

**Ricardo Scucuglia Rodrigues da
Silva**



Universidade Estadual Paulista (UNESP)
ricardo.scucuglia@unesp.br

George Gadanidis



Western University
ggadanid@uwo.ca

Rita de Cassia Idem



Universidade Estadual Paulista (UNESP)
ritaipvt@gmail.com

Lara Martins Barbosa



Universidade Estadual Paulista (UNESP)
lara-barbosa@hotmail.com

Yeda Seron Portera



Universidade Estadual Paulista (UNESP)
yedaseron@gmail.com

ASPECTOS DO PENSAMENTO COMPUTACIONAL DE ESTUDANTES DO ENSINO FUNDAMENTAL

RESUMO

Este artigo apresenta o desenvolvimento de duas tarefas matemático-artístico-computacionais realizadas com um grupo de quatro estudantes de Ensino Fundamental. As tarefas exploram a temática “padrões e álgebra” e foram realizadas utilizando aplicativos computacionais. Os dados foram produzidos em episódios de Experimento de Ensino por meio de filmagens e realização de diário de campo. O objetivo do estudo foi identificar as habilidades do pensamento computacional emergentes do contexto supracitado. O processo de interação das estudantes com os aplicativos foi interpretado por meio de ações de aprendizagem construcionistas. Como resultado, se verificou que houve indícios da emergência das habilidades de coleta de dados, análise de dados, simulação e automação em todas as atividades e das habilidades de decomposição do problema, paralelização e abstração em atividades mais complexas. A partir do estudo realizado, se concluiu que as tarefas exploradas possibilitaram a abordagem de diversos conceitos matemáticos e computacionais, e o desenvolvimento de habilidades essenciais a processos de resolução de problemas e à própria capacidade de aprender.

Palavras-chave: Pensamento computacional. Construcionismo. Padrões. Habilidades.

ASPECTS OF COMPUTATIONAL THINKING OF ELEMENTARY SCHOOL STUDENTS

ABSTRACT

This article presents the development of two mathematical-artistic-computational tasks carried out with a group of four elementary school students. The tasks explore the theme of “patterns and algebra” and were performed using computational applications. The data were produced in Teaching Experiment episodes through filming and a field diary. The aim of the study was to identify the computational thinking skills emerging from the aforementioned context. The students' interaction process with the applications was interpreted through constructionist learning actions. As a result, it was found that there was evidence of the emergence of the skills of data collection, data analysis, simulation and automation in all activities and the skills of decomposing the problem, parallelization and abstraction in more complex activities. From the study carried out, it was concluded that the tasks explored enabled the approach of several mathematical and computational concepts, and the development of skills essential to problem solving processes and to the ability to learn.

Keywords: Computational thinking. Constructionism. Standards. Skills.

Submetido em: 28/04/2020

Aceito em: 16/04/2021

Publicado em: 26/06/2021



1 INTRODUÇÃO

Neste artigo discutimos resultados de uma pesquisa em Educação Matemática cujo objetivo foi identificar habilidades do pensamento computacional de estudantes do Ensino Fundamental a partir do desenvolvimento de duas tarefas computacionais. Essas tarefas envolveram, além de conceitos computacionais, aspectos matemáticos e artísticos, e foram exploradas por meio de Experimentos de Ensino em dois anos diferentes. No primeiro ano, a Tarefa 1 foi realizada por um grupo de quatro estudantes que cursava o quinto ano do Ensino Fundamental. No segundo ano, o mesmo grupo de estudantes realizou a Tarefa 2 quando cursava o sexto ano do Ensino Fundamental. Portanto, analisamos atividades realizadas pelo grupo nos Anos Iniciais e nos Anos Finais do Ensino Fundamental. Com base nos dados produzidos, utilizamos uma lente **construcionista** (PAPERT, 1980; VALENTE, 2003) para entender o processo de realização das tarefas e identificamos habilidades do pensamento computacional emergentes.

As tarefas exploradas possuem um caráter matemático-artístico-computacional e são focadas na temática **padrões e álgebra**. De acordo com a Base Nacional Comum Curricular (BNCC), é fundamental que aspectos algébricos estejam presentes em situações de ensino e aprendizagem desde os Anos Iniciais do Ensino Fundamental, “[...] como as ideias de regularidade, generalização de padrões e propriedades da igualdade.” (BRASIL, 2017, p. 266). Além da associação de padrões à Álgebra, a BNCC destaca que essa área de estudo auxilia no desenvolvimento do pensamento computacional nos alunos, uma vez que, “[...] eles precisam ser capazes de traduzir uma situação dada em outras linguagens, como transformar situações-problema, apresentadas em língua materna, em fórmulas, tabelas e gráficos e vice-versa.” (BRASIL, 2017, p. 266).

Na seção seguinte, discutimos algumas ideias referentes ao pensamento computacional e ao construcionismo, que constituem o quadro teórico utilizado na pesquisa.

2 PENSAMENTO COMPUTACIONAL E CONSTRUCIONISMO

O pensamento computacional (PC) é uma temática, cada vez mais, investigada em interfaces envolvendo o Ensino de Ciência da Computação e outras áreas de conhecimento, e inclusive, a Educação Matemática. A expressão “pensamento

computacional” foi utilizada pela primeira vez por Papert (1980), mas ela se popularizou após o trabalho de Wing (2006), que utilizou o termo *computational thinking* ao discutir alcances e limites de processos de computação. De acordo com Wing (2008), o pensamento computacional é analítico, possuindo semelhanças com o pensamento matemático, o pensamento em engenharia e o pensamento científico. A autora aponta que o PC

Compartilha com o pensamento matemático, de maneira geral, meios pelos quais podemos abordar a solução de um problema. Ele compartilha com o pensamento em engenharia as maneiras gerais pelas quais podemos abordar um projeto e a avaliação de um sistema grande e complexo que opera dentro das restrições do mundo real. Compartilha com o pensamento científico as maneiras gerais pelas quais podemos abordar a compreensão da computabilidade, da inteligência, da mente e do comportamento humano. (WING, 2008, p. 3717, tradução nossa).

O PC possui diversas definições. Mannila *et al.* (2014, p. 2, tradução nossa) argumentam que ele “[...] é um termo que abrange um conjunto de conceitos e processos de pensamento da ciência da computação que ajudam na formulação de problemas e suas soluções em diferentes campos.”. Corroborando essa ideia, Lu e Fletcher (2009) destacam que o PC se relaciona ao processo de resolução de problemas, ao uso de algoritmos e à avaliação de diferentes dimensões de problemas e resultados. Para esses autores, PC

1) é uma maneira de resolver problemas e projetar sistemas que se baseiam em conceitos fundamentais para a ciência da computação; 2) significa criar e fazer uso de diferentes níveis de abstração, para entender e resolver problemas de forma mais eficaz; 3) significa pensar algorítmicamente e com a capacidade de aplicar conceitos matemáticos para desenvolver soluções mais eficientes, justas e seguras; 4) significa entender as consequências da escala, não só por razões de eficiência, mas também por razões econômicas e sociais. (LU; FLETCHER, 2009, p. 1, tradução nossa).

No contexto da Educação Básica, a exploração do PC possibilita uma abordagem educacional interdisciplinar (ISTE/CSTA, 2011; YADAV; STEPHENSON; HONG, 2017) e o desenvolvimento de habilidades cognitivas nos estudantes que auxiliam processos de resolução de problemas no mundo real e no mundo virtual (BARR; STEPHENSON, 2011). Como forma de promover um entendimento do PC e sua utilização no ensino, a *International Society for Technology in Education* (ISTE) e a *American Computer Science Teachers Association* (CSTA) apresentaram uma definição operacional de PC. Essa definição considera que o PC é um processo de resolução de problemas que engloba as seguintes características:

Formulação de problemas de uma maneira que nos permita usar um computador e outras ferramentas para ajudar a resolvê-los; Organização lógica e análise de

dados; Representação de dados através de abstrações, como modelos e simulações; Automatização de soluções através do pensamento algorítmico (uma série de etapas ordenadas); Identificação, análise e implementação de possíveis soluções com o objetivo de alcançar a combinação mais eficiente e eficaz de etapas e recursos; Generalização e transferência desse processo de solução de problemas para uma ampla variedade de problemas. (ISTE / CSTA, 2011, p. 7, tradução nossa).

Com base nas características supracitadas, ISTE e CSTA desenvolveram um quadro teórico que compila diferentes habilidades do PC, entendidas como práticas cognitivas, sendo elas: **coleta de dados, análise de dados, representação de dados, decomposição do problema, abstração, algoritmos e procedimentos, automação, simulação e paralelização**. A identificação dessas habilidades pode ser interpretada como um guia do trabalho docente para o desenvolvimento do PC. Neste artigo, entendemos que elas podem oferecer indícios desse desenvolvimento.

Coleta, análise e representação de dados se relacionam às ações de “[...] coletar informações apropriadas e selecionar informações relevantes; dar sentido aos dados, encontrar padrões, tirar conclusões; e organizar e representar dados em gráficos, quadros, palavras, imagens, tabelas, etc.” (MANNILA *et al.*, 2014, p. 13, tradução nossa). A **decomposição do problema** está ligada à ação de separar as tarefas em partes menores e mais fáceis de lidar. A **abstração** “[...] é um mecanismo importante no processo de solução de problemas, o qual permite simplificar a realidade e representar os aspectos mais relevantes de um problema e sua solução.” (RIBEIRO; FOZ; CARVALHEIRO, 2017, p. 8-10). Os **algoritmos e procedimentos** envolvem planejamento e desenvolvimento de etapas ordenadas na resolução de problemas. A **automação**, por sua vez, se relaciona a automatizar as tarefas repetitivas. A **simulação** envolve a construção de representações ou modelos de processos ou a utilização dos mesmos para a realização de experimentos. A **paralelização** ocorre por meio da utilização de mecanismos para se realizar tarefas simultaneamente.

Uma das maneiras de desenvolvimento e exploração do pensamento computacional é por meio da programação de computadores e atividades correlatas. O pioneirismo acerca de trabalhos sobre o uso de linguagem de programação na aprendizagem matemática de crianças está associado aos estudos de Papert (1980), os quais exploraram a utilização do *software* LOGO e a perspectiva denominada **construcionismo**. Essa noção considera que a construção do conhecimento é alcançada “[...] quando é apoiada pela construção de um tipo mais público ‘no mundo’ [isto é,] o produto pode ser mostrado, discutido, examinado, investigado e admirado. Está lá fora.” (PAPERT, 1993, p. 142, tradução nossa), como a construção de programas no

computador. Papert (1980, 1993) destaca que essa construção promove a aprendizagem ao proporcionar autonomia aos aprendizes, possibilidade de aprenderem com os próprios erros e de entenderem seus processos de pensamento por meio do favorecimento de ações de “pensar sobre o pensar”.

Com base na abordagem construcionista, Valente (2003, p. 100) desenvolveu uma visão teórica referida como ações ou espiral de aprendizagem que “[...] mostra que a interação usuário-computador pode ser vista em termos de uma sequência de ações - descrição - execução - reflexão - depuração - descrição - ...”. Essa perspectiva busca descrever o processo de interação aprendiz-computador da seguinte maneira: o processo se inicia com a proposição de um problema ao aprendiz, que **descreve** sua solução em termos compatíveis com o computador; o computador **executa** o que foi descrito; com base na execução, o aprendiz **reflete** sobre o resultado; se o aprendiz entende que o resultado é satisfatório, ele pode se empenhar na resolução de outro problema; mas se ocorrer alguma incongruência em relação ao resultado esperado, o aprendiz inicia o processo de **depuração**, que é a busca e correção de erros (MALTEMPI, 2005). Após um primeiro estágio de depuração, um novo segundo tipo de descrição surge, e assim por diante.

Embora Valente (2003, p. 101) proponha uma espiral de aprendizagem como uma sequência de ações dentro do construcionismo, “[...] isso não significa que cada ação ocorra uma de cada vez e na ordem sequencial como foi apresentada.”. Assim, os processos de aprendizagem baseados em computador são consideravelmente dinâmicos, ou seja, as ações geralmente acontecem simultaneamente, revelando a complexidade de aprendizagem.

Apresentados os fundamentos do PC e do construcionismo que guiaram a análise dos dados da pesquisa, apresentamos na seção seguinte a metodologia e os procedimentos metodológicos adotados para sua produção e análise.

3 METODOLOGIA

Esta pesquisa foi desenvolvida por meio de uma abordagem qualitativa (BICUDO, 1993). Os dados foram produzidos em um Experimento de Ensino (STEFFE; THOMPSON, 2000), procedimento metodológico que objetiva oferecer meios de compreensão da Matemática produzida por estudantes, ou seja, de como ocorre a aprendizagem e a produção de significados e de conhecimentos.

De acordo com Steffe e Thompson (2000, p. 268, tradução nossa) “A matemática dos estudantes é indicada pelo que eles falam e fazem aos se engajarem na atividade matemática e um objetivo básico dos pesquisadores em um experimento de ensino é construir modelos da matemática dos estudantes.”. Desse modo, o Experimento de Ensino ocorre por meio de episódios de ensino, encontros nos quais o pesquisador, atuando como professor, propõe atividades matemáticas a um grupo pequeno de estudantes e observa suas ações e falas na realização dessas atividades.

Nesses encontros, ocorre ainda a participação de uma testemunha, que auxilia o pesquisador na compreensão dos acontecimentos e na elaboração das atividades (ENGELHARDT *et al.*, 2004). Os episódios de ensino são gravados e as filmagens são analisadas após a realização de cada encontro. Os resultados podem, então, auxiliar na preparação de atividades seguintes. Além disso, na realização dos episódios de ensino, ocorre testes das hipóteses da pesquisa. E, durante e entre esses encontros, novas hipóteses são formuladas.

Na condução dos episódios de ensino, o professor-pesquisador atua como agente de ação, buscando interagir com os estudantes de forma responsiva e intuitiva, de modo a se engajar na maneira como eles pensam e raciocinam, como explicam Steffe e Thompson (2000, p. 278, tradução nossa)

Como agentes de ação, buscamos nos harmonizar com os alunos com quem trabalhamos, na medida em que nos “perdemos” em nossas interações. Não fazemos distinções intencionais entre nosso conhecimento e o conhecimento dos alunos e, para nós, experimentalmente, tudo é conhecimento dos alunos, enquanto nos esforçamos para nos sentirmos como eles.

Participaram do Experimento de Ensino quatro estudantes do Ensino Fundamental, identificadas por alunas A, B, C e D, que se engajaram na realização de duas tarefas. A realização da Tarefa 1 ocorreu quando as estudantes estavam no quinto ano. A Tarefa 2, por sua vez, foi realizada pelas mesmas estudantes quando cursavam o sexto ano. A produção de dados ocorreu na Escola Maria Peregrina (EMP)¹, localizada na cidade de São José do Rio Preto, SP. Os episódios de ensino foram conduzidos por dois dos autores deste artigo² e ocorreram em horário regular das aulas de forma particular com o grupo de estudantes. Tanto na realização da Tarefa 1 no quinto ano, como da Tarefa 2 no

¹ Trata-se de uma escola diferenciada, com projeto político-pedagógico fundamentado na Pedagogia de Projetos, considerada pelo Ministério da Educação como uma das 100 escolas mais inovadoras do Brasil. Do ponto de vista ético, a EMP solicita a todos responsáveis consentimento para realização de pesquisas na escola. No termo são descritas todas as dimensões sobre o uso de imagem. A EMP então concedeu anuência para a realização do Experimento de Ensino.

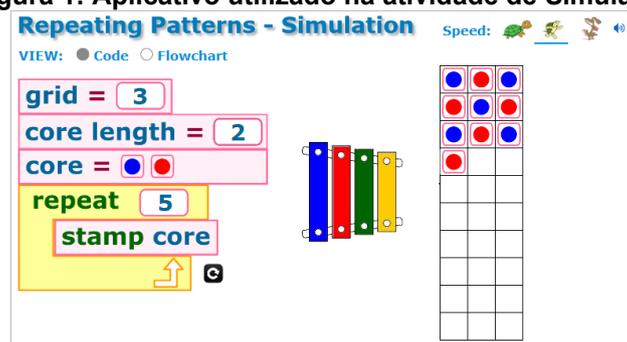
² Na análise nos referimos aos autores, de forma indistinta, por professor ou professor-pesquisador.

sexto ano, as professoras das estudantes ministraram aulas regularmente aos outros alunos durante os encontros. As estudantes foram selecionadas pela professora da turma, que levou em consideração os seguintes critérios: (a) interesse das estudantes e (b) disponibilidade de participação nos encontros considerando o “bom” desenvolvimento e adequação ao cronograma de atividades dos projetos de pesquisa das estudantes.

As tarefas exploradas possuíam um caráter matemático-artístico-computacional, pois abordavam conceitos matemáticos por meio da programação de computadores, além de elementos artísticos como cores, música e dança. A **Tarefa 1** envolveu a realização de três atividades: 1) **Simulação**, 2) **Música, Danças e Cores** e 3) **Extensões** e foi realizada em quatro episódios de ensino, com duração de três horas cada, no ano de 2017. Neste artigo analisamos as atividades de **Simulação** e de **Extensões**, atividades essas focadas na programação de computadores.

Na atividade de **Simulação** as estudantes exploraram um aplicativo de criação de padrões (GADANIDIS, 2017), disponível em <http://researchideas.ca/wmt/c2a0.html>. Nesse aplicativo, padrões de repetição são criados a partir da determinação do núcleo do padrão, o qual é formado por cores e correspondente nota musical emitida por um xilofone virtual. Na Figura 1, a seguir, apresentamos o aplicativo utilizado nessa atividade.

Figura 1: Aplicativo utilizado na atividade de Simulação



Fonte: <http://researchideas.ca/wmt/c2a0.html>

No lado esquerdo da figura se encontra o programa de criação de padrões, no qual é possível alterar diferentes parâmetros. Na parte central da figura se encontra o xilofone virtual e no lado direito se encontra a área de execução do padrão de acordo com os parâmetros determinados.

Na criação do padrão, é possível manipular os seguintes parâmetros: o número de colunas em cada linha da grade em que o padrão se constrói (parâmetro que chamaremos de **grade** na análise e é denominado “*grid* = [número]” no aplicativo), a quantidade de elementos do núcleo do padrão (**cores** – “*core length* = [número]”), as cores e ordem desses elementos (**display** – “*core* = [cores]”), e a quantidade de

repetições do núcleo na construção do padrão (**repetição** – “*repeat* [número] {*stamp core*}”).

Na atividade denominada **Extensões** as estudantes exploraram outro aplicativo (GANANIDIS, YIU; 201-), disponível em <http://mathsurprise.ca/apps/patterns/v1/>, o qual também permite a criação de padrões de repetição. Esse aplicativo foi construído por meio da programação de blocos denominada *Blockly*. Ele possui maior quantidade de parâmetros do que o aplicativo utilizado na primeira atividade, possibilitando a construção de padrões de repetição cujos elementos possuem formas, cores e sons.

No aplicativo existem dois tipos de códigos de programação por blocos que podem ser manipulados e editados. Esses códigos possibilitam a construção de padrões lineares (um exemplo desse tipo de padrão e seu código de construção é apresentado na Figura 2) e bidimensionais (um exemplo é apresentado na Figura 6). Nos códigos é possível determinar, em ambos os tipos de padrões, os seguintes parâmetros de construção: o tamanho dos elementos do padrão (*set size to* [número]), o ângulo de rotação dos elementos do padrão (*set rotation to* [número]), o ponto horizontal no plano cartesiano em que o padrão iniciará a construção (*set x to* [número]) e o ponto vertical (*set y to* [número]), a inclinação da construção do padrão (*set angle to* [número]), o distanciamento entre os elementos do padrão (*set step size to* [número]), a velocidade de construção do padrão (*set stamp rate to* [número]), cores e formas dos elementos do núcleo do padrão (*add* [cor] [forma] *to pattern*) e a quantidade de elementos do núcleo (que é representada pela quantidade de blocos do tipo “*add* [cor] [forma] *to pattern*” incluídos no código de programação). No código de construção de padrões lineares é possível alterar a quantidade de elementos no padrão (*repeat* [número] *times do* {*stamp*}). O código de construção de padrões bidimensionais se diferencia do código de padrão linear pelo seguinte código “*repeat* [número] *times do* {*repeat* [número] *times do* {*stamp*}; *set x to* [número]; *change y by* [número]}”, por meio dele é possível determinar a quantidade de linhas do padrão (*repeat* [número] *times do* {...}), a quantidade de colunas do padrão (*repeat* [número] *times do* {*stamp*}), o local de construção do primeiro elemento de cada linha (*set x to* [número]) e a distância entre as linhas (*change y by* [número]). O código de construção de padrão bidimensional explora a ideia de *nested loop*: um ciclo de programação que ocorre internamente a outro ciclo de programação.

A **Tarefa 2** foi realizada a partir de quatro atividades e também ocorreu por meio da utilização do aplicativo de Gadanidis e Yiu [201-]. As atividades foram desafios de construção de padrões de repetição. Os padrões explorados envolviam a construção de

padrões lineares (Atividade 1), de padrões lineares formados pela execução de mais de um programa (Atividade 2), de padrões bidimensionais que exploram a ideia de *nested loop* (Atividade 3), e de padrões bidimensionais que exploram a ideia de *nested loop* e que são formados pela execução de mais de um programa (Atividade 4). Essa tarefa foi realizada no ano de 2018 e ocorreu em quatro episódios de Experimento de Ensino com duração de duas horas cada.

Os encontros foram filmados com câmera de vídeo e foi elaborado diário de campo. Além disso, todas as ações das estudantes na tela do computador foram registradas (incluindo falas e expressões faciais) por meio da utilização do *software Flashback Pro5*. As filmagens foram analisadas utilizando o modelo de análise de vídeos proposto por Powell, Francisco e Maher (2004), o qual é formado pelos seguintes procedimentos não lineares: **observação** – etapa em que o pesquisador assiste as filmagens com a intenção de se familiarizar com seus dados; **descrição** – nessa etapa o pesquisador descreve o conteúdo das filmagens em determinados intervalos de tempo; **identificação de eventos críticos** – ocorre a identificação dos momentos significantes ou críticos de acordo com os objetivos do estudo; **transcrição** – o pesquisador transcreve as falas e acontecimentos dos eventos críticos; **codificação** – nessa etapa o pesquisador classifica os eventos críticos por meio de temas ou códigos; **construção do enredo** – o pesquisador busca estabelecer relações entre eventos críticos, transcrições e codificações por meio de um enredo; **composição da narrativa** – ocorre a escrita do relato da pesquisa, como artigo, livros, teses, relacionando o enredo ao referencial teórico adotado.

Na Seção que segue, apresentamos uma discussão e análise dos dados produzidos na realização das atividades propostas nas tarefas.

4 DISCUSSÃO E ANÁLISE DOS DADOS

Na análise que apresentamos sobre o Experimento de Ensino com estudantes do Ensino Fundamental, exibimos quadros formados pelos componentes da espiral de aprendizagem de Valente (2003). Em nosso estudo, identificamos as ações de descrição e execução como entradas de comandos dos alunos e execuções de computadores, respectivamente. No entanto, reflexões e ações de depuração ocorreram simultaneamente, por meio dos diálogos e interações de estudantes mediados pelo professor-pesquisador e pelos aplicativos utilizados no computador.

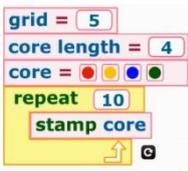
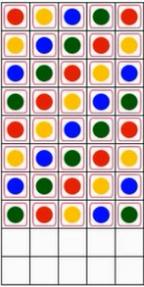
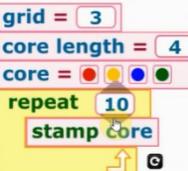
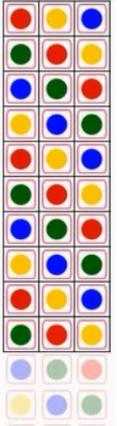
Nas subseções seguintes apresentamos o desenvolvimento das Tarefas 1 e 2, descrevendo os eventos críticos identificados na análise das filmagens. Além de evidenciar as ações de aprendizagem desses eventos, buscamos destacar as habilidades do pensamento computacional emergentes na realização das atividades.

4.1 Tarefa 1

No início da atividade de **Simulação**, o professor-pesquisador introduziu o aplicativo para as alunas apresentando-as diversos exemplos de construção de padrões. Depois da exploração inicial, ele questionou sobre como as alunas poderiam deixar as cores em uma linha diagonal. Então, a aluna A se recordou que o professor havia feito um exemplo em que as linhas ficavam na diagonal e repetiu o que havia sido feito, utilizando o *display* vermelho/amarelo/azul e o número 4 no parâmetro *grade*. Em seguida, o professor perguntou como elas poderiam deixar uma regularidade na diagonal, mas utilizando quatro cores. Elas aumentaram então uma coluna, ficando com *grade* 5. A diagonal que elas conseguiram construir ficou “para a direita” e o professor pediu que elas fizessem para “a esquerda”. Depois de algumas tentativas elas conseguiram fazer a “diagonal para esquerda”.

No Quadro 1, apresentamos a transcrição desse episódio utilizando ações de aprendizagem (VALENTE, 2003).

Quadro 1: Construção de padrões diagonais

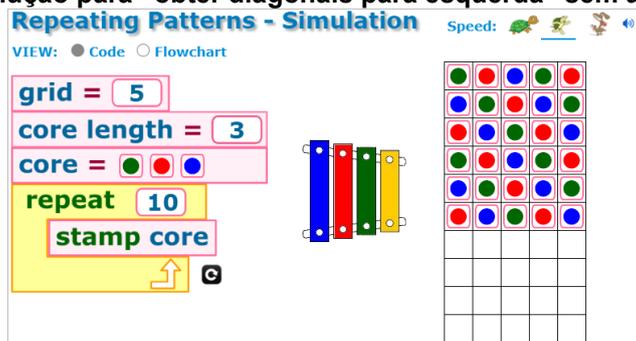
Descrição	Execução	Reflexão/Depuração
		<p>Professor: Muito bem, agora eu quero... Que tenha diagonais pra esquerda.</p> <p>Aluna B: Nossa!</p> <p>Aluna C: Ai meu Deus!</p> <p>Aluna A: É só trocar a sequência de cor aqui!</p> <p>Aluna B: É tipo tá... Coloca o verde aqui.</p> <p>Aluna A: Verde no lugar do vermelho, não!</p> <p>Aluna C: Calma gente!</p> <p>Aluna A: Verde, aqui põe o azul, aqui o amarelo.</p> <p>Aluna B: Aqui o amarelo!</p> <p>Aluna A: Vai ser a mesma coisa.</p> <p>Aluna D: Ah não!</p>
		<p>Aluna A: Vamos analisar!</p> <p>Aluna B: Certo!</p> <p>Aluna A: Põe no três, três lugares.</p> <p>Aluna D: Três grades?</p> <p>Aluna B: Isso!</p> <p>Aluna A: Ficou!</p> <p>Aluna C: Agora ficou!</p> <p>Aluna B: Nossa ficou lá embaixo!</p> <p>Aluna C: Ficou, olha aqui gente.</p> <p>Aluna A: Fico até mais.</p> <p>Aluna B: Fico até mais.</p> <p>Professor: Conseguiram? Ah foi rápido hein.</p>

Fonte: Dados da pesquisa.

Podemos notar que o professor-pesquisador busca construir um cenário de aprendizagem no qual as alunas façam conexões entre representações em um processo de simulação computacional. Nesse caso, busca-se conexões entre os parâmetros configuráveis (grade, cores, *display* e repetição) e a simulação realizada (composição visual, mas também sonora das cores na grade).

No que segue, as alunas buscaram diminuir o número de repetições, para que os elementos coubessem totalmente na grade. Utilizando conhecimento acerca de multiplicação, notaram que o total de elementos era 40 (4 elementos no *display* vezes *Repeat* 10) e que o total de espaços na grade era 30 (grade com 3 vezes 10 linhas). Depois, por meio de uma provocação do professor, as alunas buscaram outra solução para o problema “obter diagonais para esquerda” sem alterar a grade, ou seja, alterando o número de elementos e/ou as cores do *display*. Após algumas tentativas, as alunas optaram por suprimir a cor amarela com um *display* de três elementos: verde/vermelho/azul. A solução pode ser vista na Figura 2.

Figura 2: Solução para “obter diagonais para esquerda” sem alterar a grade



Fonte: Dados da pesquisa

Dessa maneira, obtiveram “diagonais para a esquerda” de todas as 3 cores.

No próximo episódio (Quadro 2), o professor propôs que as estudantes construíssem padrões que formam linhas verticais, explorando a relação da construção com a configuração da grade e do número de elementos do *display*.

Quadro 2: Construção de padrões verticais

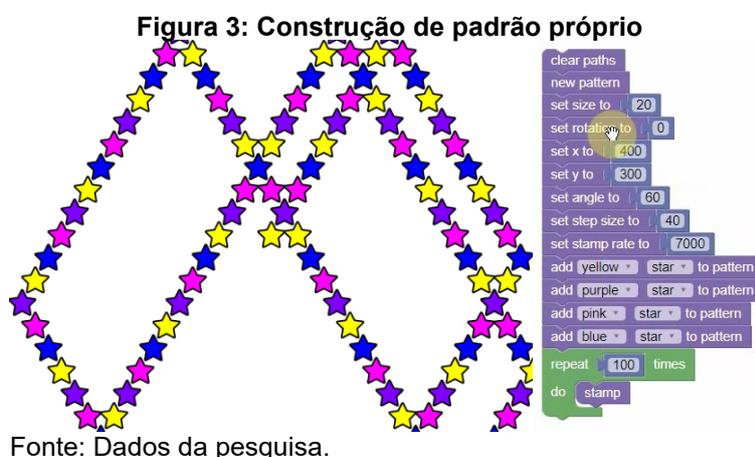
Descrição	Execução	Reflexão/Depuração
		<p>Aluna D: Aee!</p> <p>Professor: Muito bem gente! Esse vocês acharam a solução mudando o que? O número de cores né? Tem alguma solução mudando a grade?</p> <p>Aluna A: Tem! Aqui você põe quatro. Põe aqui dois. Aqui quatro. Ah não vai da certo. Põe três. Aqui põe três, põe três. Não, aqui três aqui quatro.</p> <p>Aluna C: Não vai da certo.</p> <p>Aluna A: Aqui dois.</p> <p>Professor: Lembrando que a grade pode ser mais né, a grade vai até...</p> <p>Aluna D: Ai!</p> <p>Professor: Ai é 4x4.</p> <p>Aluna C: Põe seis e três.</p>
		<p>Professor: Oh então vamos perceber... legal né? Certo? Olha só grade seis, número de cores três, tá bom? Seis dividido por três da quanto?</p> <p>Aluna A: Seis dividido por três?</p> <p>Aluna C: três!</p> <p>Aluna A: É três! Não, calma. Seis dividido por três da... dois.</p> <p>Professor: Dois! Legal está bom?</p> <p>Aluna D: Verdade!</p> <p>Professor: Isso, agora eu vou, deu e deu exato, né? Tudo bem? Deu número exato! Notem que 4 é divisível por 2 e 6 é divisível por 3. Qual seria outra solução?</p> <p>Aluna A: 8 e 4, eu acho, não é?</p>

Fonte: Dados da pesquisa.

Notamos que as alunas conjecturaram que ocorrem linhas verticais quando o número da grade é divisível pelo número de elementos do *display*.

A atividade **Extensões** buscou explorar formas de pensamento mais complexas que as requeridas na atividade anterior. As estudantes tiveram dificuldade na utilização do

aplicativo de Gadanidis e Yiu [201-] para a construção de padrões pré-determinados, mas conseguiram explorá-lo na construção de seus próprios padrões. Elas levaram cerca de 35 minutos para construir o padrão apresentado na Figura 3. Mesmo sem compreender a funcionalidade de todos os parâmetros, elas puderam construir uma figura esteticamente desejável de seu ponto de vista. Nesse processo estabeleceram relações entre os parâmetros do código e a execução do aplicativo.



Considerando as diferentes habilidades do PC elencadas por ISTE e CSTA (2011), entendemos que na interação com os aplicativos as estudantes estavam em constante processo de coleta e análise de dados. A cada execução dos aplicativos, as alunas analisavam informações que auxiliaram tanto no entendimento do funcionamento dos códigos, como no processo de resolução dos problemas propostos pelo professor-pesquisador. Os aplicativos ofereceram modelos digitais de criação de padrões de repetição e as participantes se mostraram eficientes na utilização desse modelo na resolução dos problemas propostos ou na construção de padrões de seus interesses, eficiência essa que se relaciona à habilidade denominada simulação. Além disso, a construção dos padrões foi possível por meio da automação, caracterizada pela capacidade de mecanização de processos, atividade que é intrínseca ao funcionamento dos aplicativos.

4.2 Tarefa 2

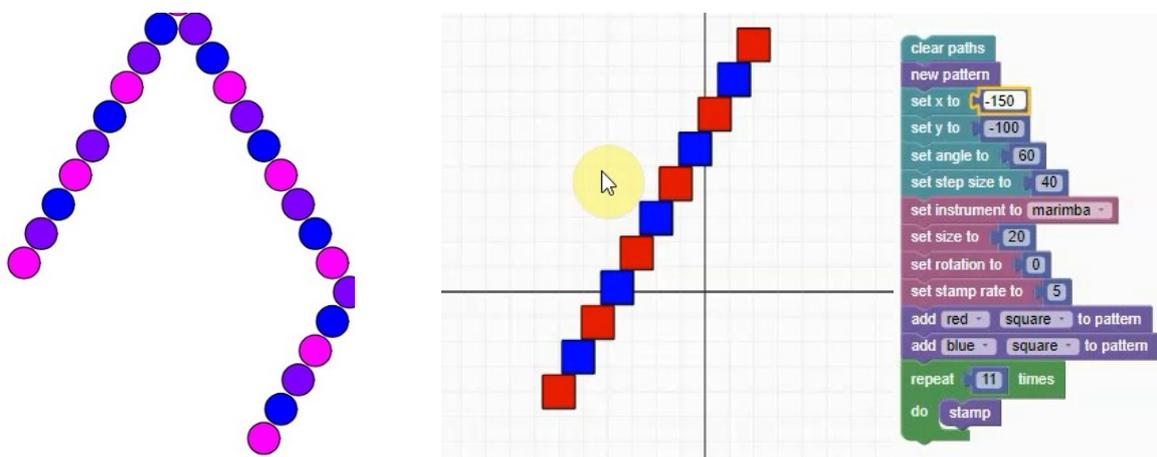
Antes da realização da Atividade 1, as participantes exploraram o aplicativo com a intenção de se familiarizar novamente com ele. Inicialmente, as participantes exploraram diferentes comandos. A partir dessa exploração observaram que as alterações resultavam na mudança do padrão exibida na tela. Apesar de perceberem as variações, as alunas

não identificaram a princípio quais os comandos responsáveis para cada mudança. Os comandos ligados às cores e formatos dos objetos da sequência foram os primeiros a serem identificados e os que abordavam conceitos foram identificados após muitas tentativas para simular o que estava sendo alterado. Exemplo disso são os comandos “set x to [número]” e “set y to [número]”, responsáveis por alterar os pontos x e y, respectivamente, como coordenadas no plano cartesiano. As primeiras tentativas das alunas foram feitas com variação muito pequena dos números, dificultando a visualização da mudança na tela. Após várias tentativas, optaram por testar um comando por vez e aumentar os valores. A mudança de estratégia permitiu que identificassem a função dos comandos em mudar a posição que a sequência de formas iniciaria, ou seja, as coordenadas.

Após a exploração do aplicativo, se deu início à Atividade 1, que envolvia a construção de padrões lineares. Foi apresentado um código como exemplo (ver Figura 4) e as alterações, para obter a nova sequência, deveriam ser realizadas a partir dele.

A primeira mudança foi na parte visual, observaram quais eram as cores, formatos e quantidade de objetos repetidos. Depois, determinaram as coordenadas e o ângulo. Começaram mudando os valores aleatoriamente e não obtiveram resultado, pois quando colocavam valores altos para x, a sequência se deslocava para direita, mas precisavam encontrar um que a deslocasse para a esquerda. Uma das alunas sugeriu que diminuíssem o valor, entretanto, ficaram limitadas em zero como menor valor. Uma das alunas recordou que poderiam obter números menores se colocassem o “sinal negativo” na frente do número, dessa forma, conseguiram encontrar a coordenada (ver Figura 4).

Figura 4: Exemplo e Primeiro padrão explorado na Atividade 1

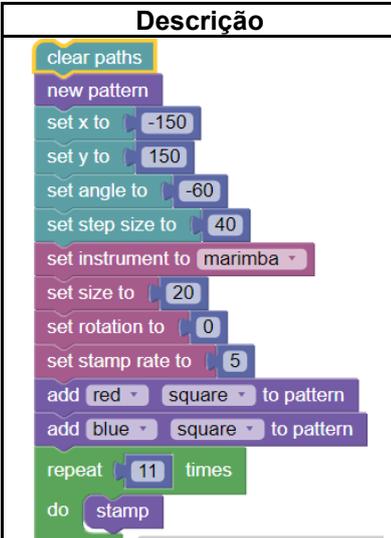
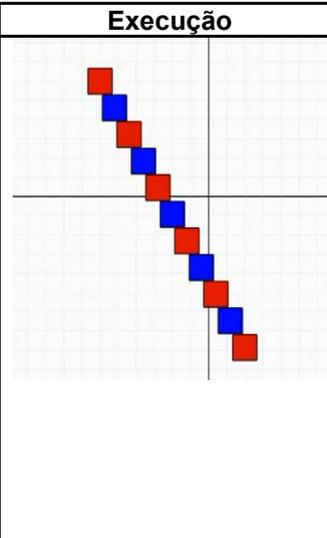


Fonte: Dados da pesquisa.

Nessa atividade, as estudantes utilizaram conhecimentos sobre os números inteiros e a partir das tentativas para modificarem o código perceberam qual era a posição deles na reta.

A segunda sequência era semelhante à construída anteriormente, as diferenças estavam no ângulo e na coordenada. Para realizar essa atividade havia a necessidade de que compreendessem o conceito de ângulo, bem como onde ele estava sendo representado no código. No decorrer da atividade percebemos que as estudantes não possuíam esta noção, as alterações que realizavam eram de maneira aleatória na tentativa de acertar o ângulo. O professor considerou conveniente a interferência para explicar sobre a circunferência com 360° e como poderiam identificar os ângulos a partir dela. Após a explicação conseguiram determinar o ângulo e a coordenada que resultava na sequência solicitada. No Quadro 3, a seguir, com base em elementos das ações realizadas, podemos acompanhar um trecho da discussão e da conclusão obtida.

Quadro 3: Construção do segundo padrão da Atividade 1

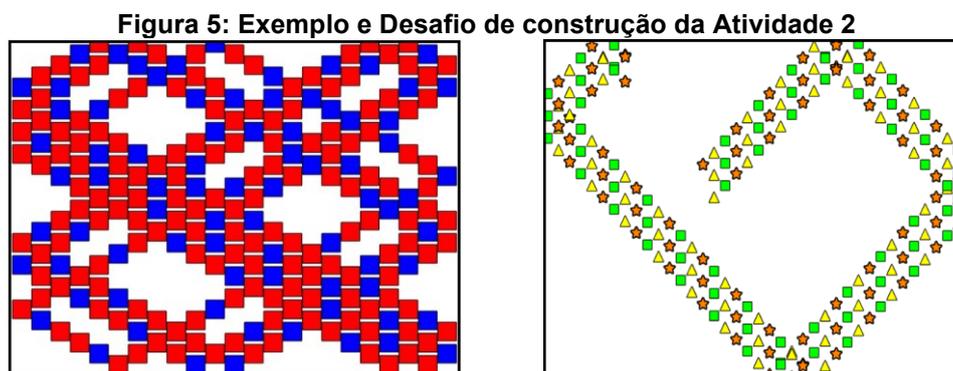
Descrição	Execução	Reflexão/Depuração
		<p>Aluna A: 120? Não, mais 150 também né? Um pouquinho mais né? Isso, agora o ângulo a gente tem que mudar. Coloca 30, sei lá. Aluna A: Mas por que ainda tá pra cima? Aluna B: O ângulo! Aluna A: Aumentar o ângulo? Coloca 40. Ou é mais do que 60? Seria mais né? Aluna B: Coloca 90... Não, não. Professor: Lembra que pode número negativo. (Explicação do professor) Aluna A: Então coloca -100. Aluna A: Agora a gente tem que ir pra lá né? Coloca -60... Isso ficou legal!</p>

Fonte: Dados da pesquisa.

No final da atividade, houve uma conversa sobre a outra forma de representar o ângulo -60° , que é o mesmo que ser representado por 300° . O valor do ângulo depende do sentido, ou seja, no sentido horário obtemos valores negativos e no sentido anti-horário obtemos valores positivos.

A Atividade 2 da Tarefa 2 abordava a execução simultânea de mais de um código, ou seja, a combinação entre os códigos criados. Dessa forma, os códigos são criados em janelas diferentes e com o botão "run all code" todos eles são reproduzidos ao mesmo tempo. Para conhecerem esse novo comando, inicialmente foi apresentado um exemplo dessa forma de construção de padrões (ver Figura 5). Depois de analisarem o exemplo,

foi proposta uma nova atividade: criar um código para obter a sequência apresentada na Figura 5 através da ideia de execução simultânea.

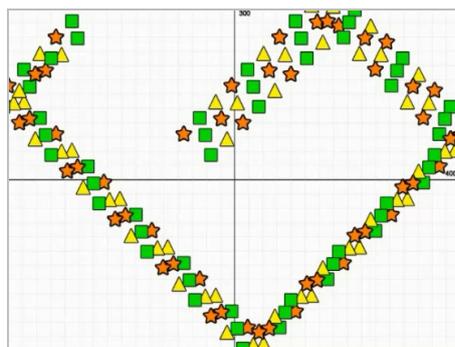


Fonte: Dados da pesquisa.

Antes de qualquer tentativa, as alunas ficaram analisando a imagem e discutindo formas que poderiam ser usadas para resolver a atividade. Foi identificado que não haviam percebido que era preciso criar os códigos separadamente. O professor questionou quantas linhas a sequência formava, dessa forma, compreenderam que seria necessário criar os códigos separados para obter as três linhas. Do mesmo modo que nas atividades anteriores, as cores e formas foram os primeiros elementos a serem modificados. Para encontrar a quantidade de vezes que os objetos se repetiam, a Aluna D contou todos os objetos que estavam na imagem e disse que na repetição deveriam colocar 187 vezes. As outras integrantes do grupo questionaram se seria isso mesmo e perceberam que na verdade deveriam dividir este número entre os três códigos. Uma nova contagem foi realizada e consideraram que cada linha possuía 60 elementos sendo repetidos.

Nas primeiras tentativas haviam colocado o mesmo ângulo e coordenada nos três códigos, porém a execução apresentava apenas uma linha. Assim, decidiram mudar os valores das coordenadas para que iniciassem lado a lado. Com a mudança das coordenadas precisavam determinar o ângulo para cada uma das linhas, que satisfizesse a sequência completamente. A construção exibida na Figura 6 foi o mais próximo que conseguiram chegar da sequência solicitada, sendo essa considerada pelas alunas a melhor resolução que puderam obter.

Figura 6: Resolução do desafio da Atividade 2 pelas estudantes

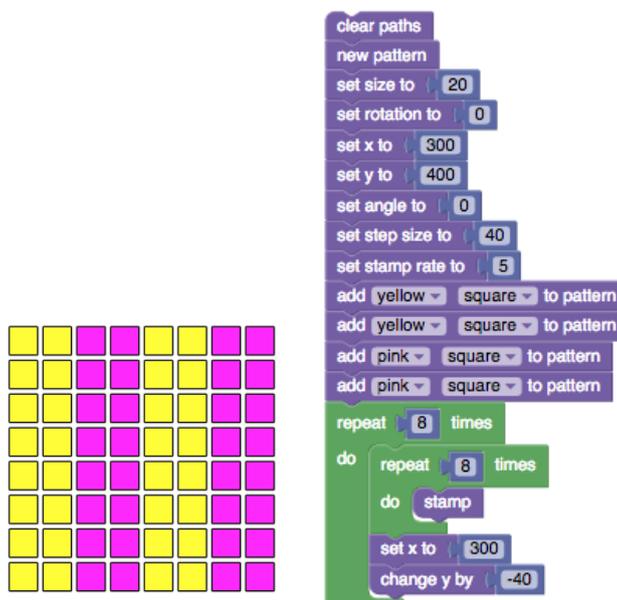


Fonte: Dados da pesquisa.

É possível notar que os valores escolhidos para os ângulos satisfaziam apenas algumas partes.

A terceira atividade da tarefa explorou a ideia de *nested loop* em computação. Trata-se de um “*loop* dentro de um *loop*” que, no caso do aplicativo, é representado por “um comando *repeat* dentro de um comando *repeat*”. Essa combinação de comandos atribui à construção uma qualidade bidimensional (Figura 7).

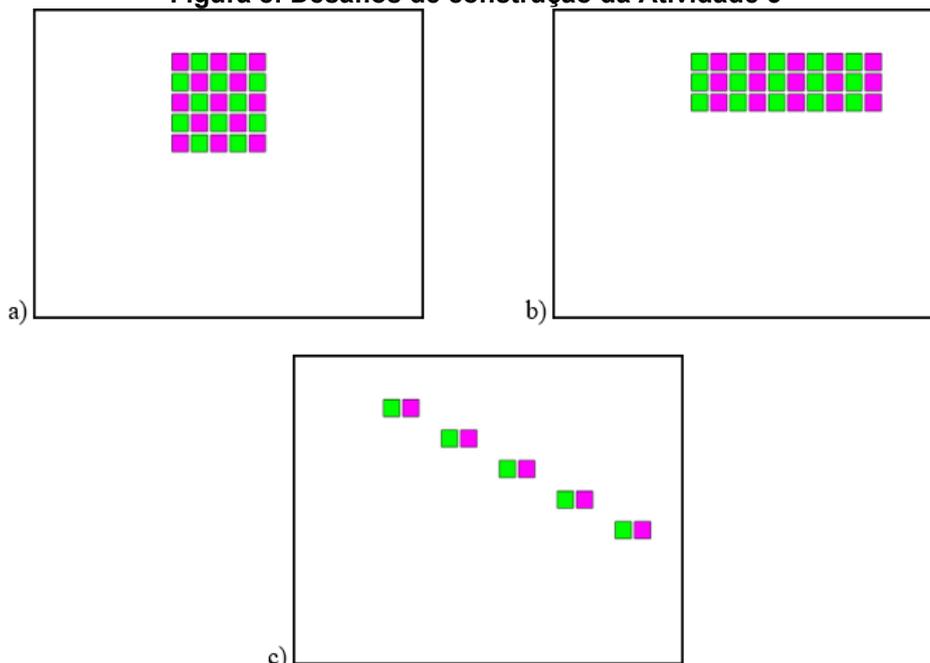
Figura 7: Exemplo de padrão bidimensional explorado na Atividade 3



Fonte: Dados da pesquisa.

As participantes não demonstraram dificuldade na construção dos padrões apresentados nos itens a e b na Figura 8. Elas entenderam e conseguiram aplicar a ideia de *nested loop* na construção desses padrões. Entretanto, elas não conseguiram associar a construção do padrão do item c a esse conceito computacional.

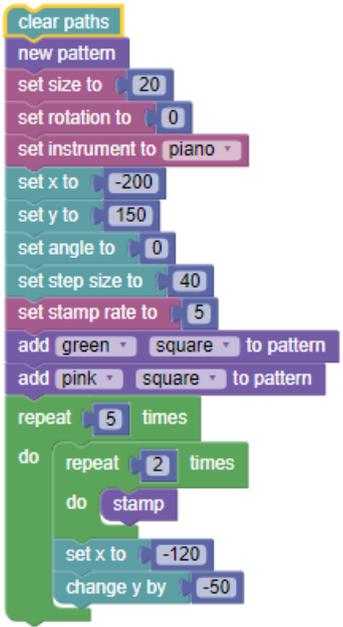
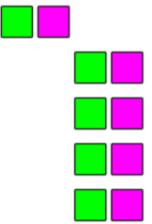
Figura 8: Desafios de construção da Atividade 3



Fonte: Dados da pesquisa.

No momento que viram a imagem do item c disseram que, para resolver, deveriam usar cinco janelas/programas e tentaram alterar os códigos. Apesar do principal objetivo dessa atividade ser o conceito de *nested loop*, poderiam ser desenvolvidas outras estratégias de resolução que satisfariam o problema proposto. A estratégia conduzida pelas alunas tinha potencial para fomentar uma resolução da atividade, porém o resultado que obtiveram não foi o esperado. Analisando os vídeos é possível notar que, durante a cópia dos códigos originalmente exibidos no exemplo, elas confundiram o que deveria ser copiado. Não identificaram o que estava causando o erro e procuraram outra estratégia. Na nova estratégia usaram o *nested loop*, ou seja, “um *repeat* em um *repeat*”. No Quadro 4 apresentamos o diálogo que aconteceu para o desenvolvimento do algoritmo, contendo um comando *repeat* inserido em outro comando *repeat*.

Quadro 4: Construção do item c da Atividade 3

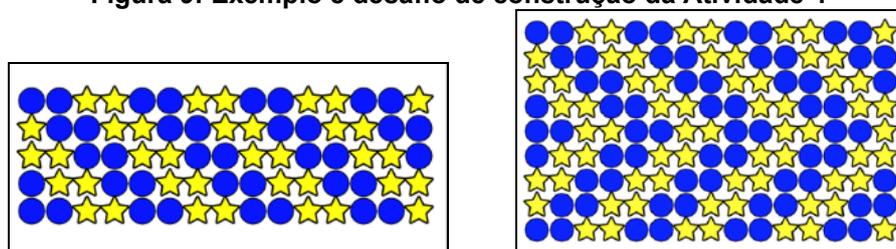
Descrição	Execução	Reflexão/Depuração
		<p>Aluna A: no repetir vamos colocar 50? Não, 5. Isso, mas agora tem que espaçar eles. Professor: Isso! Aluna D: Aqui, não é esse? Aluna A: Não! Não é nessas aqui de baixo? Professor: Tenta mudar aí alguma coisa! Aluna D: vamos tentar mudar esse daqui. Um número maior ou menor a gente coloca aqui? Professor: Você quer que ele vai pra onde? Aluna A: Pra lá (direita). Professor: Então tem que ser um número maior. Aluna A: Coloca 300. Professor: Se colocar 300 vai na onde? Aluna D: Mais ou menos pra cá. Aluna A: No meio. (colocaram $x = -100$) ... Professor: O primeiro já foi, agora precisamos pensar para todos ficarem assim Aluna A: Precisa colocar mais! Não, ou tem que ser esse daqui? E o que significa isso? Vai ter que mudar o x e y né? Aluna D: E o ângulo? Aluna A: Pode ser! Aluna D: Não, não! Desisto.</p>

Fonte: Dados da pesquisa.

O resultado não foi o que estava sendo solicitado na atividade, foi um resultado aproximado. Depois de várias tentativas as alunas não conseguiram modificar as demais repetições e desistiram de determinar uma solução mais aprimorada para o problema.

A Atividade 4 compilava tudo o que havia sido explorado nos encontros anteriores. Como nas atividades anteriores, com base em um exemplo, as modificações deveriam ser realizadas para obter uma nova sequência e um novo objeto. A Figura 9 exibe o exemplo oferecido e o objeto a ser construído com base no exemplo. Podemos notar que o objeto a ser construído envolve a produção de significados que inclui o conceito de simetria e, sua resolução perpassa pela exploração de conceitos analisados nas atividades anteriores (e.g., coordenadas no plano cartesiano e ângulos, combinação de códigos e *nested loop*).

Figura 9: Exemplo e desafio de construção da Atividade 4

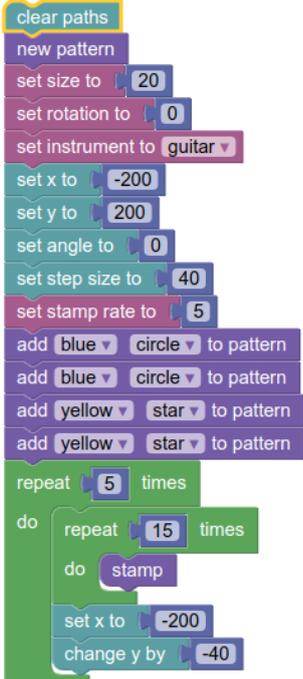
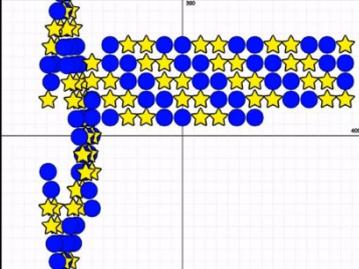


Fonte: Dados da pesquisa.

O primeiro passo realizado pelas alunas para resolver essa atividade foi copiar o código em uma segunda janela para que ficassem um na parte superior e o outro embaixo. Feito isto, identificaram quais mudanças deveriam ser realizadas para que a imagem formasse uma “seta apontando para o lado direito”. O código que estava na parte superior era o que precisava de mais alterações, pois deveriam fazer com que “a linha inclinada para a esquerda se inclinasse para a direita”. Após serem questionadas sobre qual alteração deveriam fazer para que isto ocorresse, identificaram que seria necessário mudar o valor do ângulo. A justificativa para o ângulo que iriam usar era que para ocorrer a inclinação para o outro lado deveriam colocar um “valor muito grande”. Assim que iniciaram a execução do código visualizaram os objetos em desordem e ficaram questionando por que havia acontecido aquilo e pensando em outras estratégias que poderiam desenvolver.

Era possível criar mais de uma estratégia para encontrar a solução desse problema. Uma delas era mudar a coordenada x de modo que a sequência fosse iniciada na posição em que o primeiro código terminava. Essa foi a que decidiram seguir. No Quadro 5, apresentamos a transcrição sobre o momento em que desenvolveram a primeira estratégia.

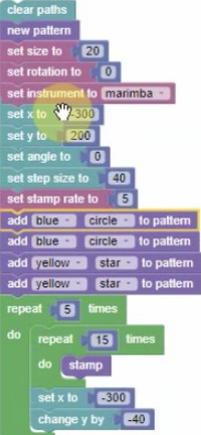
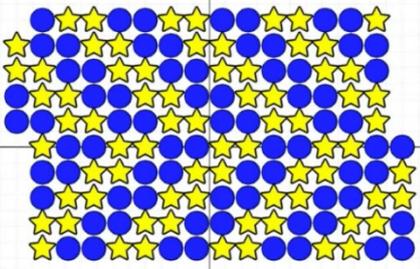
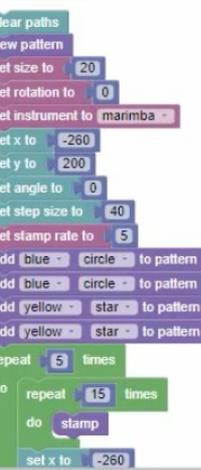
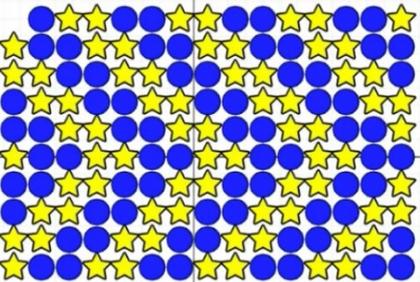
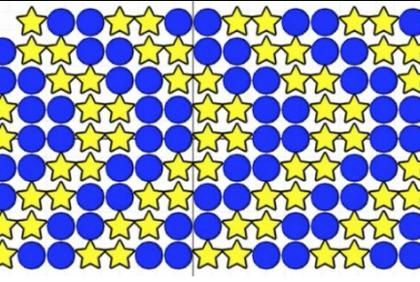
Quadro 5: Resolução da Atividade 4

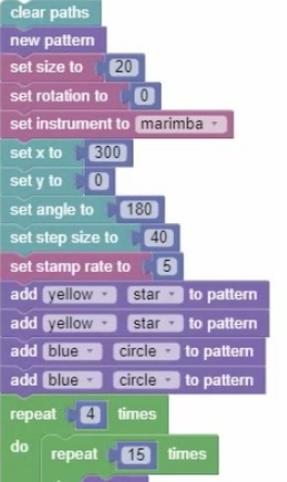
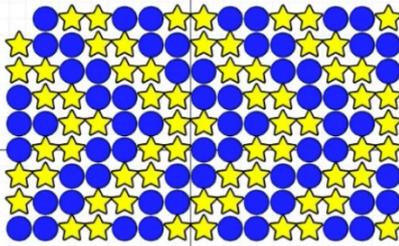
Descrição	Execução	Reflexão/Depuração
 <pre> clear paths new pattern set size to 20 set rotation to 0 set instrument to guitar set x to -200 set y to 200 set angle to 0 set step size to 40 set stamp rate to 5 add blue circle to pattern add blue circle to pattern add yellow star to pattern add yellow star to pattern repeat 5 times do repeat 15 times do stamp set x to -200 change y by -40 </pre>		<p>Professor: Agora o de cima está igual né? O que você tem que fazer para mudar a inclinação?</p> <p>Aluna A: Como que vou fazer isso? É um desses dois? É o 40! O ângulo!! Coloco um número grande?</p> <p>Professor: O ângulo. Ele está zero né? Quando está zero como que fica?</p> <p>Aluna A: No meio.</p> <p>Professor: Não... Pra cá é zero então ele vem pra direita, 90 ele vai para cima.</p> <p>Aluna A: Ah, então tenho que colocar um número grande! Tipo qual?</p> <p>Professor: Testa o que você acha!</p> <p>Aluna A: Sei lá, 200. Vou colocar qualquer número, vou colocar 100. Não, não, volta!</p> <p>Aluna D: Não, o que você fez?</p> <p>Aluna A: É precisa ser um número bem grande para ir pra lá.</p>

Fonte: Dados da pesquisa.

Para concluírem a Atividade 4, foi necessário um episódio de ensino de 2 horas, no qual fizeram várias tentativas e mudaram de estratégias diversas vezes. Para efetiva conclusão, descrita no Quadro 6, foi necessária uma mediação do professor.

Quadro 6: Resolução da Atividade 4 e mediação do professor

Descrição	Execução	Reflexão/Depuração
		<p>Aluna C: Acho que agora foi. Aluna D: Quase! Aluna A: Põe -300, aí acho que vai dar. Professor: Melhorou, mas o alinhamento ainda não está legal. Aluna C: Desorganizou tudo! Aluna B: Muda o outro x. Aluna A: -300 também?</p>
		<p>Aluna D: Agora foi demais. Professor: Olha o passo, que é o comando step size. Não é 40? Então talvez precise aumentar 40. Aluna D: 340? Professor: Não. Aluna A: Ah! -260. Professor: Legal!</p>
		<p>Aluna C: Agora deu, parece. Aluna B: Deu certo! Professor: Será? Eu acho que ainda não. Aluna A: Não. Está diferente. Professor: Quantas linhas têm no exemplo e quantas linhas têm no objeto de vocês? Aluna A: Aqui tem 9 e no outro tem 10. Professor: Acho que falta arrumar mais coisas. Aluna A: Tem que subir, mudar o y. Professor: Muito bem. Vamos tentar? Se está 200, para quanto vamos mudar? Aluna A: 240 ou 160. Tenta 160.</p>

 <pre> clear paths new pattern set size to 20 set rotation to 0 set instrument to marimba set x to 300 set y to 0 set angle to 180 set step size to 40 set stamp rate to 5 add yellow star to pattern add yellow star to pattern add blue circle to pattern add blue circle to pattern repeat 4 times do repeat 15 times do stamp </pre>		<p><i>Aluna B: Foi.</i> <i>Professor: Acho que ainda não. E se mudássemos a ordem dos elementos? Se trocássemos o azul/azul/ amarelo/amarelo?</i> <i>Aluna C: Coloca então amarelo/amarelo/azul/azul.</i> <i>Professor: Sugiro alterar apenas o padrão 1, voltar para repetir 5 vezes e modificar o y novamente. Mantém, azul/azul/ amarelo/amarelo, repeat 5 e y 200.</i> <i>Aluna A: Deu certo!</i> <i>Professor: Notem que 2 linhas se sobrepõem agora.</i> <i>Aluna B: Ufa!</i></p>
---	---	---

Fonte: Dados da pesquisa.

Todas as atividades da Tarefa 2, fizeram emergir habilidades referentes à coleta de dados, análise de dados, simulação e automação, assim como as atividades da Tarefa 1. O desenvolvimento da Atividade 2 também possibilitou a decomposição, pois na interpretação dos problemas propostos nessa atividade as participantes buscaram analisar os padrões de forma geral e em partes menores que seriam construídas por programas diferentes. Assim, essa atividade também explorou a paralelização, pois as participantes tiveram que se apropriar e utilizar de mecanismos para a execução de diversos programas simultaneamente para a resolução dos problemas propostos.

A Atividade 3 introduziu a ideia de automação em um nível diferenciado, proporcionado pelo *nested loop*. As participantes conseguiram se apropriar em partes dessa forma de automação, contudo não entenderam em profundidade o algoritmo responsável pela execução do *nested loop*. Isso se verifica quando não conseguem realizar a depuração na construção do padrão do item c da Atividade 3.

No desenvolvimento da Atividade 4, que englobou os diferentes conceitos computacionais explorados anteriormente, mediante a uma execução inesperada do programa, as participantes analisaram e buscaram estratégias de resolução do problema. Esse tipo de ação se relaciona à abstração, pois, em meio a adversidade encontrada, utilizaram mecanismos de reflexão e planejamento como forma de focar nos elementos essenciais do problema. Nessa atividade, se verificou a ocorrência das habilidades de decomposição e paralelização, pois a interpretação e execução do padrão requeria tais ações.

De forma longitudinal, considerando a realização das atividades pelas estudantes no quinto ano e no sexto ano, consideramos que houve avanços em relação à complexidade dos processos de pensamento realizados. Na atividade **Extensões** da

Tarefa 1, por exemplo, as participantes apresentaram dificuldade de assimilação do aplicativo utilizado. Mas no desenvolvimento da Tarefa 2, elas foram eficientes na realização da maioria das atividades propostas, demonstrando maior autonomia no uso do aplicativo durante a resolução dos problemas.

5 CONSIDERAÇÕES FINAIS

Neste artigo descrevemos a condução de duas tarefas matemático-artístico-computacionais, que exploraram a temática **padrões e álgebra**. As tarefas foram realizadas por um grupo de quatro estudantes em dois momentos: quando cursavam o quinto ano e o sexto ano do Ensino Fundamental. Utilizando a lente analítica das ações de aprendizagem (VALENTE, 2003), conseguimos identificar os diferentes momentos da interação das estudantes com os aplicativos explorados, dando destaque para os processos de reflexão e depuração. Nesses momentos, observamos também o papel do professor, que se torna mediador, não apenas instrutor, garantindo autonomia das estudantes no processo de aprendizagem.

Do ponto de vista pedagógico, é importante destacar que a natureza computacional das tarefas é significativamente diferente. O aplicativo utilizado na Tarefa 1, por um lado, oferece uma quantidade limitada de *inputs*, mas por outro, explora diversificados aspectos artísticos como sons, cores, padrões e corporeidade. A Tarefa 1 fomenta a exploração da Tarefa 2 enquanto uma extensão, mas as alunas apresentaram dificuldade para explorá-la naquele momento (Anos Iniciais do Ensino Fundamental). A Tarefa 2 permite a descrição de um número maior de comandos e sua natureza artística é diferenciada em relação à Tarefa 1. Na situação analisada, a Tarefa 1 ofereceu meios para exploração de aspectos artísticos específicos, principalmente musicais.

Além de descrever o processo de interação aprendizes-computador, verificamos que as atividades propostas fizeram emergir algumas habilidades do pensamento computacional, cujo desenvolvimento possibilita ao sujeito a construção de capacidades cognitivas no enfrentamento de problemas nos âmbitos virtual e real. Em todas as tarefas, houve indícios da ocorrência das habilidades de coleta de dados, análise de dados, simulação e automação. Na Tarefa 2, cujo desenvolvimento requeria maior complexidade de pensamento, além dessas habilidades, sinais das habilidades de decomposição do problema, paralelização e abstração foram notados. Embora ainda com algumas dificuldades conceituais, as alunas tiveram melhor performance no desenvolvimento da

Tarefa 2 e de maneira geral, os aspectos do pensamento computacional que emergiram nessa tarefa possuem maior complexidade pedagógica, havendo coerência com nível de ensino no qual foi investigada.

A exploração de padrões de repetição por meio de tarefas matemático-artístico-computacional ofereceu oportunidade de utilização e construção de diversos conceitos matemáticos e computacionais como o conceito de números inteiro, ângulos, *loop*, entre outros. Além disso, permitiu a emergência de habilidades essenciais em processos de resolução de problemas que vão além da Matemática Escolar e da atividade de programação de computadores. Essa exploração também possibilitou a emergência de elementos essenciais do desenvolvimento da capacidade de **aprender a aprender**: autonomia e aprendizagem com os próprios erros.³

REFERÊNCIAS

- BARR, V.; STEPHENSON, C. Bringing Computational Thinking to K-12: What is Involved and What is the Role of the Computer Science Education Community? **ACM Inroads**, New York, v. 2, n. 1, p. 48-54, mar. 2011. Disponível em: <https://www.iste.org/docs/nets-refresh-toolkit/bringing-ct-to-k-12.pdf?sfvrsn=2>. Acesso em: 16 ago. 2018.
- BICUDO, M. A. V. Pesquisa em educação matemática. **Pro-posições**, Campinas, v. 4, n. 10, p. 18-23, mar. 1993. Disponível em: <http://mail.fae.unicamp.br/~proposicoes/textos/10-artigos-bicudomav.pdf>. Acesso em: 16 jul. 2017.
- BRASIL. **Base Nacional Comum Curricular**. Brasília: MEC/Secretaria de Educação Básica, 2017.
- ENGELHARDT, P. V.; CORPUZ, E. G.; OZIMEK, D. J.; REBELLO, N. S. The teaching experiment- what it is and what it isn't. **AIP Conference Proceedings**, [s. l.], v. 720, n. 1, p. 157-160, 2004. Disponível em: <https://aip.scitation.org/doi/abs/10.1063/1.1807278>. Acesso em: 2 out. 2019.
- GADANIDIS, G. Repeating Patterns – Simulation. **Repeating Patterns – Introduction**, 2017. Disponível em: <http://researchideas.ca/wmt/c2a0.html>. Acesso em: 18 fev. 2020.
- GADANIDIS, G.; YIU, C. Repeating Patterns + Code + Art **Math Surprise: Conceptual, Emotional, Engaging**, London, v. 1, 201-. Disponível em: <http://researchideas.ca/patterns/>. Acesso em: 18 fev. 2020.
- ISTE; CSTA. **Computational thinking**: Teacher resources. 2. ed. [S. l.]: Computer Science Teachers Association (CSTA), International Society for Technology in Education (ISTE), 2011. Disponível em: https://id.iste.org/docs/ct-documents/ct-teacher-resources_2ed-pdf.pdf?sfvrsn=2. Acesso em: 28 set. 2018.

³ Agradecemos ao Projeto de Internacionalização da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES - PrInt) e ao Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq - 28323/2018-9) pelo apoio no financiamento da pesquisa.

LU, J. J.; FLETCHER, G. H. Thinking about computational thinking. *In: ACM TECHNICAL SYMPOSIUM ON COMPUTER SCIENCE EDUCATION*, 40., 2009, Chattanooga.

Proceedings... New York: ACM, 2009, p. 260-264, 2009. Disponível em:

<https://dl.acm.org/doi/abs/10.1145/1508865.1508959>. Acesso em: 18 fev. 2020.

MALTEMPI, M. V. Novas Tecnologias e Construção de Conhecimento: Reflexões e perspectivas. *In: CONGRESSO IBERO-AMERICANO DE EDUCAÇÃO MATEMÁTICA*, 5., 2005, Porto. **Anais...** Porto: Universidade do Porto, 2005, p. 1-11. Disponível em:

<http://www.rc.unesp.br/igce/demac/maltempi/Publicacao/Maltempi-cibem.pdf>. Acesso em: 18 fev. 2020.

MANNILA, L.; DAGIENE, V.; DEMO, B.; GRGURINA, N.; MIROLO, C.; ROLANDSSON, L.; SETTLE, A. Computational Thinking in K-9 Education. *In: ANNUAL CONFERENCE ON INNOVATION AND TECHNOLOGY IN COMPUTER SCIENCE EDUCATION*, 19., 2014, Uppsala. **Proceedings...** New York: ACM, 2014. p. 1-29. Disponível em:

<https://dl.acm.org/doi/10.1145/2713609.2713610>. Acesso em: 18 fev. 2020.

PAPERT, S. **Mindstorms**. New York: Basic Books, 1980.

PAPERT, S. **The Children's Machine: Rethinking School in the Age of the Computer**.

New York: Basic Books, 1993.

POWELL, A. B.; FRANCISCO, J. M.; MAHER, C. A. Uma Abordagem à Análise de Dados de Vídeo para Investigar o Desenvolvimento das Idéias Matemáticas e do Raciocínio de Estudantes. **Bolema**, Rio Claro, v. 17, n. 21, p. 1-47, mai. 2004. Disponível em:

<http://www.periodicos.rc.biblioteca.unesp.br/index.php/bolema/article/view/10538>. Acesso em: 12 jun. 2019.

RIBEIRO, L.; FOSS, L.; CAVALHEIRO, S. A. C. Entendendo o pensamento computacional. **arXiv**, Ithaca, p. 1-18, 2017. Disponível em:

<https://arxiv.org/abs/1707.00338>. Acesso em: 16 abr. 2020.

STEFFE, L. P.; THOMPSON, P. W. Teaching experiment methodology: Underlying principles and essential elements. *In: LESH, R.; KELLY, A. E. (Ed.). Research design in mathematics and science education*. Hillsdale: Erlbaum, 2000. p. 267-307.

VALENTE, J. A. In Service Teacher Development Using ICT: First Step in Lifelong Learning. *In: DOWLING C.; LAI, K.W. (Ed.) Information and Communication Technology and the Teacher of the Future*. IFIP – The International Federation for Information Processing, vol. 132. Boston: Springer, 2003. p. 97-108.

WING, J. M. Computational thinking and thinking about computing. **Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences**, London v. 366, n. 1881, p. 3717-3725, 2008. Disponível em:

<https://www.cs.cmu.edu/~wing/publications/Wing08a.pdf>. Acesso em: 17 jan. 2020.

WING, J. M. Computational thinking. **Communications of the ACM**, New York, v. 49, n. 3, p. 33-35, 2006. Disponível em:

<https://saintmaryschool.net/documents/2017/5/Wing%20on%20what%20computational%20thinking%20is%20and%20is%20not.pdf>. Acesso em: 17 jan. 2020.

YADAV, A.; STEPHENSON, C.; HONG, H. Computational Thinking for Teacher Education. **Communications of the ACM**, New York, v. 60, n. 4, p. 55-62, abr. 2017. Disponível em:

<https://m-cacm.acm.org/magazines/2017/4/215031-computational-thinking-for-teacher-education/fulltext>. Acesso em: 11 set. 2018.