

**POLÍGONOS NO SCRATCH: UM OLHAR PARA O PENSAMENTO COMPUTACIONAL
ATRAVÉS DE ABSTRAÇÕES REFLEXIONANTES***POLYGONS IN SCRATCH: A LOOK AT COMPUTATIONAL THINKING THROUGH
REFLECTING ABSTRACTIONS**POLÍGONOS EN SCRATCH: UNA MIRADA AL PENSAMIENTO COMPUTACIONAL
A TRAVÉS DE ABSTRACCIONES REFLECTANTES***KÁTIA COELHO DA ROCHA¹**
MARCUS VINICIUS DE AZEVEDO BASSO²**RESUMO**

O Pensamento Computacional tem ganhado destaque desde 2006, sob diferentes olhares e práticas. Visando contribuir com essa discussão, apresentamos nesse artigo parte de uma pesquisa de doutorado que tem como objetivo identificar as contribuições do Pensamento Computacional, a partir da resolução de problemas investigativos, para promover processos de abstração reflexionante em professores de Matemática. A pesquisa foi realizada com cinco professores de matemática, através de uma formação online e individual, em que eram desafiados a resolver problemas investigativos utilizando a computação como um objeto-de-pensar-com. Os dados foram analisados a partir da teoria de Piaget ([1977], 1995), destacando a abstração reflexionante e suas relações com os objetos-de-pensar-com de Papert (1985). Observamos que, mesmo em professores de Matemática que já possuem conhecimentos mais avançados, o uso do Pensamento Computacional dentro dessa perspectiva provocou novos processos de abstração empírica e reflexionante sobre conceitos matemáticos e computacionais.

Palavras-chave: abstração reflexionante; objetos-de-pensar-com; pensamento computacional.

ABSTRACT

Computational Thinking has gained prominence since 2006, under different perspectives and practices. Aiming to contribute to this discussion, we present in this article part of a doctoral research that aims to identify the contributions of Computational Thinking, from the resolution of investigative problems, to promote reflecting abstractions processes in Mathematics teachers. The research was carried out with five mathematics teachers, through online and individual training, in which they were challenged to solve investigative problems using computing as an object-to-think-with. Data were analyzed based on Piaget's theory ([1977], 1995), highlighting reflecting abstraction and its relations with Papert's objects-of-think-with (1985). We observed that even in Mathematics teachers who already have more advanced knowledge, the use of Computational Thinking within this perspective provoked new processes of empirical abstraction and reflecting on mathematical and computational concepts.

Keywords: reflecting abstraction; objects-to-think-com; computational thinking.

¹ Doutora em Informática na Educação pelo Programa de Pós-Graduação em Informática na Educação da Universidade Federal do Rio Grande do Sul (PPGIE/UFRGS). Professora no Núcleo de Tecnologia Educacional Municipal (NTM) da Secretaria Municipal de Educação de São Leopoldo/RS. E-mail: katiacoelho-rocha@gmail.com. ORCID: <https://orcid.org/0000-0002-7433-4471>

² Doutor em Informática na Educação pelo Programa de Pós-Graduação em Informática na Educação da Universidade Federal do Rio Grande do Sul (PPGIE/UFRGS). Professor Titular do Instituto de Matemática e Estatística da UFRGS. Diretor do Centro Interdisciplinar de Novas Tecnologias na Educação (CINTED) e coordenador o Centro de Formação Continuada de Professores (FORPROF) da UFRGS. E-mail: mbasso@ufrgs.br. ORCID: <https://orcid.org/0000-0002-2312-9056>

RESUMEN

El Pensamiento Computacional ha ganado protagonismo desde 2006, bajo diferentes perspectivas y prácticas. Con el objetivo de contribuir a esta discusión, presentamos en este artículo parte de una investigación doctoral que tiene como objetivo identificar las contribuciones del Pensamiento Computacional, a partir de la resolución de problemas de investigación, para promover procesos de abstracción reflexiva en profesores de Matemáticas. La investigación se realizó con cinco docentes de matemáticas, a través de capacitaciones en línea e individuales, en las que se les desafió a resolver problemas investigativos utilizando la computación como objeto-con-pensar. Los datos fueron analizados con base en la teoría de Piaget ([1977], 1995), destacando la abstracción reflexiva y sus relaciones con los objetos-de-pensar-con de Papert (1985). Observamos que incluso en profesores de Matemática que ya tienen conocimientos más avanzados, el uso del Pensamiento Computacional dentro de esta perspectiva provocó nuevos procesos de abstracción empírica y reflexión sobre conceptos matemáticos y computacionales.

Palabras-clave: abstracción reflexiva; objetos-para-pensar-com; pensamiento computacional.

INTRODUÇÃO

O Pensamento Computacional tornou-se foco de pesquisas e discussões a partir da publicação de um artigo em 2006 por Jeannette Wing, em que ela afirma que todos, independente da área de atuação, deveriam pensar como cientistas da computação (WING, 2006). Aliado a essa publicação, empresas da área da tecnologia passaram a apoiar iniciativas que incentivavam jovens a aprender a programar e a olhar a computação como um conhecimento necessário, diante da carência de mão de obra no mercado de trabalho. Esses eventos instigaram debates e pesquisas na área da Ciência da Computação que influenciaram a inclusão do tema nos currículos escolares pelo mundo.

Desde então, o conceito de Pensamento Computacional vem sendo construído através de debates de diferentes seguimentos, a própria Wing foi reelaborando sua definição sendo que na mais atual afirma que “O Pensamento Computacional é o processo de pensamento envolvido na formulação de um problema e na expressão de sua(s) solução(s) de forma que um computador - humano ou máquina - possa efetivamente executar” (WING, 2014, p. 1). Essa definição aponta que há processos mentais envolvidos durante a solução e a formulação de problemas e a forma de expressá-los pode ser feita entre humanos-humanos e/ou humanos-máquina. Ela destaca que este pensamento não se resume a uma linguagem de programação, mas sim a uma forma de comunicação linguística (linguagem visual, natural, linguagem de programação, entre outras) que seja compreendida e esteja em um formato acessível ao público que se destina.

A falta de uma definição universal e precisa do termo Pensamento Computacional, já apontada por Denning (2017) e Valente (2019), e a falta de uma análise mais detalhada sobre possíveis efeitos do Pensamento Computacional levam a diversos tipos de práticas que fazem uso desse termo como referência. Algumas estão mais engajadas em ensinar conceitos computacionais outras em inserir a computação no dia a dia, porém nem sempre são analisados os processos cognitivos que essas práticas podem proporcionar. Diante disso, o tema central desta pesquisa não é o ensino de conceitos computacionais, mas como eles podem contribuir para o desenvolvimento de abstrações reflexionantes.

O presente artigo se refere a parte de uma pesquisa de doutorado que tem como questão central quais as contribuições do Pensamento Computacional, a partir da resolução de problemas investigativos, para promover processos de abstração reflexionante em professores de Matemática?

Diante da questão norteadora, temos como objetivo geral identificar as contribuições do Pensamento Computacional, a partir da resolução de problemas investigativos, para promover processos de abstração reflexionante em professores de Matemática.

Nas próximas seções desse artigo apresentamos uma análise das diferentes definições de Pensamento Computacional, o referencial teórico ancorado na abstração reflexionante de Jean Piaget e nos objetos-de-pensar-com de Papert. A metodologia de pesquisa também está explicitada, destacando os procedimentos específicos das atividades descritas nesse artigo. Os resultados e discussão dos dados coletados são analisados a partir do referencial teórico apresentado. Ao final retomamos a questão central e os possíveis encaminhamentos da pesquisa.

APROXIMAÇÕES E DISTANCIAMENTOS ENTRE AS DEFINIÇÕES DE PENSAMENTO COMPUTACIONAL

Além de Wing, diversos pesquisadores tanto da área da Ciência da Computação quanto da Educação pesquisam sobre o tema e não há um consenso da definição de Pensamento Computacional o que gera uma variedade de práticas e análises (DENNING, 2017; VALENTE, 2019; GROVER; PEA, 2013; YASAR, 2018; DISESSA, 2018). Durante a nossa pesquisa, confrontamos diversas definições e identificamos que a maioria dos autores associa ao Pensamento Computacional habilidades fundamentais que seriam a base para a construção desse pensamento. Ele está relacionado com habilidades além da computação, portanto algumas delas não traduzem apenas conceitos computacionais. Apesar disso, observamos que há práticas que envolvem uma análise dessas habilidades apenas sob o foco da computação, examinando programas escritos pelos sujeitos (LI *et al.*, 2020).

As habilidades relacionadas ao Pensamento Computacional variam entre os autores, de acordo com Brackmann (2017), pesquisas de diversas instituições sintetizam essas habilidades ou pilares em: Decomposição, Reconhecimento de Padrões, Abstração e Algoritmos. Uma habilidade comum a todos, e a principal delas, é a abstração. Wing (2006), desde o seu primeiro artigo, destaca a importância da abstração para o Pensamento Computacional, pois as “abstrações são as ferramentas mentais da computação” (WING, 2008, p. 2).

Diante das definições pesquisadas identificamos outras semelhanças entre elas e, a partir disso, agrupamos essas definições de acordo com suas características que indicam que o Pensamento Computacional pode ser: programação que gera outras habilidades e reflexões; uma metodologia para a resolução de problemas; uma fusão de outros pensamentos; uma capacidade para usar conceitos computacionais; um gerador de processos cognitivos.

As pesquisas mostram semelhanças entre si, como o reconhecimento de que a programação tem laços estreitos com o Pensamento Computacional, mesmo que ele não seja sinônimo de programação. Identificamos, assim como Li *et al.* (2020) constataram, práticas que estão voltadas estritamente ao ato de programar, argumentando que a prática promove o Pensamento Computacional, porém a própria Wing (2006) admite que pensar como cientista da computação vai além de programar um computador, mas pensar em vários níveis de abstração (WING, 2006). As práticas mais voltadas ao ato de programar podem ser compreendidas sobre dois aspectos relevantes que ocorrem em pesquisas no exterior e aqui no Brasil, o uso de *softwares* de programação para promover o Pensamento Computacional (SHUTE; SUN; ASBELL-CLARKE, 2017) e a condução de estudos por profissionais mais da área da Ciência da Computação do que da Educação (LI *et al.*, 2020).

A resolução de problemas está presente de alguma forma em todas as definições pesquisadas. Wing (2006) considera a resolução de problemas como um ponto central para o desenvolvimento

do Pensamento Computacional, pois é através dos problemas a resolver que o sujeito vai utilizar as ferramentas e conceitos computacionais. Há autores (GROVER; PEA, 2013; ISTE/CSTA, 2011) que em sua definição consideram o Pensamento Computacional como um processo de resolução de problemas, uma espécie de metodologia em que o sujeito usa ferramentas e técnicas computacionais para solucioná-los.

Brackmann (2017), Shute, Sun e Asbell-Clarke (2017) e a SBC (2019) apontam que o Pensamento Computacional é uma capacidade que envolve um conjunto de habilidades que permitem ao sujeito integrar conhecimentos da computação para resolver problemas. Brackmann (2017) e Shute, Sun e Asbell-Clarke (2017) destacam ainda a possibilidade dos diversos contextos em que pode ser aplicado, ampliando o domínio de atuação às diversas áreas. Weintrop *et al.* (2015) também focam na aplicação do Pensamento Computacional, mais especificamente nas áreas da matemática e ciências. Enquanto isso a SBC (2019) considera essas habilidades mais relacionadas ao desenvolvimento de algoritmos, remetendo o Pensamento Computacional e suas práticas de computação no ensino mais próximas da Ciência da Computação do que das diversas áreas de ensino (RAABE; ZORZO; BLIKSTEIN, 2020).

Pensamento Computacional como uma mistura de outros pensamentos é considerada por autores como Hu (2011) e Wing (2008). Pensamento Matemático, Algébrico, Algorítmico, de Engenharia, Crítico, são alguns dos pensamentos que estariam atuando em conjunto para formar o Pensamento Computacional que alinha todos eles na direção da computação. Hu (2011) argumenta que, à luz do Pensamento Crítico, todos os paradigmas de pensamento podem compartilhar elementos comuns e que a mesma habilidade de pensamento pode ser desenvolvida através de diferentes experiências de aprendizagem. Muitas habilidades relacionadas ao Pensamento Computacional também fazem parte do Pensamento Matemático, como o reconhecimento de padrões, a abstração, a generalização, a decomposição, os algoritmos, o pensamento recursivo, abstrato, lógico e processual (HU, 2011; WING, 2006, 2008).

Valente *et al.* (2017) ao analisar diversos autores que definem o pensamento algébrico encontraram intersecções entre o que pertenceria ao pensamento algébrico e ao Pensamento Computacional e concluem que a dificuldade em distinguir esses pensamentos se dá por serem atividades cognitivas complexas e que envolvem processos comuns a esses e a outros tipos de pensamentos. Neste sentido, o Pensamento Computacional inclui habilidades que não são exclusivas a ele, mas com o auxílio da computação, o sujeito pode ampliar suas possibilidades de investigação de fenômenos e do mundo, ampliando seu poder de reflexão (SNEIDER *et al.*, 2014; HU, 2011, WEINTROP *et al.*, 2015).

Com maior evidência percebe-se que Wing (2017), Aho (2011), Voogt *et al.* (2015), Zipitria (2018), Yasar (2018), Li *et al.* (2020), relacionam o Pensamento Computacional com processos mentais que estão articulados aos processos computacionais ou elementos da computação na solução e formulação de problemas. Os autores afirmam que são necessárias mais investigações a cerca deste tema.

As diversas definições apresentam diferentes formas de abordar o Pensamento Computacional, mas todas destacam a emergência do tema no cenário mundial diante da crescente evolução tecnológica. Como educadores e pesquisadores, estamos interessados em fornecer elementos que possam contribuir com a discussão sobre como esse pensamento pode estar presente no ambiente escolar para potencializar a aprendizagem dos estudantes. Assim, propomos um aprofundamento teórico a respeito do tema embasado nas pesquisas de Piaget e Papert que serão apresentadas na próxima seção.

REFERENCIAL TEÓRICO

A abstração é considerada um dos pilares do Pensamento Computacional. Porém, a abstração apresentada em vários estudos nessa área está mais relacionada com o sentido operacional do conceito observando técnicas computacionais que permitem a criação de modelos e programas cada vez mais elaborados. No caso desta pesquisa, observamos como essas práticas computacionais contribuem para processos de construção do conhecimento por parte dos sujeitos, levando-os a abstrações reflexionantes cada vez mais elaboradas.

Diante disso, nosso foco de estudo está nesses processos mentais envolvidos que estão embasados na teoria psicogenética de Jean Piaget, destacando a abstração reflexionante.

A construção do conhecimento, de acordo com Piaget ([1977],1995), ocorre a partir de assimilações e acomodações que levam a tomadas de consciência, responsáveis por ações reflexivas. O sujeito assimila o objeto a partir de abstrações empíricas, retirando propriedades físicas do objeto ou aspectos materiais da sua própria ação, como empurrões e movimentos. Essas abstrações são realizadas a partir de instrumentos de assimilação, que são resultado de esquemas construídos anteriormente. O objeto também pode ser assimilado pelo sujeito a partir de coordenações de ações que são retiradas por abstração reflexionante.

A abstração reflexionante envolve um processo de abstração mais elevado, em que são consideradas características não observáveis do objeto, que são fonte de atividades cognitivas do sujeito (esquemas, coordenações, operações, estruturas). Mesmo que o sujeito se apoie em coordenações, essas coordenações e o processo de reflexionamento podem permanecer inconscientes, ou podem gerar tomadas de consciência e conceituações variadas (PIAGET, [1977], 1995). Há um caso particular de abstração reflexionante, a abstração pseudoempírica, em que a propriedade que é abstraída não é a do objeto, ela é fruto de modificações, por parte do sujeito, que enriquece, através da sua ação, com propriedades que retira de suas coordenações.

A abstração reflexionante envolve dois processos complementares, o reflexionamento que é uma projeção para um patamar superior do que foi retirado de um patamar inferior; e a reflexão que é um ato mental em que o sujeito reconstrói e reorganiza no patamar superior o que estava em um patamar inferior. Ao reorganizar esse patamar, produz uma nova construção juntamente com outras que poderá ser transferida para patamares ainda mais elevados, levando a um processo cada vez mais elevado e sem fim, independentemente do nível de desenvolvimento do sujeito.

Observando a definição de Pensamento Computacional e o pilar da abstração associado a ele identificamos que esse pensamento envolve processos de abstração empírica e reflexionante propostos por Piaget, mesmo que eles não sejam analisados nos estudos. Por isso, esta pesquisa visa analisar as abstrações reflexionantes desencadeadas por processos computacionais que podem contribuir para tomadas de consciência. Os processos computacionais são considerados, nesse estudo, sob a perspectiva dos objetos-de-pensar-com de Papert.

Ao criar a linguagem LOGO no final da década de sessenta, Seymour Papert e sua equipe, proporcionaram um ambiente de programação mais simples, de forma que as crianças pudessem usar o computador para criar, ampliando suas possibilidades de aprender, de pensar e crescer, destacando o potencial do desenvolvimento cognitivo e emocional provocado por esse ambiente (PAPERT, 1985).

Enquanto programa o computador o sujeito “ensina o computador” a realizar uma determinada tarefa e durante esse processo de descrever uma sequência de ações é levado a explorar seu próprio pensamento, ampliando o poder de reflexão sobre o seu pensamento, desencadeando abstrações.

Dessa forma, o computador é usado como um “objeto-de-pensar-com”, um instrumento de pensamento que é uma junção de cultura, conhecimento implícito e identificação pessoal (PAPERT, 1985).

A expressão “objeto-de-pensar-com” usada pelo autor faz referência ao computador como um objeto de pensamento que permite ao sujeito utilizá-lo para investigar conceitos, analisar situações, fazer hipóteses, testar teorias, auxiliando em processos de abstração. Assim, o computador permite que o sujeito possa expressar seus pensamentos, ampliando suas possibilidades de interação com os conceitos, servindo “[...] às crianças como instrumentos para trabalhar e pensar, como meios para realizar projetos, como fonte de conceitos para pensar novas ideias” (PAPERT, [1993], 2008, p. 158).

Papert (1985) também utiliza o termo Pensamento Computacional, ao relacionar as habilidades mentais que são desenvolvidas durante os processos de programação. Ele preocupa-se com o exame mais minucioso das construções mentais, que são aprimoradas pelas ideias e modelos que surgem em atividades em LOGO e não com a programação em si, ao seu resultado expresso na tela. Ao programar a tartaruga LOGO, por exemplo, o sujeito utiliza uma linguagem específica para se comunicar com o computador, criando modelos computacionais, que refletem suas formas de pensar sobre aquele problema proposto.

Usar o computador como um objeto-de-pensar-com é o que, na visão desta pesquisa, dá sentido ao termo Pensamento Computacional. Porém, pensar computacionalmente dentro desta perspectiva implica também em pensar a escola como um espaço investigativo, que valoriza uma aprendizagem natural, em um ambiente real ou em um micromundo, em que é possível criar, experimentar hipóteses, testar teorias, oportunizando uma aprendizagem em que o sujeito possa se envolver “tateando” as situações (PAPERT, 1985). Aprender é um processo ativo e construtivo, em que o sujeito está em constante ação. Nesse sentido, definimos que o Pensamento Computacional “são os processos de pensamento que utilizam elementos computacionais como objetos-de-pensar-com para contribuir nos processos de abstração reflexionante do sujeito” (ROCHA, 2023, p. 45).

Dentro dessa perspectiva, o desenvolvimento de algoritmos por si só não é suficiente para o desenvolvimento do Pensamento Computacional. O foco não está no ensino de conceitos computacionais ou em um código, mas nas atividades a serem realizadas pelo sujeito que devem lhe permitir investigar conceitos, de diferentes áreas, com o suporte de elementos da computação desencadeando processos de abstração reflexionante. Entendemos que essas atividades precisam pertencer a uma classe específica de problemas, os problemas investigativos.

No caso da matemática, onde centra-se essa pesquisa, uma investigação matemática desenvolve-se em torno de um ou mais problemas e envolve “descobrir relações entre objetos matemáticos conhecidos ou desconhecidos, procurando identificar as respectivas propriedades” (PONTE; BROCARD; OLIVEIRA, 2006, p. 13). Os problemas investigativos não são exercícios do tipo: calcule, siga o modelo. Eles exigem mais do que um resultado, levam o sujeito a questionar, fazer relações, compartilhar ideias, identificar propriedades, criar representações para a situação. Entendemos que os problemas investigativos aliados a elementos computacionais, sejam em ambientes plugados (que usam o computador ou alguma tecnologia) ou desplugados (sem uso de tecnologia), podem oferecer diferentes contextos para os sujeitos experimentarem os conceitos matemáticos, gerando os desequilíbrios necessários para que novas abstrações reflexionantes aconteçam (PIAGET, [1977], 1995; BONA, 2021, 2022; BONA; BOBSIN; KOLOGESKI, 2020).

MÉTODOS

A pesquisa foi implementada com cinco professores de matemática de uma rede municipal de ensino do Vale do Rio dos Sinos, no Rio Grande do Sul (Brasil). Todos eles possuem licenciatura em matemática, sendo que dois possuem pós-graduação na área (um é especialista e outro mestre) e atuam como professores de matemática com estudantes dos anos finais do ensino fundamental. Destacamos que os participantes dessa pesquisa são referenciados, no decorrer do texto, com nomes fictícios inspirados em cientistas da área da Matemática e da Ciência da Computação. São eles: Ada Lovelace, Annie Easley, Grace Hopper, Mary Jackson e Charles Babbage.

Três professores, Annie, Grace e Charles, afirmavam ter maior afinidade com o uso da tecnologia e já conheciam algum *site* ou *software* que envolvia programação, de acordo com eles, em níveis exploratórios. As demais professoras, Ada e Mary, afirmavam ter pouca afinidade com a tecnologia. Um dado interessante é que essas duas professoras relataram que já haviam utilizado algum recurso tecnológico em sala de aula com atividades que permitiam que os estudantes explorassem conceitos, usando a tecnologia como um objeto-de-pensar-com. Já os demais professores relataram um uso em que a tecnologia serviu para ilustrar algum conceito já desenvolvido anteriormente em sala de aula.

Os sujeitos da pesquisa participaram de uma formação *online* e individual, através de videoconferência, via plataforma Google Meet. Cada participante teve em média, para a atividade descrita nesse estudo, dois encontros individuais de uma hora e meia com a pesquisadora em que eram desafiados a resolver os problemas investigativos em que a computação pudesse ser usada como um objeto-de-pensar-com. Durante cada encontro a pesquisadora solicitava que o participante compartilhasse a sua tela enquanto este resolvia o problema proposto. Todos os encontros realizados foram gravados permitindo reconstruir os momentos de interação e as construções realizadas pelos participantes, suas reflexões e respostas às intervenções.

A fim de investigar com profundidade cada participante, visando identificar possíveis abstrações reflexionantes ocorridas durante a resolução dos problemas investigativos, utilizamos observações sistemáticas e entrevistas amparados no Método Clínico de Jean Piaget, além do acompanhamento da produção de cada um durante a resolução do problema investigativo proposto. Esse conjunto de dados, composto por múltiplas fontes, permitiu formar uma série de evidências de cada indivíduo, em que a retirada de exemplos desses casos serviu para formar o estudo que abrangeu todos os participantes (YIN, 2015).

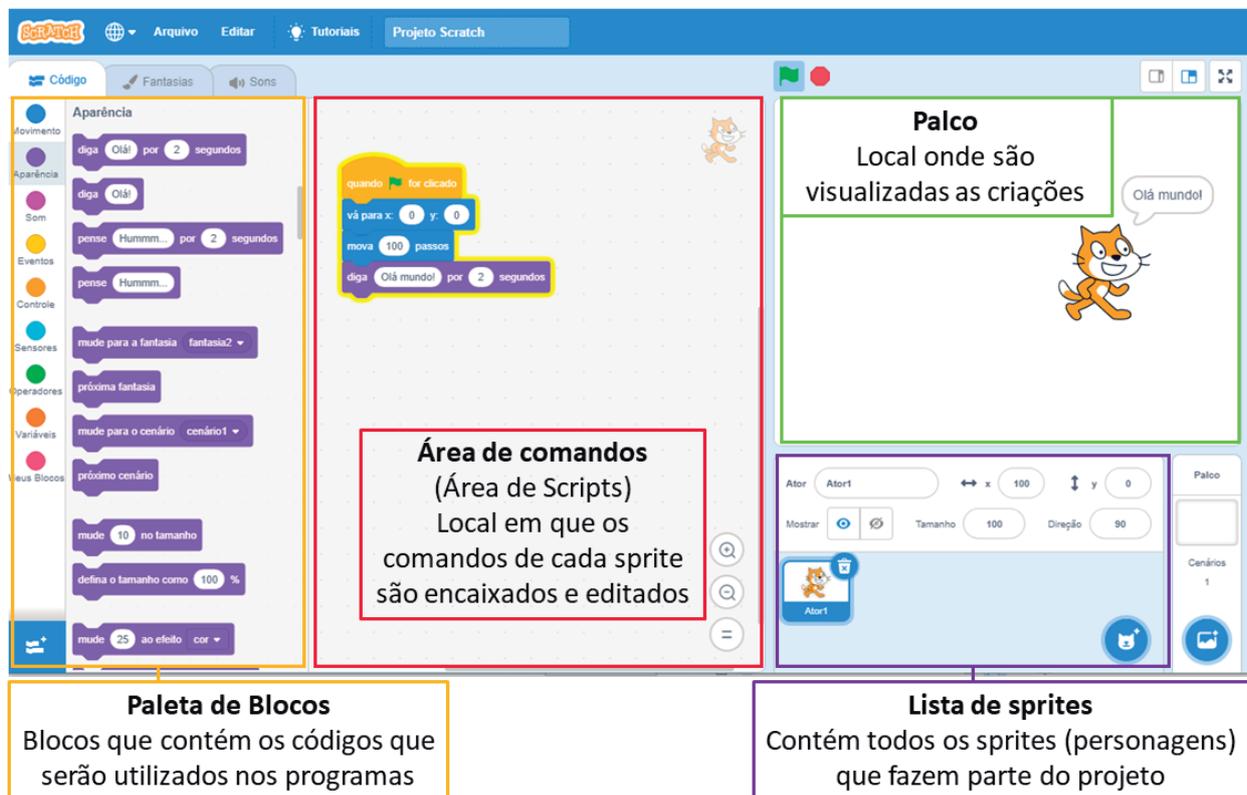
Investigar possíveis abstrações reflexionantes ocorridas durante toda a formação exigiu um acompanhamento sistemático em que o Método Clínico permitiu que a pesquisadora pudesse guiar sua entrevista clínica, já que esse método “[...] ajuda a desvendar como funciona sua mente mediante suas explicações e sua ação [...]” (DELVAL, 2002, p. 15). Diante disso, a pesquisadora deveria estar atenta à resolução do problema, aos gestos, ações e falas do participante, perguntando sobre o significado da ação e qual sua relação com as capacidades mentais do participante. Para compreender, fazia questionamentos que lhe permitiam reconstruir o modelo mental de cada indivíduo, olhando para cada um (DELVAL, 2002).

A resolução dos problemas investigativos explorou o desenvolvimento de algoritmos através da ferramenta *Scratch*. A escolha do *software* se deu em virtude de suas semelhanças com o LOGO, por explorar uma linguagem de programação visual, sendo de fácil acesso aqueles que nunca programaram.

O *Scratch* permite uma mistura de diferentes recursos em um mesmo projeto: sons, animações, gráficos, imagens, sensores, entre outros. Esse diferencial do *software* o torna versátil e atrativo,

facilitando a introdução aos conceitos de programação. A figura 1 ilustra a tela inicial do *Scratch* destacando suas quatro áreas principais.

Figura 1 - Interface do Scratch com suas áreas principais.



Fonte: Elaboração pessoal.

O *software* utiliza uma linguagem de programação em blocos, em que os comandos já estão prontos, evitando erros de sintaxe, comuns em linguagens usuais de programação. Através dos blocos de comandos o usuário cria seus programas combinando blocos, encaixando-os como peças de Lego. Na figura 1, acima, é possível visualizar que na área de comandos há um programa em que o ator (sprite) executa a seguinte ação quando o usuário clica na bandeira verde: vai até a posição central do palco, em que as coordenadas x,y são (0,0), depois se move cem passos para a direita e finaliza mostrando um balão de fala com a frase “Olá Mundo!” por dois segundos. Nota-se que os comandos são executados na sequência em que foram encaixados, já que essa é uma linguagem sequencial. O *Scratch* também admite uma programação paralela, ou seja, que eventos diferentes sejam executados simultaneamente.

As atividades apresentadas nesse artigo foram realizadas, em média, depois de quatro encontros com os participantes, sendo que nos encontros anteriores já haviam resolvido problemas investigativos desplugados e com o *Scratch*. Em um deles tiveram a oportunidade de explorar os blocos de movimento absoluto e relativo do *software*.

As atividades foram elaboradas considerando que os participantes não teriam conhecimento de práticas que envolviam Pensamento Computacional, ferramentas da computação e programação. Buscamos aproximar os participantes dos conceitos da computação a partir de problemas investigativos que pudessem

ser aplicadas com os estudantes, que possuíssem um caráter mais aberto e que não, necessariamente, se buscasse uma resposta final única. Desta forma, acreditávamos que elementos da computação (recursos computacionais e técnicas da computação, com destaque para a programação) aplicados poderiam contribuir para a compreensão do problema, ampliação das capacidades de resolução e análise, permitindo que a computação fosse utilizada como um objeto-de-pensar-com para potencializar processos de abstração reflexionante (ROCHA, 2023).

As atividades descritas nesse artigo consistiam em realizar o desenho, no *Scratch*, de dois polígonos regulares: quadrado e triângulo equilátero. Para realizar as atividades cada participante deveria abrir o *software* sem qualquer programação prévia e criar o seu algoritmo, utilizando o recurso da caneta, para que o ator deixasse na tela o rastro do seu movimento. Após foi apresentado o recurso “criar blocos” do *software* que permitia ao usuário criar subprocedimentos que poderiam ser utilizados em outros programas. Todos os subprocedimentos criados nessa atividade foram adicionados na “mochila” do *Scratch* para que pudessem ser reaproveitados em outras atividades.

Ao concluir a construção cada participante foi desafiado a pensar em um conjunto de procedimentos que desenhasse um polígono regular qualquer, ou seja, um bloco que permitisse reproduzir qualquer polígono regular.

RESULTADOS E DISCUSSÃO

Ao criar os dois primeiros programas do quadrado e do triângulo, acreditávamos que os professores os utilizariam como base para observações e generalizações, auxiliando na criação do último programa. Após a construção de cada um dos programas a pesquisadora mostrou aos participantes como criar um bloco com a função executar um quadrado, por exemplo. Essa função da programação também é conhecida como programação estruturada em que um programa é dividido em subprogramas (procedures ou subprocedimentos) para simplificar o processo de criação, compreensão e manutenção do programa (MARJI, 2014).

Abaixo serão analisadas cada uma das três criações dos professores propostas nessa atividade. As falas apresentadas no decorrer dessa seção são resultado da transcrição de cada encontro. Em alguns momentos, durante a fala, o professor expressava alguma reação ou movimentava algum bloco no *software*, essas ações foram registradas ao lado da fala entre parênteses.

Quadrado

Ao iniciar a atividade cada professor abriu o *Scratch* sem nenhuma construção anterior ou blocos de início. Apesar da proposta envolver apenas a criação do quadrado, os professores iniciaram sua construção estabelecendo um comando de início que envolvesse uma posição e orientação para o seu ator, quatro deles utilizaram o evento da bandeira verde e um deixou, em sua área de códigos, os blocos de reinício soltos para clicar quando desejasse. A figura 2 mostra os comandos das professoras Mary e Ada, que ilustram essa necessidade de marcar um início para o programa.

Figura 2 - Conjunto de códigos de início - professoras Mary e Ada.



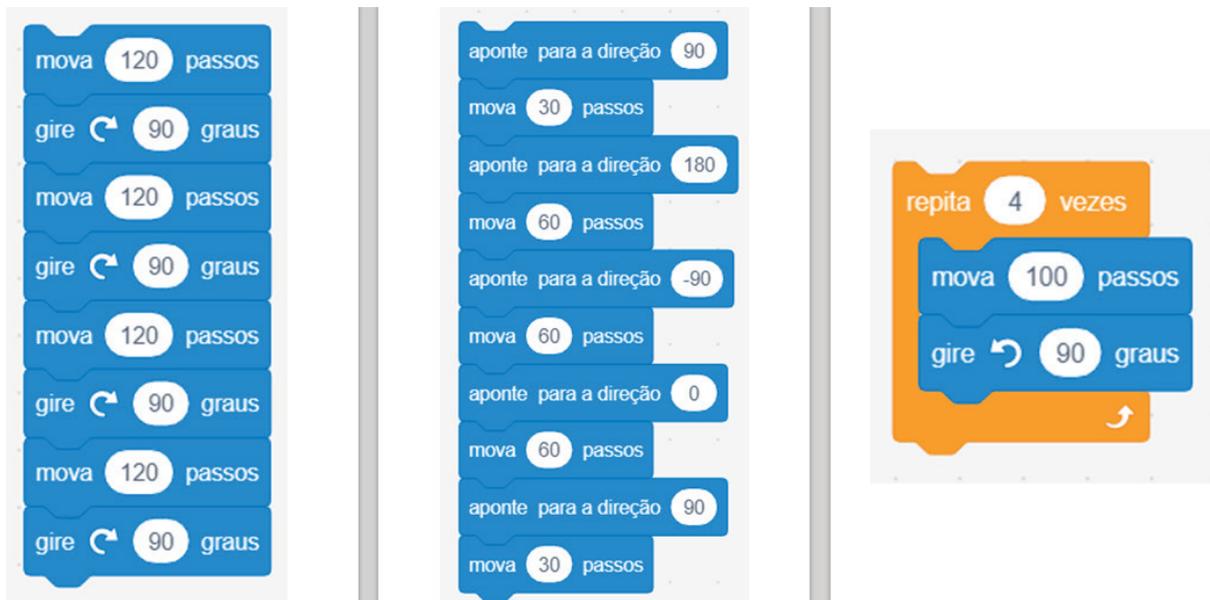
Fonte: Elaboração própria, dados coletados.

Os blocos de início dos professores consideraram a necessidade de estabelecer uma orientação para o ator, indicando suas abstrações a respeito de como o ator se movimenta e, no caso do *Scratch*, dependendo desse movimento a orientação anterior é referência para os demais movimentos. A maioria dos professores também considerou necessário estabelecer uma posição através de uma coordenada (x,y), a professora Mary não determinou essa posição, mas a fazia manualmente arrastando o ator pelo palco. Ao criarem um conjunto de blocos de início, os professores demonstraram abstrações a respeito da concepção de programa, ou seja, ele não só executa um conjunto de ações, ele as executa a partir de um conjunto de condições que são estabelecidas pelo programador. Neste caso, conceitos da computação (condicional, programa) e da matemática (coordenadas, orientação espacial) estão articulados, e entram em ação nos programas dos professores de forma natural, evidenciando o potencial criado por esse micromundo em que o *Scratch* é um objeto-de-pensar-com (PAPERT, 1985).

A estrutura do código quanto a ordem que cada bloco deveria ser encaixado foi explorada por abstrações empíricas e reflexionantes, em que seus efeitos eram visualizados e acompanhados de observações dos professores. Como é o caso do bloco “apague tudo” em que a sua posição no início do código implicava em não deixar o palco totalmente “limpo”, após análises da sequência elaborada juntamente com seus efeitos na tela implicavam em um novo posicionamento deste bloco no final do código.

A elaboração do código que desenhava o quadrado envolveu o uso dos blocos “mova__passos” e “gire__graus” para a maioria dos professores, apenas um professor optou pelo uso dos blocos “mova__passos” e “aponte para a direção”. A figura 3 apresenta o código de alguns professores.

Figura 3 - Código do quadrado dos professores Mary, Charles e Ada.



Fonte: Elaboração própria, dados coletados.

Os três programas apresentados acima, quando executados, desenharam um quadrado. Os programas das professoras Mary e Ada são semelhantes, porém Ada trouxe uma generalização para o código, que surgiu a partir de uma abstração reflexionante da repetição dos blocos. Já a estrutura criada pelo professor Charles não admite a mesma generalização, porque o bloco “aponte para a direção” não considera a orientação anterior do ator, implicando em estabelecer, a cada etapa do código, uma nova direção. Observamos também que o primeiro bloco “mova” desenha apenas a metade do lado do quadrado, sendo necessário concluir esse lado ao final do programa. Uma generalização, para o conjunto do professor Charles implicaria mais elementos e processos de abstração do que apenas acrescentar uma repetição como a realizada pela professora Ada.

Durante a elaboração dos códigos, as professoras Ada, Annie e Grace já expressaram que os blocos seriam repetidos e que poderiam usar algum bloco de repetição, além de montarem partes do programa sem a necessidade de testá-lo. Esses fatos podem indicar patamares de reflexionamento, já que demonstraram conceitualização e construíram uma representação coordenada dos movimentos a serem executados (PIAGET, [1977], 1995). Ao finalizar o código, Mary também indicou esses patamares, pois evidenciava o padrão de repetição dos blocos em sua fala, mas como não conhecia o bloco “repita__vezes” não o usou (após deixar claro a concepção da repetição a pesquisadora lhe apresentou o bloco). Outro destaque de reflexionamento dessa professora foi quanto à orientação do seu ator ao finalizar o código, pois nesse momento o código não contemplava um giro no final e, segundo ela:

Mary: Ele fez o quadrado só não voltou para a posição inicial porque faltou um giro. (Acrescenta o bloco “gire 90 graus”)

Durante essa fala Mary evidenciou sua conceitualização sobre como ocorreu o movimento, identificando como aprimorar a orientação final do ator, sem a necessidade de testar os blocos, ou seja, trouxe uma representação em pensamento do que anteriormente era feito apenas na ação (MONTANGERO; NAVILLE, 1998). Notamos que aqui está o Teorema do Giro Completo da Tartaruga: “Se uma tartaruga percorre um caminho ao redor do perímetro de qualquer área e termina no mesmo estado em que começou, então a soma total de todos os giros será de 360 graus” (PAPERT, 1985, p. 101). Apesar de Mary e os demais professores terem incluído o giro no final para que o ator estivesse com a mesma orientação do início do programa, não há dados que levem a concluir, até esse momento, que a inclusão desse bloco era fruto de um processo de tomada de consciência do teorema proposto por Papert.

Transformar o programa do quadrado em um subprocedimento trouxe aos professores mais um elemento da computação, expandindo suas possibilidades de criação e de exploração da programação paralela, conforme expressou a professora Ada:

Ada: É estranho essa coisa de usar blocos separados. Eu achava que tinha que ser tudo num comando só. Agora eu vi que não precisa, que as coisas acontecem em momentos diferentes.

A criação dos comandos para reinício também trouxe ao programa uma programação paralela, mas a criação do subprocedimento deixou isso em evidência. Pensar em subprocedimentos proporciona ao sujeito uma forma de pensar mais prática e conceitual, lhe permitindo pensar em pequenas partes, subdividindo o programa. Desta forma, facilita testagens, a localização de possíveis *bugs* e a construção de um subprocedimento que executa toda a ação desejada (PAPERT, 1985). Após a construção do subprocedimento do quadrado, os professores passaram a incluí-lo como um novo objeto-de-pensar-com, tentando usá-lo com outros blocos, criando desenhos aleatórios na tela para testá-lo.

Triângulo Equilátero

A construção do triângulo equilátero iniciou novamente com todos os professores estabelecendo um conjunto de comandos para o reinício do programa, partindo de comparações com a construção anterior, sem a necessidade de testar a montagem de cada etapa.

Ao criar a sequência de comandos que desenhava o triângulo equilátero os professores optaram pelos blocos “mova__passos” e “gire__graus”. O giro inicialmente era o de sessenta graus, mas após algumas testagens com esse valor e processos de abstração pseudoempírica, os professores externavam seus reflexionamentos, como pode ser visto nos trechos das falas abaixo:

Ada: Ah não é 60 não! (coloca o 120 e testa)

Ada: Quando eu cliquei no 60 ele inclinou para a direita e se fosse esse comando ele ia abrir e ia fazer um ângulo aberto, não ia formar o meu triângulo. Eu queria que ele retornasse em direção à esquerda para poder formar o triângulo. Ai eu fui pensando e olhando a posição que ele estaria e o que eu precisava acrescentar para dar o formato do triangulo. Eu queria na verdade que os ângulos internos tivessem 60.

Charles: Que estranho! O que aconteceu aqui? (testa novamente)

Charles: Já entendi! (altera para 120 graus)

*Pesquisadora: No início tu tinhas colocado 60, depois de observar alterou para 120.
Charles: Sim ele faz o giro, mas ele sempre toma como referência o ponto inicial.
Aqui no giro ele tá pegando o ângulo externo.*

As falas registradas acima indicam os processos de abstração reflexionante dos professores em que retiraram de seus patamares anteriores o conceito de giro e a concepção de giro do *software* para projetar em outro patamar uma nova construção, que resultou de uma abstração reflexionante sobre o conceito de ângulo (BECKER, 2012). Desta forma, foram necessárias reorganizações mentais, ou seja, uma reflexão sobre o giro do ator que se referia ao ângulo externo, permitindo que alterassem os demais valores de giro sem a necessidade de novas testagens. As falas dos professores expressaram também sua conceitualização sobre o funcionamento do *software* relacionando-o com o conceito matemático presente nessa construção, demonstrando as conclusões a respeito do ângulo externo que foram mobilizadas pelo uso do *software* que, a partir de sua conexão com a estrutura computacional, serviu como um objeto-de-pensar-com (PAPERT,1985).

Antes de finalizar a construção do programa do triângulo equilátero os professores já apresentavam uma generalização para o conjunto de blocos, conforme pode ser visto nos códigos representados na figura 4.

Figura 4 - Código do triângulo equilátero das professoras Mary e Grace.



Fonte: Elaboração própria, dados coletados.

O código das professoras Mary e Grace reproduzem um triângulo equilátero da mesma forma, mas a professora Grace aplicou o recurso da repetição, generalizando e otimizando o código. Em um primeiro momento, olhando apenas para o conjunto de blocos criado por cada uma, seria possível afirmar que apenas a professora Grace generalizou o problema, porém, a partir do acompanhamento através do método clínico, observou-se que Mary também fez esse processo mentalmente e expressou-o em sua fala:

Mary: No do triângulo eu poderia repetir também, como eu fiz no quadrado.

Observamos que ambas as professoras realizaram essa reflexão a respeito da construção que consistia em uma repetição das ações mover e girar. Mary demonstrou que o recurso de repetição do *software* se tornou um objeto de pensamento, evidenciando seu processo de reflexão sobre reflexões anteriores (PIAGET, [1977], 1995). Neste caso, o acompanhamento da pesquisadora durante o processo de criação da participante e o objetivo proposto nessa pesquisa, foram determinantes para identificar esse patamar de reflexionamento. Ao trazer a computação como um objeto-de-pensar-com essa pesquisa propõe um olhar para esses processos mentais, destacando como essa integração entre sujeito e objeto ampliam seu poder de reflexão, algo que apenas olhando para o código final e seu efeito na tela não seria possível (PAPERT, 1985).

Polígono Regular

Nessa atividade a pesquisadora propôs aos participantes que criassem um bloco que desenhasse um polígono regular qualquer de forma mais automatizada. Esperávamos que essa atividade, a partir dos reflexionamentos provocados pelas construções anteriores, evidenciasse uma generalização para o desenho de polígonos no *Scratch*.

Observamos novamente uma automatização dos professores em relação a necessidade de um programa de reinício, sendo essa a primeira etapa construída. As professoras Grace e Ada reutilizaram o programa criado anteriormente para essa nova situação, através do recurso da “mochila”.

A construção do programa do polígono regular partiu, para as professoras Ada, Annie, Grace e Mary, de uma análise dos programas construídos anteriormente. A partir desse processo elas verificaram como foram construídos o quadrado e o triângulo, trazendo em suas falas elementos muito semelhantes, conforme expressa a transcrição da professora Grace:

Grace: Cada um dos que fiz tem gira e move, o move dá o tamanho, eu teria que ver o giro. Eu preciso saber quantos são os lados para ver o repita e pensar nos ângulos externos. Se aumentar o número de lados o ângulo externo vai diminuir e o de dentro vai aumentar.

A fala da professora Grace expressou seu processo de abstração pseudoempírica, em que enriqueceu o programa com propriedades (PIAGET, [1977], 1995), identificando a relação entre a quantidade de lados e o número de repetições, o valor do ângulo com a quantidade de lados. Essas propriedades também foram destacadas pelas demais professoras, trazendo para a área de comandos os blocos repita, mova e gire. A reflexão da professora Grace de que o valor do ângulo externo diminuía à medida que o número de lados aumentava apresenta um patamar superior de reflexionamento, caracterizado pela reflexão sobre reflexões anteriores. Assim, o que antes estava em um patamar inferior, como um instrumento para o seu pensamento passa a ser um objeto de pensamento (PIAGET, [1977], 1995).

Após criar o conjunto de blocos as professoras Ada, Grace e Mary passaram a investigar a relação do ângulo com o número de lados, algumas delas faziam inicialmente um ou mais testes para outros polígonos, diferentes dos usados nas construções anteriores. Esses testes eram acompanhados de expressões como:

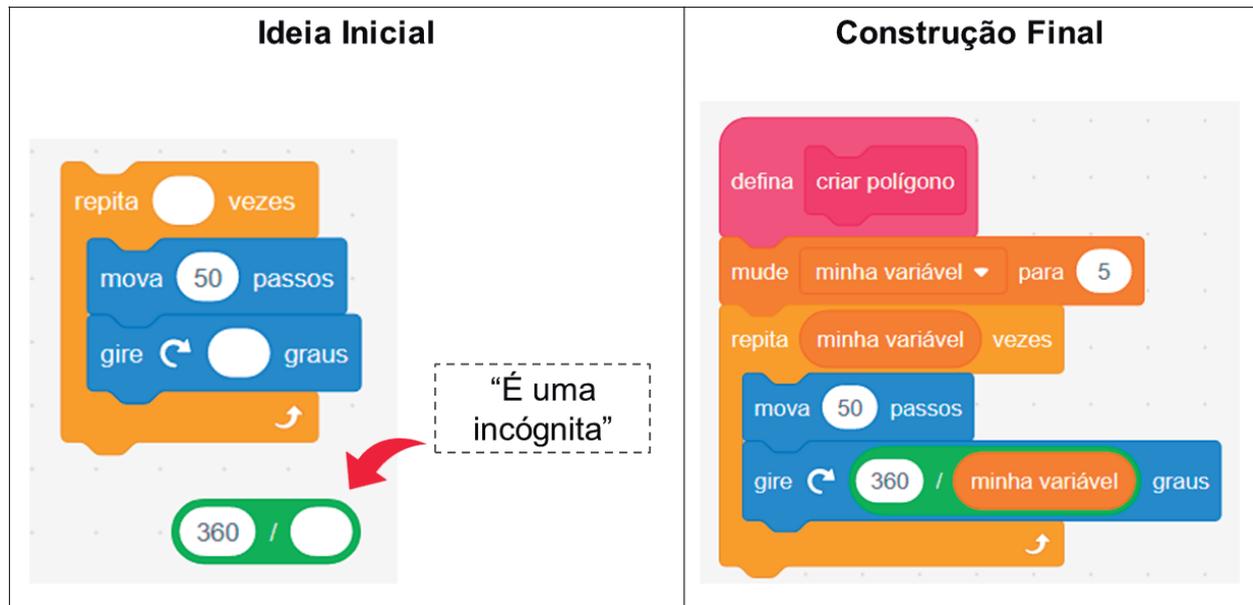
Ada: Todos eles na soma final vão ter que dar 360.

Grace: Ele sempre gira 360 e ele tem que estar na jogada do giro. Tem que dividir o 360 pelo número de lados para ter o ângulo.

Mary: A soma dos ângulos fecha os 360.

As falas das professoras expressaram suas generalizações a respeito do Teorema do Giro Completo da Tartaruga, que foram apoiadas em suas abstrações pseudoempíricas, ou seja, seus reflexionamentos foram concretizados no objeto (ator, programa) e enriquecidos por suas atividades (BECKER, 2012). A partir desse patamar de reflexionamento as professoras demonstraram um novo patamar, uma reflexão, indicando que a quantidade de lados seria um valor variável, necessitando de uma representação que permitisse seu uso em outras situações (a construção de qualquer polígono regular). A figura 5 mostra a tentativa inicial da professora Mary de fazer essa representação e a sua construção final.

Figura 5 - Construção da professora Mary para o polígono regular.



Fonte: Elaboração própria, dados coletados.

Ao iniciar sua construção a professora Mary trouxe o bloco de divisão para a área de comandos, iniciando uma generalização para que o programa calculasse o valor do ângulo. Ela digitou inicialmente o valor do 360 no dividendo e antes de digitar o campo do divisor afirmou:

Mary: Mas isso aqui seria uma variável, o valor por quanto eu iria dividir. É uma incógnita.

Nesse momento a professora expressou seu novo patamar de reflexionamento, evidenciando uma generalização da situação apresentada. A professora seguiu sua generalização tentando inserir no campo a letra “x” e o programa não aceitou, nesse momento a pesquisadora lhe mostrou o bloco de variável. O mesmo ocorreu com as demais participantes.

Os professores Charles e Annie trouxeram uma construção diferente para essa situação, apoiada no cálculo do ângulo interno de cada polígono e não no Teorema do Giro Completo da Tartaruga. Abaixo apresentamos a figura 6 com o programa escrito pela professora Annie e após a transcrição da sua fala que demonstrou as etapas dessa construção.

Figura 6 - Subprocedimento do polígono regular - professora Annie.



Fonte: elaboração própria, dados coletados.

Annie: a soma dos ângulos internos do triângulo é 180, do quadrilátero é 360, é sempre o número de lados menos 2 vezes 180. Dividido esse valor pelo número de lados, isso dá o ângulo interno, o externo é 180 menos essa fórmula. Eu vou ter que colocar essa fórmula ali dentro do bloco do giro (começa a observar os blocos de operadores).

Annie: Mas aí eu vou colocar ao invés do "n" o número de vezes. Como vou fazer essa referência?

Pesquisadora: O que vai ser o "n"?

Annie: O número que a pessoa digitar vai ser o "n". Eu tenho que colocar um comando que assim que a pessoa digitar entre aqui automaticamente nesse repita tantas vezes, é uma variável (abre o conjunto de blocos das variáveis). Mas a variável é a resposta, tem que abrir também uma caixa de resposta como aquele do calendário.

A fala da professora Annie destacou sua forma de pensar o valor do ângulo externo, enriquecendo a situação com propriedades de suas coordenações, demonstrando uma abstração pseudoempírica (PIAGET, [1977], 1995). O professor Charles também trouxe essa propriedade, porém antes de iniciar a montagem dos blocos criou a variável.

Os professores Charles e Annie, trouxeram um processo de pensamento diferente, que talvez possa indicar um caminho mais relacionado à matemática formal e articulado às suas coordenações

anteriores. Já as demais professoras realizaram a construção com base na observação dos movimentos no *software* e em relação ao seu próprio movimento, fazendo um caminho mais próximo do proposto por Papert (1985), desprendendo-se dos conceitos escolares, dando oportunidade à investigação e à exploração de novas formas de pensar (PAPERT, 1985). Nesse caso não evidenciamos que um grupo tenha apresentado uma solução melhor que o outro, mas soluções diferentes e que dizem respeito a sua forma de pensar-com o *Scratch* baseado em suas experiências anteriores, seus processos de abstração dos conceitos matemáticos e do próprio funcionamento do *software*. O que destacamos é que um mesmo problema investigativo, no sentido proposto nessa pesquisa ancorado nas concepções de Bona (2021, 2022); Bona, Bobsin e Kologeski (2020) e Ponte Brocardo e Oliveira, (2006), pode proporcionar diferentes abstrações reflexionantes em que a linguagem computacional é um objeto-de-pensar-com estruturas matemáticas.

CONSIDERAÇÕES FINAIS

O presente artigo apresenta parte dos dados de uma pesquisa de doutorado em que buscamos observar e analisar as contribuições do Pensamento Computacional no desenvolvimento de abstrações reflexionantes em professores de Matemática.

O Pensamento Computacional, apesar de estar na literatura desde a década de sessenta, em 2006 ganhou destaque com o artigo de Wing (2006). A presença da Ciência da Computação nas diversas áreas do conhecimento contribui para diferentes enfoques e abordagens do tema, gerando práticas variadas. Independente do viés metodológico utilizado todos enfatizam a sua importância nos espaços escolares (RAABE; ZORZO; BLIKSTEIN, 2020).

Visando contribuir com essa discussão e trazer elementos que embasam a sua presença no ambiente escolar como forma de expandir as potencialidades dos estudantes propomos nossa visão de Pensamento Computacional. Considerando, de acordo com Piaget ([1977], 1995), que o processo de aprendizagem ocorre a partir de processos de assimilação e acomodação, que levam a tomadas de consciência, identificamos como a inserção de elementos computacionais, a partir de problemas investigativos, podem provocar novos desequilíbrios, levando a abstrações reflexionantes cada vez mais elaboradas, mesmo por professores que possuem um maior domínio de conceitos matemáticos.

Analisamos que inserir a Ciência da Computação sob a perspectiva dos objetos-de-pensar-com de Papert (1985), ancorada em problemas investigativos, pode provocar diferentes abstrações empíricas e abstrações reflexionantes, pois pensar-com essa nova estrutura computacional provocou novas reorganizações matemáticas e computacionais nos sujeitos. Desta forma, desenvolver o Pensamento Computacional dentro da abordagem proposta pode trazer novas formas de pensar e, no caso dos professores de matemática, a formação permitiu um novo contato com os conceitos matemáticos, diferente dos habituais. Esse espaço de formação serviu como um espaço de experimentar, para que o professor pudesse explorar os conceitos de sua área como um aprendiz, Tateando as novas situações e encorajando-o a pensar sobre esse processo com os estudantes.

Assim, desejamos que essa pesquisa possa estar a serviço das discussões sobre a inserção do Pensamento Computacional da educação, além dos programas de formação de professores, visando um olhar para a tecnologia como um objeto-de-pensar-com para ampliar os processos de abstração reflexionante.

REFERÊNCIAS

- AHO, A. Computation and Computational Thinking. *In: Ubiquity Symposium*, jan. 2011, ACM Ubiquity
- BECKER, F. **Educação e construção do conhecimento**. 2 ed., Porto Alegre: Penso, 2012. 200p.
- BONA, A. S. Atividades Desplugadas e Investigativas de Matemática: a abstração enquanto integrante do processo de aprendizagem e da prática do Pensamento Computacional. *In: HABOWSKI, A. C et al. (Org). Sobre as tecnologias no contexto educativo [livro eletrônico]: abordagens comunicativas, autocríticas e 9re0 construtivas*. Santa Maria: Arco Editores, 2021. p. 27-45
- BONA, A. S. A Resolução de Problemas Investigativos de Matemática e o Pensamento Computacional na Escola Básica: um processo complexo de abstração segundo a Teoria de Piaget. **Revista Brasileira de Ensino de Ciências e Matemática**, Passo Fundo, v. 5, edição especial, p. 149-164, 2022
- BONA, A. S.; BOBSIN, R. S.; Kologeski, A. L. Contextualizando a Matemática em Oficinas com o Pensamento Computacional. *In: FREITAS, P. G.; MELLO, R. G. (Org). Educação em foco [recurso eletrônico]: Tecnologias Digitais Inovação em práticas de Ensino*. Rio de Janeiro: e-Publicar, 2020. p. 189-206.
- BRACKMANN, C. P. **Desenvolvimento do Pensamento Computacional através de atividades desplugadas na Educação Básica**. Porto Alegre: PPGIE/UFRGS, 2017. 226p. Tese de Doutorado.
- DELVAL, J. **Introdução à prática do método clínico: descobrindo o pensamento das crianças**. Tradução: Fátima Murad. Porto Alegre: Artmed, 2002.
- DENNING, P. Remaining Trouble Spots with Computational Thinking. **Communications of the ACM**, v. 60, n. 3, p. 33-39, mar. 2017.
- DISESSA, A. Computational Literacy and “The Big Picture” Concerning Computers in Mathematics Education. **Mathematical Thinking and Learning**, vol.20, n.1, p. 3-31, jan. 2018.
- GROVER, S. PEA, R. Computational Thinking in K-12 A Review of the State of the Field. **Educational Researcher**. v. 20, p. 1-6, fev. 2013
- HU, C. Computational Thinking - What It Might Mean and What We Might Do About It. *In: Proceedings of the 16th annual joint conference on Innovation and technology in computer science education*, ITICSE'11, 2011, New York: ACM, 2011 p. 223-227.
- ISTE/CSTA. **Computational Thinking Teacher Resource**. 2 ed., 2011. Disponível em: www.iste.org/docs/ct-documents/ct-teacher-resources_2ed-pdf.pdf?sfvrsn=2. Acesso em: 20 out. 2020.
- LI, Y., SCHOENFELD, A. H., DISESSA, A. A., GRAESSER, A. C., BENSON, L. C., ENGLISH, L. D. e DUSCHL, R. A. Computational thinking is more about thinking than computing. **Journal for STEM Education Reserach**, p. 1-18, maio 2020.
- MARJI, M. **Aprender a programar com Scratch**. Tradução: Lúcia Kinoshita. São Paulo: Novatec, 2014. 284 p.

MONTANGERO, J; MAURICE-NAVILIE, D. **Piaget ou a Inteligência em evolução: Sinopse cronológica e Vocabulário.** Tradução de Tânia Beatriz Iwaszko Marques e Fernando Becker. Porto Alegre: Artmed, 1998. 764 p

PAPERT, S. **A máquina das crianças: repensando a escola na era da informática.** Tradução: Sandra Costa. Porto Alegre: Artmed,[1993] 2008. 224 p.

PAPERT, S. **Logo: computadores e educação.** Tradução: José Armando Valente. São Paulo: Brasiliense, 1985. 253 p.

PIAGET, J. **Abstração reflexionante: Relações lógico-aritméticas e ordem das relações espaciais.** Porto Alegre: Artes Médicas, [1977] 1995. 292 p.

PONTE, J. P. ; BROCARD, J.; OLIVEIRA, H. **Investigações matemáticas na sala de aula.** Belo Horizonte, MG: Autêntica, 2006.

RAABE, A. ; ZORZO,A. F.; BLIKSTEIN, P. **Computação na Educação Básica: fundamentos e experiências.** Porto Alegre: Penso, 2020.316 p.

ROCHA, K. C. **Pensamento Computacional para professores de Matemática: pensar-com abstrações reflexionantes.** Porto Alegre: PPGIE/UFRGS, 2023. 258p. Tese de Doutorado.

SNEIDER, Cary *et al.* Exploring the science framework and NGSS: Computational thinking in the science classroom. **Science Scope**, v. 38, n. 3, p. 10-15, nov. 2014.

SHUTE, V. J.; SUN, C.; ASBELL-CLARKE, J. Demystifying computational thinking. **Educational Research Review**, v. 22, p. 142-158, nov. 2017

VALENTE, J. A.; FREIRE, F. M. P. ; ARANTES, F. L.; ABREU, J. V. V.; AMIEL, T.; BARANAUSKAS; M. C. C. Alan Turing tinha Pensamento Computacional? Reflexões sobre um campo em construção. **Tecnologias Sociedade e Conhecimento**, Campinas, v. 4, Dez. 2017. Disponível em: <http://www.nied.unicamp.br/ojs/> Acesso em: 17 out. 2020.

VALENTE, J. A. Pensamento computacional, letramento computacional ou competência digital? Novos desafios da educação. **Revista educação e cultura contemporânea**, v. 16, n. 43, p. 147-168, 2019.

VOOGT, J.; FISSER, P. ; GOOD, J.; MISHRA, P. ; YADAV, A. Computational thinking in compulsory education. **Towards an agenda for research and practice. Education and Information Technologies**, v. 20, p. 715-728, 2015.

WEINTROP, David; BEHESHTI, Elham; HORN, Michael; ORTON, Kai; JONA, Kemi; TROUILLE, Laura; WILENSKY, Uri. Defining computational thinking for mathematics and science classrooms. **Journal of science education and technology**, v. 25, n. 1, p. 127-147, 2016.

WING, J. M. Computational Thinking. **Communications of the ACM**, v. 49, n. 3, p. 33-35, mar 2006.

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

WING, J. M. Computational Thinking Benefits Society. **Social Issues In Computing**, New York, 10 jan. 2014. Disponível em: <http://socialissues.cs.toronto.edu/2014/01/computational-thinking/>. Acesso em: 15 out. 2020.

YASAR, O. A New Perspective on Computational Thinking. **Communications of the ACM**, v. 61, n. 7, p. 33-39, jul. 2018.

YIN, R. K. **Estudo de caso: planejamento e métodos**. Tradução: Cristhian Matheus Herrera. 5. ed. Porto Alegre: Bookman, 2015

ZIPITRÍA, S. R. 2018. Piaget and Computational Thinking. **In The 7th Computer Science Education Research Conference (CSERC '18)**, October 10-12, 2018, Saint Petersburg, Russian Federation. ACM, New York, NY, USA, p. 44-50, 2018.