específicos:

- Desenvolver o conceito de função afim através de atividades de intervenção nas aulas utilizando o software de programação Scratch;
- Determinar a lei de formação de uma função afim a partir de dados presentes numa situação-problema;
- Verificar as possíveis contribuições do software de programação Scratch no estudo da função afim em uma intervenção de ensino;
- Verificar o efeito do pensamento computacional no desenvolvimento dos conceitos de função afim.

Diante dos objetivos, no desenvolvimento da investigação, foi levantada como hipótese que o estudo da função poderia ser explorado a partir de um tema de interesse dos estudantes, que facilitaria o emprego das metodologias ativas com o uso do *software Scratch*.

Cabe salientar que o recorte deste artigo ficará restrito à produção das narrativas digitais para aprendizagem do conceito de função afim. Do ponto de vista pedagógico esse trabalho traz contribuições relevantes uma vez que aponta um caminho para possíveis mudanças no modelo educacional que ainda se baseia na transmissão de informações e que concebe o aluno como um ser passivo, incapaz de agir de maneira crítica e reflexiva. O que vem ao encontro do que foi explicitado até aqui acerca da utilização de metodologias ativas na resolução de problemas com *softwares* como o *Scratch*.

Aporte Teórico

A Base Nacional Comum Curricular (BNCC) (BRASIL, 2018) enfatiza a importância do desenvolvimento do Pensamento Computacional para a capacitação do estudante no que se refere ao domínio de diferentes linguagens, entre elas a algébrica, e na importância do algoritmo. A linguagem algorítmica tem pontos em comuns com a linguagem algébrica, sobretudo em relação ao conceito de variável.

Para o Ensino Médio, a BNCC (BRASIL, 2018) na área de matemática e suas tecnologias, propõe que as aprendizagens essenciais desenvolvidas no Ensino Fundamental sejam retomadas de forma mais aprofundada para que haja uma consolidação e ampliação desses conhecimentos. Para isso, propõe que esses conhecimentos sejam retomados de forma mais inter-relacionada para que os estudantes construam uma visão mais integrada da matemática numa perspectiva de aplicabilidade com a realidade em diferentes contextos.

Dessa forma, deve-se levar em conta as vivências cotidianas dos estudantes do Ensino Médio que são impactados de diferentes formas pelos avanços tecnológicos, pelas exigências do mercado de trabalho, pelas potencialidades das mídias sociais, entre outros. Sendo assim, para o Ensino Médio, a BNCC destaca a “importância do recurso a tecnologias digitais e aplicativos tanto para a investigação matemática como para dar continuidade ao desenvolvimento do pensamento computacional iniciado na etapa anterior” (BRASIL, 2018, p. 528).

Assim, a BNCC (2018) pressupõe que seja desenvolvido no estudante o comportamento protagonista no processo de construção de seu conhecimento de forma que competências como raciocinar, representar, comunicar e argumentar sejam desenvolvidas. Tal determinação vem ao encontro do trabalho com softwares, como o Scratch, que permite que essas competências sejam mobilizadas.

¹ *Scratch* é uma linguagem de programação criada no MIT (Massachusetts Institute of Technology). É inspirada em linguagens de programação para jovens como LOGO e Squeak Etoys (RESNICK, 2007). O objetivo primário do *Scratch* é ajudar as crianças (já a partir dos 8 anos de idade) a desenvolver competências essenciais de aprendizagem do século XXI (RUSK; RESNICK; MALONEY, 2006).

De acordo com Silva, Almeida e Silva (2017), Teixeira (2017), Tucker *et al.* (2003), Valente (2016) e Wing (2006), apenas operar o computador (saber navegar na internet, enviar um e-mail, publicar em um blog ou utilizar um processador de texto) não é mais suficiente para as competências exigidas pela sociedade atual. Por esse motivo, o “conceito de pensamento computacional é importante para a nossa geração e cada vez mais importante para as gerações futuras” (TEIXEIRA, 2017, p. 31).

Nesse sentido, é possível encontrar diversos trabalhos que tratam o desenvolvimento do pensamento computacional (PC) não apenas voltados para a Ciência da Computação mas também para educação, principalmente para a Educação Matemática, utilizando os chamados softwares de autoria (VALENTE, 2016; TEIXEIRA, 2017; FRANÇA; AMARAL, 1999; PINTO, 2010). Desse modo, o foco aqui será algumas características e definições acerca das ideias de PC e de como essas ideias podem ser desenvolvidas utilizando um software de autoria.

O termo “pensamento computacional” (WING, 2006) foi apontado e amplamente divulgado no ano de 2006 em um artigo denominado “Computational Thinking” (WING, 2006), da pesquisadora e professora de Ciências da Computação da Universidade Carnegie Mellon, Jeannette M. Wing. Apesar de esse termo ter causado grande repercussão depois do artigo “Computational Thinking” (WING, 2006), ele é mais antigo e foi utilizado por Papert (1980) em seu livro intitulado “Mindstorms: children computers and powerful ideas”. Na edição brasileira, intitulada “Logo: Computadores e Educação”, publicada em 1985, o termo “pensamento computacional” também é encontrado (PAPERT, 1985, p. 217). Nessa obra, Papert aborda de forma pioneira o papel da tecnologia no ensino de crianças, mas a expressão “pensamento computacional” não obteve grande mobilização na época nem seus princípios foram difundidos.

Para Blikstein (2008), o PC é saber usar o computador como um instrumento de aumento do poder cognitivo e operacional humano. Ou seja, é utilizar o computador para aumentar a produtividade, inventividade e criatividade.

Por sua vez, Wing (2006) afirma que o pensamento computacional é associado às ideias de resolução de problemas, design de sistemas e compreensão do comportamento humano norteado por conceitos fundamentais da Ciência da Computação.

Segundo Brackmann (2017), o PC apoia-se em quatro pilares. O primeiro pilar, chamado de “decomposição”, é o processo pelo qual problemas complexos são quebrados em partes mais simples. As partes menores podem ser analisadas e resolvidas ou concebidas individualmente por serem mais fáceis de trabalhar. O segundo, “reconhecimento de padrões”, envolve a identificação de similaridades entre as partes decompostas, isto é, entre os subproblemas gerados. A solução encontrada para resolver problemas mais simples pode ser replicada e ajudar a resolver outros mais complexos. O reconhecimento de padrões também pode ser denominado pelo termo “generalizações”.

O terceiro pilar, denominado “abstração”, envolve o processo de análise dos elementos. Trata-se da filtragem e classificação de dados de forma a ignorar os elementos desnecessários para concentrar-se naquilo que é relevante. A partir do pensamento abstrato é possível criar uma representação do que se pretende resolver. Wing (2006) afirma ser esse o conceito mais importante do PC, pois o processo de abstrair é utilizado em diversos momentos, como, por exemplo, na escrita de algoritmos, na seleção dos dados importantes, na escrita de uma pergunta entre outros.

Por seu turno, tem-se o quarto pilar, “algoritmos”, considerado por Wing (2014) como o elemento agregador de todos os demais. O algoritmo é um plano detalhado, um conjunto de instruções, de etapas (passos) para resolver um problema. De forma bastante simples, pode-se dizer que um algoritmo é um conjunto de instruções para se resolver um problema.

Em Valente (2016), o pensamento computacional é visto como um processo de resolução de problemas por meio de uma linguagem de programação, que acontece por intermédio de um ciclo de ações “descrição-execução-reflexão-depuração”, que são úteis para explicitar a interação do aprendiz com as tecnologias digitais, que o ajuda a entender como essa interação contribui para o desenvolvimento do pensamento computacional.

Nesse sentido, “destaca-se como um dos desafios à educação o repensar sobre novas propostas educativas que superem a instrução ditada pelo livro didático, centrada no dizer do professor e na passividade do aluno” (VALENTE; ALMEIDA; GERALDINI, 2017, p. 458).

Uma das propostas para essa mudança é a adoção de metodologias ativas voltadas para promover a aprendizagem que são:

Estratégias pedagógicas para criar oportunidades de ensino nas quais os alunos passam a ter um comportamento mais ativo, envolvendo-os de modo que eles sejam mais engajados, realizando atividades que possam auxiliar o estabelecimento de relações com o contexto, o desenvolvimento de estratégias cognitivas e o processo de construção de conhecimento (VALENTE; ALMEIDA; GERALDINI, 2017, p. 464).

Entra em cena a ação do professor no desenvolvimento de atividades que possibilitem o protagonismo do estudante. A sua orientação é primordial na atribuição dos significados das informações encontradas pelos estudantes, assim o professor precisa priorizar as atividades em que os “estudantes possam questionar, levantar hipóteses, investigar, desenvolver a autonomia, o senso crítico, a ética e a estética; despertando a curiosidade por meio de desafios e oportunidades para aguçar o pensamento” (ALMEIDA, 2016, p. 530-531).

As Metodologias Ativas podem ser abordadas por meio de diferentes estratégias. As formas mais tradicionais de sua implantação são: a aprendizagem baseada em projetos, a aprendizagem por meio de jogos, o método do caso ou discussão, solução dos casos e a aprendizagem em equipe.

Nesse contexto, as narrativas digitais relacionam-se com a utilização das Tecnologias Digitais de Informação e Comunicação (TDIC) e das metodologias ativas para produzir registros que antes eram orais ou impressas (VALENTE, 2016), o que foi um objeto importante no desenvolvimento da pesquisa.

Nesse sentido, articular a proposta de estudar as relações funcionais utilizando um *software* de autoria como o *Scratch* vem ao encontro da proposta de metodologia ativa e das ideias de pensamento computacional. Isso porque ambas colocam o aluno em uma postura de autor (construtor) de seu conhecimento de forma criativa.

É possível perceber que as funções podem ser representadas de diferentes maneiras, tais como por uma equação, por meio de uma tabela, de gráficos, por regras verbais, regras matemáticas e modelos. As representações são resultantes da existência de padrões. Matos e Ponte (2008, p. 200) afirmam que “o estudo de funções inclui a necessidade de compreender como as variáveis se relacionam, isto é, de conseguir explicitar uma relação funcional a partir de um certo conjunto de dados. Este processo baseia-se, essencialmente, na exploração de padrões e regularidades”.

Teixeira (2016), ao pesquisar sobre a introdução do raciocínio funcional em estudantes do 5º ano do ensino fundamental, utiliza em seu trabalho os estudos da *Early Algebra*, no qual um dos focos é descrever relações funcionais a partir de generalizações de padrões numéricos ou geométricos.

Diante do exposto, assume-se neste trabalho como relação funcional toda relação que define o conceito de função. Ou seja, em uma relação funcional, cada elemento de um conjunto A está relacionado a um único elemento do conjunto B. Tais relações podem, como já foi dito anteriormente, ser verificadas em situações em que é possível identificar padrões e regularidades. Tais padrões podem ser representados por gráficos, tabelas, situações de proporcionalidades, progressões, entre outras.

Na próxima seção serão apresentados os procedimentos metodológicos para o desenvolvimento da pesquisa realizada.

Desenvolvimento

A pesquisa foi realizada durante o ano de 2018 em uma turma do 1º ano do curso técnico em Administração, em uma instituição pública de autarquia estadual de ensino regular no município de Ilhéus/BA contando com a participação de aproximadamente 25 estudantes, sem a utilização de um grupo de controle. O Protocolo de pesquisa foi aprovado conforme parecer de número: 2.596.412 do CEP-UESC; CAAE: 85814218.9.0000.5526.

Os dados apresentados nesse artigo se referem ao recorte e foram obtidos a partir das narrativas digitais produzidas pelos alunos no *software Scratch* em uma intervenção de ensino. Para tal, foi utilizado um *software* de captura de tela e gravação das ações realizadas pelos alunos na tela do computador.

A intervenção foi realizada em duas fases, sendo a primeira para a apresentação e exploração do *software Scratch*, e a segunda para a aplicação das atividades, conforme apresentado no quadro 1.

Quadro 1 – Fases da Intervenção

Fases	Encontros	Aulas	Procedimentos
1ª	1º	3	Exibição do filme Code Stars ² , Exploração do software Scratch utilizando o tutorial ³ por meio de uma programação orientada.
	2º	3	Programação orientada, construção do “Jogo das Maçãs”, criação de histórias por parte dos estudantes.
2ª	3º	3	Atividade 1 – Noção intuitiva de função (domínio e imagem); Programação utilizando o software Scratch
	4º	3	Atividade 2 – Noção intuitiva de função inversa (domínio e imagem); Programação utilizando o software Scratch
	5º	3	Atividade 3 – Noção intuitiva de função definida por mais de uma sentença. Programação utilizando o software Scratch

Fonte: Adaptado de Trancoso (2019).

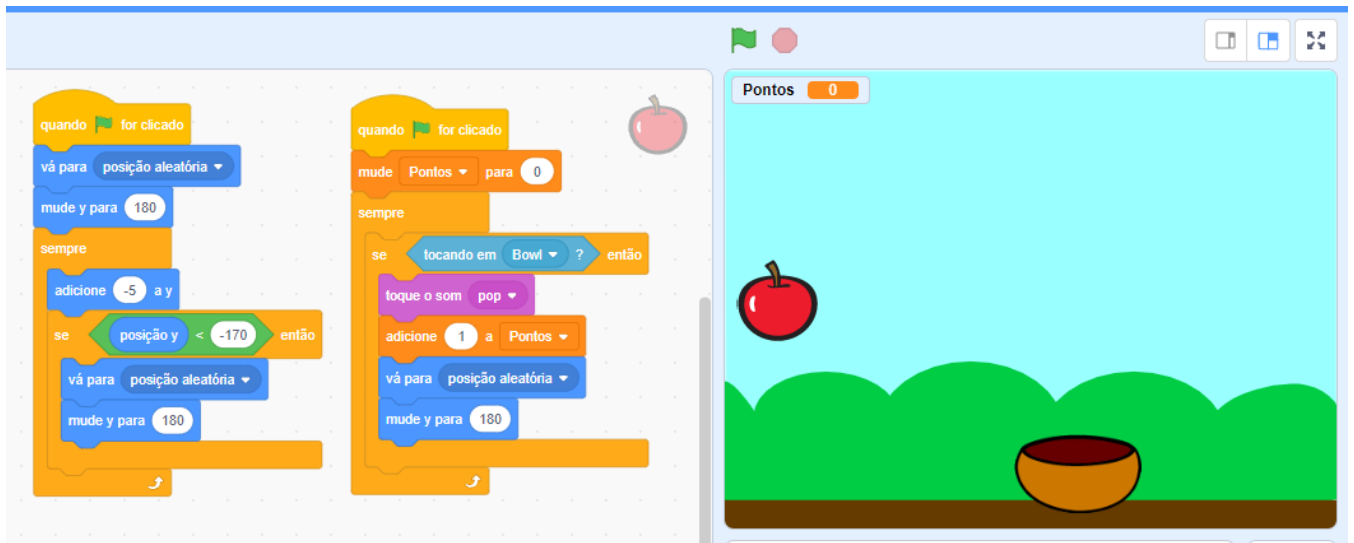
Na primeira fase, os estudantes foram levados a explorar o *Scratch*, suas funções, seus comandos, os personagens e cenários. Para isso, procurou explicar, de forma sucinta, o que é programar (“dizer”, “explicar” ou “ensinar” a um computador o que ele deve fazer) e a importância da programação e do pensamento computacional não apenas para os profissionais de programação mas também para as diferentes áreas da sociedade. Para corroborar essas ideias, também foi exibido um pequeno vídeo, “Code Stars”, que tratava desse tema e cuja finalidade era sensibilizar os estudantes, despertando o interesse para a programação de computadores.

Para apresentar o *software Scratch* foi utilizado o método da “programação orientada”, com o recurso do *datashow* e do computador para realizar a programação do jogo das maçãs, apresentando as funções dos comandos envolvidos, conforme ilustrado na figura 1.

² Filme publicado pela Code.org, no YouTube, em 26 de fevereiro de 2013. Code.org é uma organização sem fins lucrativos dedicada a expandir o acesso à ciência da computação nas escolas. Disponível em <https://www.youtube.com/watch?v=dU1xS07N-FA&t=43s>. Acesso em: 6 jun. 2019.

³ Tutorial Adaptado do site www.scratchbrasil.net.br. Acesso em: 01 ago. de 2018.

Figura 1 – Jogo das maçãs



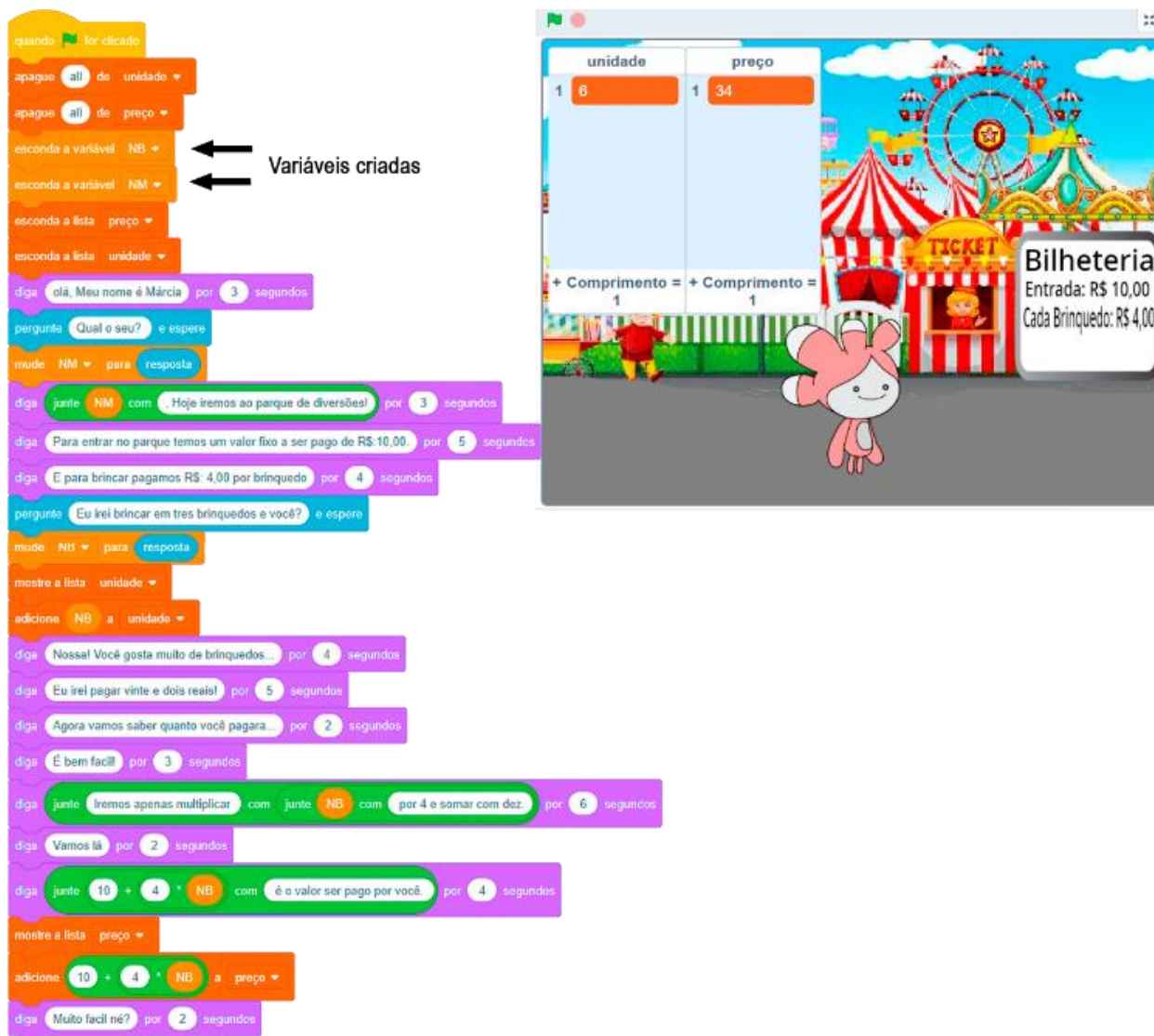
Fonte: Trancoso (2019).

Esse tipo de programação foi importante para capacitar os estudantes na realização das atividades previstas na segunda fase e que tinham por objetivo promover a construção do conceito de função afim.

Na segunda fase, os estudantes foram desafiados a assumir a autoria de programações, envolvendo a noção intuitiva de função em situações modeladas por uma função afim. Para isso, foram desenvolvidas três atividades diferentes que buscavam construir os conceitos matemáticos de função por meio das programações produzidas pelos estudantes, a saber: a noção intuitiva de função, de função inversa e de função definida por mais de uma sentença.

Na figura 2, a seguir, é apresentada um exemplo de narrativa digital produzida por um estudante com o cenário e a programação ao resolver a segunda parte da Atividade 1.

Figura 2 – Narrativa Digital do Estudante 1



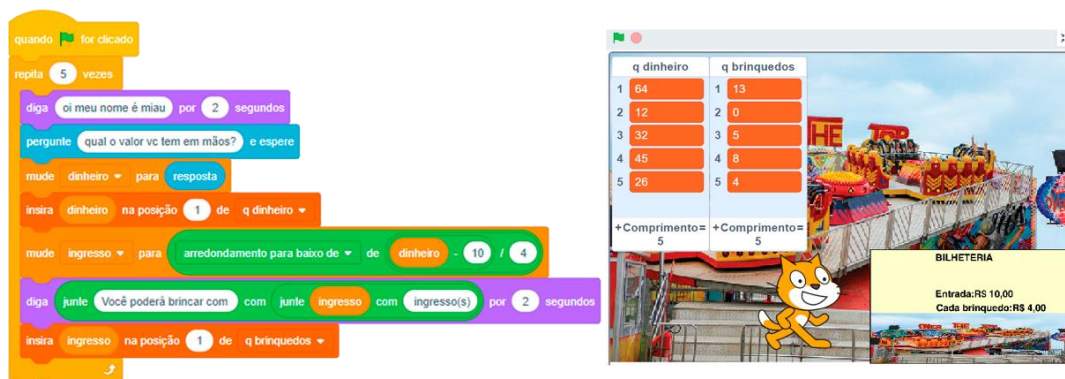
Fonte: Trancoso (2019).

Conforme mostrado na figura 2, para responder o valor a ser pago pelo número de brinquedos utilizados foram criadas as variáveis *NB* e *NM*, que representam, respectivamente, número de brinquedos e o nome do usuário. A programação produz um diálogo entre o usuário do computador e um personagem que se apresenta e pergunta quantos ingressos serão comprados. A partir dessa informação, o personagem realiza o cálculo e responde o valor que deverá ser pago em função do número de ingressos comprados. A relação entre número de ingressos comprados e valor a ser pago é apresentada em uma lista criada no cenário da sua história.

É interessante notar que a narrativa elaborada pelo estudante extrapolou o objetivo da tarefa, que era simplesmente fazer o personagem determinar o valor a ser pago em função do número de brinquedos utilizados. Em seu enredo, o personagem não só calcula o valor a ser pago, como também explica como isso é feito. Apesar de não ter explicitado a variável *y*, que representa o preço a pagar, ela escreve a expressão que representa essa variável “ $10+4 \cdot NB$ ”.

A figura 3 apresenta outra narrativa digital com o cenário e a programação utilizados.

Figura 3 – Narrativa Digital do Estudante 2



Fonte: Trancoso (2019).

Na figura 3, nota-se à direita o cenário com o personagem que dialoga com o usuário, duas tabelas onde são inseridos os valores que representam a quantia que se deseja gastar e a quantidade de brinquedos (ingressos) possíveis de se adquirir com a quantia em dinheiro informada pelo usuário. Essa figura também apresenta, de forma sucinta, os passos percorridos pelo estudante ao realizar sua programação.

Utilizando o comando “criar uma variável” o estudante criou as variáveis *dinheiro* e *ingresso* que representam, respectivamente as variáveis *x* e *y*, no contexto de função, e que são utilizadas corretamente em sua programação, tendo em vista que ele articulou de forma lógica o comando “pergunte ... e espere a resposta”, responsável pela entrada de dados, no computador, pelo usuário. O estudante fez a associação correta do comando “resposta” às variáveis criadas por meio dos comandos “mude dinheiro para resposta” e “mude ingresso para ((dinheiro – 10) ÷ 4)”.

Para construir as tabelas “q dinheiro” e “q brinquedos” que representam, no contexto matemático, os valores do domínio e da imagem da função, ele utilizou o comando “criar listas”. A programação realizada pelo estudante atribui para a lista (tabela) “q dinheiro” os valores digitados pelo usuário por meio do comando “insira dinheiro a q dinheiro”, domínio da função; e atribui para a lista “q brinquedos” os valores transferidos do comando “insira ingresso a q brinquedos”; valores estes obtidos pela expressão $((\text{dinheiro} - 10) \div 4)$, resultantes do comando “mude ingresso para $((\text{dinheiro} - 10) \div 4)$ ”. Além do comando de recursividade “repita 5 vezes”, que gerou, nessa programação, os cinco valores que foram inseridos na tabela.

Cabe ressaltar que durante a produção dessa narrativa foi necessário explicar para os estudantes acerca de alguns resultados obtidos com números decimais, por exemplo, a variável *b* cuja resposta era de 13,5, referente a $((64-10) \div 4)$. Como essa resposta representa o número de brinquedos (ingressos) possíveis de se adquirir, todos concordaram que no caso aplicado ela era absurda. Esse momento foi oportuno para se discutir o domínio dessa função que é o dos números naturais.

Houve também a descoberta de outros comandos como “operadores”, “raiz quadrada de”, “arredondar para baixo”, “mude ingresso para” para facilitar o arredondamento de valores na tabela “q brinquedos” com as respostas pertencendo ao conjunto dos números naturais e os mesmos valores em “q dinheiro”.

Em outra narrativa (Figura 4) foi utilizado o comando “criar variáveis”, as variáveis *Qt de bilhetes* e *nome* para representar a variável *x*, domínio da função afim e o nome do usuário que será inserido no diálogo. Percebe-se a articulação lógica entre os comandos “pergunte e espere a resposta” que abrirá a janela de entrada de dados pelo usuário. Através do comando “criar listas”, o estudante criou a tabela “preço a pagar”, presente no cenário e que mostrará o valor a ser pago pelo número de bilhetes comprados. A figura 4 apresenta a narrativa digital criada.

Figura 4 – Narrativa Digital do Estudante 3



Fonte: Trancoso (2019).

Nesta narrativa não foi criada uma tabela que representa a quantidade de bilhetes, como aconteceu com as anteriores, mas a alternativa adotada pelo estudante foi o comando de ativar a visualização da variável *Qt de bilhetes*. Desse modo, uma caixa (letreiro) foi inserida no cenário. Além disso, foi utilizada a estrutura de condição composta do comando “se... então... senão”, suprimindo a necessidade de criar a variável que representa o preço a pagar, o que facilitou generalizar corretamente ao criar a estrutura que representa a lei de formação da função quando a quantidade de bilhetes é maior que 10, mostrando a apropriação do conceito pelo estudante.

Um maior detalhamento será realizado na próxima seção, na qual será apresentada a discussão dos resultados.

Discussão dos resultados

Ao analisar o conteúdo de narrativas contidas nos registros digitais, é possível que o professor compreenda como o estudante desenvolve seu conhecimento para poder orientá-lo em processos de depuração de aspectos que necessitam ser reformulados, para se alcançar um estágio mais alto de compreensão do conhecimento (ALMEIDA; VALENTE, 2012). Nesse contexto, a linguagem de programação do *Scratch* permitiu desenvolver narrativas digitais que puderam ser exploradas pelos participantes.

O ensino e a aprendizagem com a abordagem das metodologias ativas, os registros dos estudantes no meio digital assumiram um valor importante para esta pesquisa, ou seja, levou em consideração “a construção e desconstrução das experiências vivenciadas” no decorrer da intervenção de ensino (ALMEIDA, 2016, p. 533).

As atividades, elaboradas pelos estudantes, são denominadas “histórias digitais, narrativas digitais, narrativas interativas, [...], ou digital *storylling*” (VALENTE, 2016, p. 876). E, nesse contexto, as narrativas produzidas pelos estudantes evidenciaram a dinâmica do ciclo de ações descrição-execução-reflexão-depuração, possibilitando a espiral de aprendizagem (VALENTE, 2018).

Isso revela que os estudantes se apropriam das ideias discutidas nas atividades propostas e utilizam essas ideias para desenvolver outros projetos escolares em outras disciplinas. Essa atitude revela uma aprendizagem baseada na investigação, o que caracteriza uma metodologia ativa (VALENTE; ALMEIDA; GERALDINI, 2017).

No decorrer da produção das narrativas ficou evidente o papel protagonista dos sujeitos na construção do próprio conhecimento, característica de uma metodologia ativa que deve promover um ambiente em que eles

consigam “questionar, levantar hipóteses, investigar, desenvolver a autonomia, o senso crítico, a ética e a estética; despertando a curiosidade por meio de desafios e oportunidades para aguçar o pensamento” (ALMEIDA, 2000, p. 530-531).

As atividades propostas abordaram situações que envolviam a obtenção da lei de formação da função afim, ou seja, relação funcional, de forma lúdica, envolvendo preço de bilhetes em um parque de diversões.

Durante a produção das narrativas digitais, o contato com o *software* e a situação de função afim, os estudantes conseguiram compreender bem a ideia de variável e a relação de dependência entre as variáveis, pois o *software* não calcula sozinho sem a informação de “quantos”, no caso, a quantidade de brinquedos.

Ao longo da intervenção, conseguiu-se abordar outros conceitos implícitos como o de divisibilidade, a estimativa dos restos, na função do programa “se ...senão”. Os estudantes foram adquirindo novos conhecimentos por meio do processo do reconhecimento de padrões, encontrando similaridades que foram transferidas para a resolução das outras duas atividades (generalização), chegando até a abstração, desprezando os elementos desnecessários para concentrar-se naquilo que era relevante, até enunciar a lei de formação da função (algoritmo).

Cabe ressaltar que foi de extrema importância a interação do professor na solicitação da escrita inicial de um roteiro, o passo a passo (algoritmo), antes de começar a implementação no *software*. Os estudantes mobilizaram os elementos do pensamento computacional, depurando suas ações e o roteiro inicial foi alterado à medida que experimentaram.

Considerações finais

As narrativas digitais com o uso do *software Scratch* mostraram a autoria dos estudantes bem como os aspectos implícitos do seu PC diretamente relacionado às relações funcionais.

Destaca-se também a importância das metodologias ativas bem empregadas no processo de construção do conhecimento, visto que foram fundamentais para desenvolver o potencial criativo, o compartilhamento de ideias e a participação nas atividades.

As características das metodologias ativas conferiram uma atitude transformadora do modo de pensar, de comunicar, de representar o conhecimento e de realizar as tarefas propostas em sala de aula. Ainda mais, o ambiente computacional exerceu um papel catalisador das ideias, indo além da manipulação da ferramenta tecnológica, transformando-se em um elemento de estímulo a autoria.

Nesse interim, por meio das narrativas digitais construídas pelos estudantes, foi possível observar como estes organizavam suas ideias e conceitos, ratificando a ideia de “janela da mente” (ALMEIDA; VALENTE, 2012; VALENTE, 2016).

Durante a intervenção, no contato com o *software* e a situação de função afim, os estudantes conseguiram compreender bem a ideia de variável e a relação de dependência entre as variáveis, pois o *software* não calcula sozinho sem a informação de “quantos”, no caso a quantidade de brinquedos.

A partir do exposto, é importante incluir o desenvolvimento do PC articulada as metodologias ativas por meio do *software Scratch* como componente curricular da matemática, porque a facilidade de uso, a visualização e a ludicidade atraem os estudantes a compreenderem o lugar da matemática no cotidiano.

Referências

ALMEIDA, M. B. T.; VALENTE, J. A. *Integração currículo e tecnologias e a produção de narrativas digitais*. Currículo sem Fronteiras, v. 12, n. 3, p. 57-82, set./dez. 2012. Disponível em: <http://www.curriculosemfronteiras.org/vol12iss3articles/almeida-valente.pdf>. Acesso em: 8 de set. de 2019.

ALMEIDA, M. E. B. T. M. P. *O computador na escola: contextualizando a formação de professores. Praticar a teoria e refletir a prática.* 2000. Tese (Doutorado em Educação: Currículo) – Pontifícia Universidade Católica de São Paulo, São Paulo, 2000. Disponível em: <https://sapiencia.pucsp.br/handle/handle/9948>. Acesso em: 02 mar. 2019.

BLIKSTEIN, P. *O Pensamento Computacional e a Reinvenção do Computador na Educação.* 2008. Disponível em: <http://bit.ly/1XlbnNn>. Acesso em: 10 dez. 2017.

BRASIL. Ministério da Educação. Secretaria da Educação Básica. *Base nacional comum curricular – BNCC Versão Final.* Brasília, DF, 2018. Disponível em: http://basenacionalcomum.mec.gov.br/wp-content/uploads/2018/12/BNCC_19dez2018_site.pdf. Acesso em: 10 jan. 2019.

BRASIL, Ministério da Educação. Secretaria da Educação Básica. SEMTEC. PCN+ *Ensino Médio: orientações educacionais complementares aos Parâmetros Curriculares Nacionais. Ciências da Natureza, Matemática e suas Tecnologias.* Brasília, 2002.

BRACKMANN, C. P. *Desenvolvimento do pensamento computacional através de atividades desplugadas na educação,* 2017. 226. Tese (Doutorado) – Centro de Estudos Interdisciplinares em Novas Tecnologias na Educação, Programa de Pós-Graduação em Informática na Educação. Universidade Federal do Rio Grande do Sul, Porto Alegre, 2017.

FRANÇA, R. S.; AMARAL, H. J. C. *Proposta Metodológica de Ensino e Avaliação para o Desenvolvimento do Pensamento Computacional com o Uso do Scratch.* Disponível em: <http://www.br-ie.org/pub/index.php/wie/article/view/2646> Acesso em: 15 jan. 2019.

MATOS, A.; PONTE, J. P. *O estudo de relações funcionais e o desenvolvimento do conceito de variável em alunos do 8. Ano.* 2008. Disponível em http://www.scielo.org.mx/scielo.php?script=sci_arttext&pid=S1665-24362008000200003 Acesso em: 05 abr. 2018.

PAPERT, S. *A Máquina das Crianças: Repensando a Escola na Era da Informática.* Porto Alegre: Artes Médicas, 1994.

PAPERT, S. *Mindstorms: children, computers, and powerful ideas.* Basic Books, Inc, 1980. Disponível em <http://worrydream.com/refs/Papert%20-%20Mindstorms%201st%20ed.pdf>. Acesso em: 10 out. 2018.

PAPERT, S. *LOGO: computadores e educação.* São Paulo: brasiliense, 1985.

PINTO, A. S. *Scratch na aprendizagem de matemática no 1º Ciclo do Ensino Básico: estudo de caso na resolução de problemas.* 2010. 128p. Dissertação (Mestrado em Estudos da Criança – Tecnologias de Informação e Comunicação) – Universidade de Minho 2010. Disponível em: <http://goo.gl/gLKrdj>. Acesso em: 04 maio 2017.

PONTE, J. P. O conceito de função no currículo de Matemática. *Revista Educação e Matemática*, APM, Portugal, n. 15, p. 3-9, 1990.

SILVA, F. S.; ALMEIDA, A. C. F.; SILVA, K. A. G. *Perspectivas sobre o pensamento computacional no ensino superior com o software Scratch em cenários da Educação Financeira e da Economia Doméstica.* Disponível em: <http://tecedu.pro.br/wp-content/uploads/2017/12/Art24-vol.23-Dezembro-2017.pdf>. Acesso: 30 mai. 2018.

TEIXEIRA, A. C. *A introdução do raciocínio funcional no 5º ano do ensino fundamental: uma proposta de intervenção*. 2016. 134 f. Dissertação (Mestrado em Educação Matemática) – Universidade Estadual de Santa Cruz, Ilhéus, 2016.

TEIXEIRA, J. *Contribuições para o ensino de programação de computadores a futuros professores de matemática*. 2017. 219p. Tese (Doutorado em Ciências da Educação-Tecnologias Educativas) – Universidade de Minho 2017. Disponível em: <http://hdl.handle.net/1822/48711>. Acesso em: 15 mai. 2018.

TRANCOSO, F. F. *Implicações do pensamento computacional no desenvolvimento das relações funcionais com o software scratch: o caso da função afim*. 2019. 159p. Dissertação (Mestrado em Educação Matemática) – Universidade Estadual de Santa Cruz, Ilhéus, 2019.

TUCKER, A. et al. *A model curriculum for K–12 computer science: Final Report of the ACM K–12 Task Force Curriculum Committee*. Computer Science Teachers Association, New York, NY. 2003.

VALENTE, J. A. Integração do pensamento computacional no currículo da educação básica: diferentes estratégias usadas e questões de formação de professores e avaliação do aluno. *Revista e-Curriculum*, São Paulo, v. 14, n. 03, p. 864 – 897, jul./set.2016. Disponível em: <https://revistas.pucsp.br//index.php/curriculum/article/view/29051>. Acesso em: 14 mar. 2018.

VALENTE, J. A.; ALMEIDA, M. E. B.; GERALDINI, A. F. S. Metodologias ativas: das concepções às práticas em distintos níveis de ensino. *Revista Diálogo Educacional*, v. 17, n. 52, p. 455-478, 2017.

WING, J. M. Computational thinking. *Communications of the ACM*, v. 49, n. 3, p. 33–35, mar 2006. Disponível em: <https://www.cs.cmu.edu/~15110-s13/Wing06-ct.pdf> Acesso em: 05 abr. 2018.

WING, J. M. *Computational thinking benefits society*. on-line, 2014. Disponível em: <http://socialissues.cs.toronto.edu/index.html%3Fp=279.html>. Acesso em: 4 mar. 2019.

RECEBIDO: 21/01/2023
APROVADO: 07/04/2023

RECEIVED: 21/01/2023
APPROVED: 07/04/2023

RECIBIDO: 21/01/2023
APROBADO: 07/04/2023