

PENSAMENTO COMPUTACIONAL E ATIVIDADE DE PROGRAMAÇÃO: PERSPECTIVAS PARA O ENSINO DA MATEMÁTICA

COMPUTATIONAL THINKING AND PROGRAMMING ACTIVITY: PERSPECTIVES FOR TEACHING MATHEMATICS

Maria Elisabette Brisola Brito Prado
Universidade Anhanguera de São Paulo – UNIAN
bette.prado@gmail.com

Angélica da Fontoura Garcia Silva
Universidade Anhanguera de São Paulo – UNIAN
angelicafontoura@gmail.com

Ruy Cesar Pietropaolo
Universidade Anhanguera de São Paulo – UNIAN
rpietropaolo@gmail.com

Samira Fayes Kfourri da Silva
Universidade Norte do Paraná – UNOPAR
samira.kfourri@unopar.br

Resumo

Por meio de uma pesquisa bibliográfica/documental, realizou-se um estudo teórico para discutir questões relativas ao pensamento computacional desenvolvidas a partir da proposição de atividades de programação nas escolas de educação básica. Para realizar tal discussão, expôs-se, inicialmente, o olhar de alguns pesquisadores sobre o conceito do pensamento computacional e, com um breve resgate histórico, procurou-se estabelecer relações entre os pressupostos presentes nesse tipo de raciocínio e os fundamentos do construcionismo, conforme descrito por Papert. Diante de tal análise, buscou-se identificar nos documentos oficiais da Base Nacional Comum Curricular (BNCC) da área de Matemática evidências de indicações do desenvolvimento de habilidades ligadas ao raciocínio computacional, sendo que este último foi notado em atividades de programação. Foi possível observar que há proposição desse desenvolvimento na preocupação em oferecer a alunos da educação básica vivências que favoreçam a resolução de problemas, envolvendo abstração lógica e representação de dados, além da utilização de conceitos de uma linguagem de programação na implementação de algoritmos escritos em linguagem corrente e/ou matemática.

Palavras-chave: Construcionismo. Resolução de problema. Algoritmo. Raciocínio lógico. Pensamento Computacional. Base Nacional Comum Curricular.

Abstract

Through a bibliographic/documentary research, a theoretical study was carried out to discuss issues related to computational thinking developed from the proposal of programming activities in basic education schools. For this discussion, we brought some researchers' perspectives on the concept of computational thinking, and, with a brief historical review, we sought to establish relationships between the assumptions present in this type of reasoning and the fundamentals of constructionism, as described by Papert. In the face of such an analysis, we aimed to identify in the official documents of the National Common Curricular Base (BNCC) in the Mathematics area evidence of indications of the development of skills related to computational reasoning, the latter being noted in programming activities. The concern with offering basic education students experiences that favour problem-solving involving logical abstraction and data representation, besides concepts of a programming language to implement algorithms written in current and/or mathematics language, shows that there is a focus on the development of such skills.

Keywords: Constructionism. Problem-solving. Algorithm. Logical reasoning. Computational thinking. National Common Curricular Base.

INTRODUÇÃO

Este artigo tem como objetivo apresentar um estudo teórico sobre um tema que vem sendo discutido em vários países: a atividade de programação nas escolas de educação básica visando ao desenvolvimento do pensamento computacional. Essa abordagem torna-se bastante relevante neste momento de implementação da Base Nacional Comum Curricular (BNCC)¹ em nosso país, uma vez que o conceito desse tipo de raciocínio consta como indicativo necessário na área de Matemática.

Assim, organizamos este estudo apresentando a visão de alguns autores sobre o conceito de pensamento computacional e dialogando com os pressupostos teóricos do construcionismo a partir de uma breve revisita ao percurso da tecnologia na Educação. Na sequência, abordamos a tendência que vem sendo enfatizada em vários países e envolve a atividade de programação nas escolas de educação básica. A partir dessa análise, buscamos compreender as possibilidades de promover o desenvolvimento do pensamento computacional por meio da atividade de programação e dos argumentos constantes na BNCC (BRASIL, 2018) da área de Matemática. Ademais, destacamos alguns pontos que necessitam ser analisados e discutidos mais profundamente, especialmente a questão da formação do professor.

PENSAMENTO COMPUTACIONAL: O QUE DIZEM OS AUTORES

O conceito de pensamento computacional vem sendo explicitado e discutido tanto por pesquisadores da área da Ciências da Computação como por educadores, especialmente da área

¹ Brasil (2018).

de Matemática. Destaca-se que esse tipo de pensamento pode favorecer o desenvolvimento de competências cognitivas, tendo como base a resolução de problemas.

Embora ainda não exista, segundo Valente (2016, 2019), uma definição unânime sobre o pensamento computacional acordada entre os pesquisadores nacionais e internacionais interessados nesse tema, alguns autores têm sido amplamente referenciados por elucidar elementos envolvidos nesse tipo de pensamento. Wing (2006, p. 33), por exemplo, frisa que “o pensamento computacional é uma habilidade fundamental para todos, não somente para cientistas da computação”, e acrescenta que, além da “leitura, escrita e aritmética, deveríamos incluir o pensamento computacional na habilidade analítica de todas as crianças.”

Os autores Ramos e Espadeiro (2014, p. 2), apoiados nas ideias de Wing, elencam algumas das capacidades analíticas consideradas fundamentais para o pensamento computacional, tais como: “pensamento recursivo, pensamento sequencial e paralelo, abstração, automação, decomposição, modelação, simulação”, entre outros. Liukas (2015 *apud* BRACKMANN, 2017, p. 28) destaca algumas características do pensamento computacional, as quais envolvem: o “pensamento lógico, a habilidade de reconhecimento de padrões, raciocinar através de algoritmos, decompor e abstrair um problema.”

A Associação de Professores de Ciência da Computação (Computer Science Teachers Association, CSTA), em parceria com a Sociedade Internacional de Tecnologia em Educação (International Society of Technology and Education, ISTE), elaborou uma definição de pensamento computacional baseada em uma pesquisa com mais de 700 professores e pesquisadores de diversas partes do mundo e na resolução de problema. Assim, relacionou tal conceito a estes aspectos:

Formulação de problemas de uma maneira que permita usar um computador e outras ferramentas para ajudar a resolvê-los; organização lógica e análise de dados; representação de dados por meio de abstrações, como modelos e simulações; automação de soluções por meio do pensamento algorítmico (uma série de etapas ordenadas); identificação, análise e implementação de possíveis soluções com o objetivo de alcançar a mais eficiente e efetiva combinação 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).

Já o pesquisador Blikstein (2008) destaca que o pensamento computacional envolve o uso do computador como uma ferramenta que amplia o poder cognitivo e operacional humano de modo a favorecer a produção de conhecimento e a criatividade. Sob esse enfoque, uma possibilidade a ser considerada seria a atividade de programação, na perspectiva desenvolvida na década de 1960 por Seymour Papert (1985), criador da linguagem de programação *Logo*. Para Papert (1985), a atividade de programação poderia viabilizar que as crianças e os jovens

aprendessem interagindo com o computador de forma atrativa e interessante, ensinando a máquina, ou seja, construindo programas e vivenciando um processo criativo e reflexivo de aprender, no sentido do *hands-on and head-in* (mão na massa e mente envolvida). Segundo Papert e Harel (1991), a atividade de programação deve proporcionar às pessoas “pensar com” as máquinas e “pensar sobre” o próprio pensamento.

Podemos perceber que tais ideias de Papert (1985) e de Papert e Harel (1991) vão ao encontro do movimento atual de sublinhar a importância de proporcionar o desenvolvimento do pensamento computacional aos estudantes desde a educação básica. Valente (2019, p. 5) relembra que “o termo pensamento computacional já havido sido mencionado por Papert (1992, p. 184)” ao destacar que, na atividade de programação com a linguagem *Logo*, as crianças poderiam explorar conceitos computacionais por meio da resolução de problemas, o que permitia desenvolver o pensamento lógico e as ideias matemáticas.

Papert, como educador e matemático, estudioso da Inteligência Artificial e humana elaborou um modelo teórico, denominado de construcionismo². Ele foi ampliado e sistematizado por Valente (2002) por meio da criação de um modelo constituído por um ciclo de ações que ocorre durante a atividade de programação: descrição-reflexão-depuração-(nova)descrição, que se configura em uma espiral de aprendizagem. Esse modelo tem orientado, ao longo dos anos, práticas que enfatizam ao aluno a relevância de assumir uma postura ativa e um papel de protagonista no processo de aprender com as tecnologias digitais. Considerando que a atividade de programação no enfoque dado pelas ideias de Papert (1985) no passado expressa elementos que atualmente vem sendo discutidos no âmbito do pensamento computacional, apresentamos a seguir um pouco da história da tecnologia na educação de nosso país.

REVISITANDO A HISTÓRIA PARA ENTENDER O PRESENTE

No Brasil, a atividade de programação da linguagem *Logo*, utilizada na educação básica, teve início no final dos anos 80. Nessa época, dois grandes pesquisadores — Prof.^a Dra. Léa Fagundes, da Universidade Federal do Rio Grande do Sul (UFRGS) e Prof. Dr. José Armando Valente, da Universidade Estadual de Campinas (Unicamp) —, notadamente, foram precursores fundamentais na implementação e no desenvolvimento de pesquisas sobre o uso da linguagem de programação *Logo* no contexto da escola pública. Decorrente dessas

² No construcionismo, o aluno aprende fazendo e refletindo, construindo algo que seja significativo, de modo que possa se envolver afetiva e cognitivamente com aquilo que está sendo produzido (PRADO, 2003).

experiências, várias pesquisas foram realizadas enfocando essa temática, como as de Valente (1996, 1999), a de Fagundes *et al.* (2019), entre outras. Essas investigações destacam possibilidades e dificuldades que ocorreram ao integrar a atividade de programação na prática escolar. As possibilidades, principalmente quando as práticas eram desenvolvidas usando a metodologia de projetos, causaram impactos significativos no engajamento dos alunos, pois estes estavam aprendendo — ensinando a máquina — por meio da criação de programas. Embora essas experiências tenham sido enriquecedoras, foram restritas e pontuais. Muitas delas constam publicadas em periódicos, dissertações, teses e livros, como vemos em Almeida (2000) e Maltempi (2004), por exemplo. Dentre as dificuldades, as pesquisas realçaram dois aspectos cruciais: (1) a infraestrutura das escolas (laboratórios, manutenção dos equipamentos, planejamento, gestão da aula etc.); (2) a formação do professor (técnico e pedagógico) para vincular a atividade de programação aos conteúdos curriculares. São aspectos que, entre outros, naquela época, contribuíram para o enfraquecimento do uso da programação na educação básica, principalmente nas escolas públicas.

Aquele momento evidenciou a existência de um descompasso entre a implementação do uso da atividade de programação da linguagem *Logo* nas escolas e o crescente avanço das tecnologias, tais como os recursos de multimídia, internet, *softwares* educacionais, jogos, objetos de aprendizagem e simuladores. No entanto, a expansão e a diversidade tecnológica produziram novas perspectivas educacionais, pois ampliaram as possibilidades de inserir novos recursos didáticos tecnológicos na prática escolar. Inclusive, foi nesse momento – no final da década de 90 – que essa discussão ganhou respaldo nos documentos oficiais curriculares, os quais destacaram a importância da inserção dos computadores na educação, visando a preparar os estudantes para o desenvolvimento de novas competências e demandas sociais. Nesse momento, orientações curriculares como os Parâmetros Curriculares Nacionais (PCN), por exemplo, já abordavam tais discussões:

É indiscutível a necessidade crescente do uso de computadores pelos alunos como instrumento de aprendizagem escolar, para que possam estar atualizados em relação às novas tecnologias da informação e se instrumentalizarem para as demandas sociais presentes e futuras. (BRASIL, 1997, p. 67)

Justamente nesse período, em 1997, foi criado o Programa Nacional de Informática na Educação (Proinfo), vinculado ao Ministério da Educação (MEC), com o propósito de disseminar nas escolas públicas do país o uso pedagógico dos computadores. Para tanto, o governo desenvolveu, em parceria com os estados e os municípios, várias ações, tanto relacionadas à infraestrutura envolvendo a implantação de laboratórios de informática nas escolas públicas quanto relativas à criação de Núcleos e Centros de Informática na Educação,

compostos por equipes de profissionais preparados para dar suporte e acompanhar a inserção do computador nas práticas escolares. Para preparar essas equipes de profissionais, houve também um grande incentivo do governo, que, juntamente com alguns profissionais das universidades, realizaram diversos cursos de formação continuada voltados para o uso das tecnologias na educação (VALENTE, 1999).

O resultado desse investimento propiciou a elaboração de práticas inovadoras com a utilização das tecnologias em várias escolas públicas espalhadas em diversas regiões do país. Tais práticas se deram de várias maneiras, demonstrando diferentes níveis de apropriação de tecnologia e de metodologias de implementação, de acordo com cada realidade. O efeito dessas ações, enquanto uma política pública abrangência nacional, provocou algumas mudanças no contexto das escolas, oportunizando aos estudantes explorar o uso pedagógico dos recursos tecnológicos. Porém, também houve a descontinuidade desse processo em escolas de diferentes regiões do país. As razões dessa interrupção foram diversas e retratam a complexidade de fatores relacionados às distintas instâncias da sociedade (política, econômica, educacional).

Os estudos de pesquisadores tanto nacionais como internacionais, como Sandhotz, Ringstaff e Dwyer (1997), Prado e Valente (2003) e Lobo da Costa e Prado (2015), que abordam temas relacionados à formação de educadores para o uso pedagógico das tecnologias na prática escolar, constatam que esse processo não é simples nem imediato, pois requer a reconstrução de conhecimentos.

[...] a formação do profissional prático não pode apenas enfatizar o aprendizado operacional das ferramentas tecnológicas, tampouco o aprendizado sobre o que postula uma determinada teoria educacional. Este profissional precisa construir novos conhecimentos; relacionar, relativizar e integrar diferentes conteúdos; ressignificar aquilo que sabe fazer com vistas a reconstruir um referencial pedagógico *na e para* uma nova prática. (PRADO; VALENTE, 2003, p. 22)

De fato, a formação do professor tem sido um grande desafio ao longo do tempo, e esse processo necessita ser repensado considerando os impactos das tecnologias digitais na sociedade atual. Estamos vivenciando o desenvolvimento de uma nova cultura, que se caracteriza por uma maneira muito diferente de as pessoas desenvolverem suas atividades em praticamente todos os segmentos da sociedade. Isso, sem dúvidas, acaba influenciando a forma de se comunicar, de obter informações, de resolver problemas, de se relacionar e de pensar. Portanto, implica em novas formas de aprender e, conseqüentemente, requer novas formas de ensinar.

Diante desse cenário, observamos um movimento que busca ressignificar as metodologias de natureza ativa, resgatando e recriando estratégias inovadoras. Citamos como

exemplo: o ensino *maker*, a aprendizagem criativa, a robótica, o ensino híbrido e a programação. O foco dessas práticas deve ser propiciar ao estudante o aprender-fazendo e, principalmente, desenvolver o pensamento computacional, assim como os multiletramentos (o matemático, o estatístico, o de leitura e escrita, o digital, o científico), que se referem às competências necessárias para o aprender a aprender na sociedade tecnológica.

TENDÊNCIAS PARA A EDUCAÇÃO DO SÉCULO XXI

Há um movimento crescente de pesquisadores e profissionais das áreas da Educação e da Computação dos Estados Unidos, do Brasil e de vários países da Europa que discutem e desenham estratégias de como inserir ações que viabilizem ao estudante da educação básica desenvolver o pensamento computacional. Na realidade brasileira, existem iniciativas inovadoras que vêm sendo desenvolvidas em algumas escolas privadas e, mais timidamente, em escolas públicas e envolvem, por exemplo, as linguagens de programação *Scratch*, *Arduíno* e *Robótica*, que, muitas vezes, são colocadas em prática nos espaços *maker*³.

É interessante observar que, neste momento histórico, o documento oficial curricular contempla orientações que marcam a importância da utilização das Tecnologias de Informação e Comunicação e da Cultura Digital.

[...] a cultura digital tem promovido mudanças sociais significativas nas sociedades contemporâneas. Em decorrência do avanço e da multiplicação das tecnologias de informação e comunicação e do crescente acesso a elas pela maior disponibilidade de computadores, telefones celulares, tablets e afins, *os estudantes estão dinamicamente inseridos nessa cultura, não somente como consumidores*. (BRASIL, 2018, p. 61, grifos nossos)

Essa citação apresenta argumentos que podem impulsionar a concretização de novas práticas que integrem o uso das tecnologias na educação básica. Além disso, sinaliza um aspecto importante: cabe à escola cumprir seu papel de formação de cidadãos preparados não apenas para serem consumidores das tecnologias digitais, mas também para serem produtores de novas estratégias para a solução de problemas e para a criação de artefatos tecnológicos virtuais digitais.

Notadamente, a Base Nacional Curricular Comum (BRASIL, 2018), especificamente da área de Matemática, enfatiza a utilização das tecnologias na perspectiva da resolução de problemas. Com isso, visa a contribuir para o desenvolvimento de competências relativas ao letramento matemático (raciocínio, representação, comunicação e argumentação) e ao

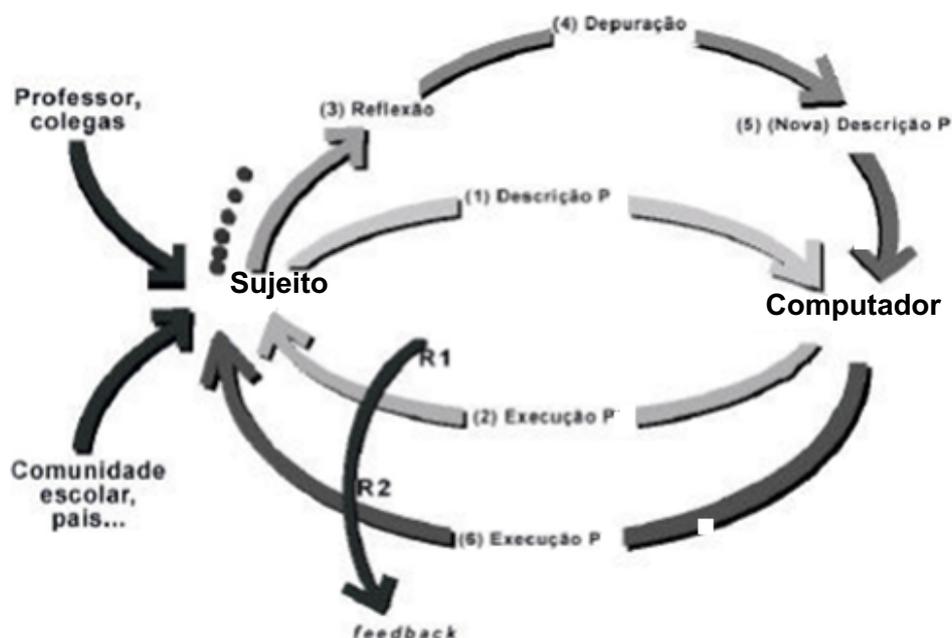
³ Segundo Dougherty (2013), para o desenvolvimento das práticas *maker*, podem ser utilizados os princípios de programação e robótica, plataformas como o Arduíno e fabricação digital, como as impressoras 3D, por exemplo.

pensamento computacional. Sob esse enfoque, entre as dez competências a serem desenvolvidas nas três etapas da Educação Básica (Educação Infantil, Ensino Fundamental e Médio), a quinta ressalta aspectos que devem estar inseridos na prática do professor na utilização das tecnologias digitais, como mostra o texto a seguir:

Competência 5 – [...] compreender, utilizar e criar tecnologias digitais de informação e comunicação de forma *crítica, significativa, reflexiva e ética* nas diversas práticas sociais (incluindo as escolares) para se comunicar, acessar e disseminar informações, *produzir conhecimentos, resolver problemas e exercer protagonismo e autoria na vida pessoal e coletiva.* (BRASIL, 2018, p. 9, grifos nossos)

A compreensão dos aspectos destacados nessa competência é de extrema importância para que seja evitado um uso das tecnologias apenas aparente, com a intenção de modernizar práticas cristalizadas que não favorecem o protagonismo e a autoria do estudante na aprendizagem. Valente (2002) alerta que a ênfase no uso pedagógico das tecnologias não pode ser apenas centrada na técnica, mas também deve englobar o processo de pensamento envolvido na atividade de programar o computador. O autor explica isso ao especificar os elementos constitutivos do ciclo de ações que se estabelece na interação do sujeito com o computador durante a atividade de programação, o qual vai se configurando na espiral de aprendizagem, como ilustra a Figura 1.

Figura 1 – Espiral de aprendizagem na atividade de programação



Fonte: Prado (2003, p. 24)

A Figura 1 mostra a interação do sujeito com o computador, que se inicia quando ele elabora a *Descrição P* (etapa 1) para o computador, que se refere ao algoritmo da solução do problema. Para isso, o sujeito precisa interpretar o problema, escrever o algoritmo com base em conceitos formais ou intuitivos de Matemática, os quais são traduzidos para a linguagem computacional. Portanto, na descrição, são utilizados conceitos e estratégias relacionados ao problema e ao ambiente computacional.

A partir dessa descrição, o computador faz a *execução* (etapa 2) imediata e mostra na tela do computador o *resultado (RI)*, que corresponde ao *feedback* da máquina e expressa aquilo que o sujeito pensou. Em termos de aprendizagem, é interessante quando surge um confronto entre o resultado obtido e o esperado, porque essa situação pode favorecer a ocorrência da *reflexão* (etapa 3) do sujeito sobre o próprio pensamento em relação aos conceitos e às estratégias utilizadas na resolução do problema. É, portanto, vivenciando esse processo reflexivo que o sujeito se sente instigado a buscar novas compreensões para fazer a *depuração* (etapa 4) de suas ideias, revendo sua interpretação do problema, bem como os conceitos envolvidos; a partir daí, o ciclo de ações recomeça num outro patamar de compreensão, revelado por uma *nova-descrição* (etapa 5).

Embora a reflexão possa perpassar todo o ciclo da espiral de aprendizagem, consideramos que é no instante em que o sujeito se depara com o *feedback* do computador que a reflexão ganha uma nova dimensão. Isso acontece porque nessa interação dinâmica do sujeito com o computador, em que é vivenciada a atividade de programação, o *feedback* pode causar o conflito cognitivo, na concepção de Piaget (1976). Tal conflito favorece que o sujeito estabeleça novas relações, as quais o levam a rever suas próprias compreensões sobre os conceitos, que vão sendo explicitados por novas descrições.

No entanto, nem sempre esse processo de reflexão e depuração ocorre espontaneamente, apenas na interação do sujeito que está programando. Segundo Valente (2002) e Prado (2003), o *feedback* do computador tem um caráter lógico, próprio da linguagem computacional, e pode não ser suficiente para desencadear a reflexão. Muitas vezes, torna-se necessário haver interação humana, tanto do professor que faz a mediação pedagógica como dos demais atores envolvidos no contexto, por exemplo, os outros alunos.

Em relação ao processo mental que ocorre durante a atividade de programação desenvolvida na perspectiva do construcionismo, Duda, Pinheiro e Silva (2019, p. 51) destacam que

[...] é importante contemplar o caráter exploratório, de forma que se dê liberdade ao aprendiz para criar soluções e, no que diz respeito à abordagem construcionista com o intuito de estimular o pensamento abstrato, criação de cenários de exploração conceitual de estruturas algébricas que regem determinados processos.

Concordando com esse entendimento sobre a atividade de programação, a BNCC explicita que esse tipo de pensamento “envolve as capacidades de compreender, analisar, definir, modelar, resolver, comparar e automatizar problemas e suas soluções, de forma metódica e sistemática, por meio do desenvolvimento de algoritmos.” (BRASIL, 2018, p. 474).

E ainda destaca que

a linguagem algorítmica tem pontos em comum com a linguagem algébrica, sobretudo em relação ao conceito de variável. Outra habilidade relativa à álgebra que mantém estreita relação com o pensamento computacional é a identificação de padrões para se estabelecer generalizações, propriedades e algoritmos. (BRASIL, 2018, p. 269, grifos nossos).

Tais ideias – conceito de variável e generalização – estão presentes nesses dois pensamentos – o algébrico e o computacional – e parecem os aproximar. No âmbito da Educação Matemática, os autores Blanton e Kaput (2005) chamam a atenção para a centralidade da generalização no pensamento algébrico; da mesma forma, consideramos que a generalização ocupa um lugar de destaque no pensamento computacional. Todavia, é preciso salientar outras especificidades envolvidas nesses dois tipos de pensamento, como as mudanças de registros e representações. Liukas (2015 *apud* BRACKMANN, 2017, p. 28) afirma que “o pensamento computacional é executado por pessoas e não por computadores” e acrescenta que é preciso “pensar nos problemas de forma que um computador [usando uma linguagem própria de programação] consiga solucioná-los.”

Vale ressaltar que, no contexto da educação básica, são utilizadas as linguagens de programação de alto nível, ou seja, com maior poder de abstração, e uma sintaxe mais próxima das linguagens naturais. Entre essas linguagens, atualmente vem sendo utilizadas no contexto escolar aquelas que se enquadram, segundo Baranauskas (1998), no paradigma computacional orientado a objetos, por exemplo, a linguagem de programação *Scratch*. Ela foi desenvolvida pelo grupo de pesquisa *Lifelong Kindergarten* do Massachusetts Institute of Technology (MIT) Media Lab, com o objetivo de propiciar a participação de pessoas das mais variadas idades, formações e nacionalidades na criação de *softwares* como jogos, animações e simulações, conforme indicado por Resnick (2009). Segundo este autor, o *Scratch* foi criado não com o intuito de formar programadores profissionais, mas para que as pessoas pudessem se expressar de forma criativa por meio da programação computacional.

Considerando tais argumentos sobre o pensamento computacional apresentados tanto em pesquisas como nas orientações curriculares, ficam claras sua importância e a possibilidade da retomada da atividade de programação na educação básica. Isso deve ser levado em conta principalmente para esta geração, que, num futuro próximo, deverá estar preparada para atuar numa sociedade cada vez mais tecnológica.

ALGUMAS CONSIDERAÇÕES

Neste estudo teórico, buscamos compreender as características do pensamento computacional e as possibilidades de promover seu desenvolvimento mediante a atividade de programação na educação básica, com base nos argumentos constantes na BNCC da área de Matemática (BRASIL, 2018). A partir disso, destacamos alguns pontos que necessitam ser analisados e discutidos mais profundamente.

Consideramos que a atividade de programação propicia ao aprendiz, na vivência com o ciclo da espiral de aprendizagem, explorar e compreender diversas ideias matemáticas, além daquelas explicitadas na BNCC (BRASIL, 2018). Cabe, portanto, ampliar e aprofundar os estudos sobre o pensamento computacional, pois nossa análise destaca a atividade de programação como uma das possibilidades de construí-lo.

É importante ressaltar que atividade de programação desenvolvida na educação básica precisa estar pautada nos princípios construcionista, os quais concebem o aluno como participante ativo e o professor como mediador nos processos de ensino e aprendizagem. Sob esse enfoque, a atividade de programação pode ter um papel fundamental, permitindo ao estudante, enquanto ensina o computador a resolver o problema, colocar em ação, compreender na ação e refletir sobre as propriedades e a construção de fórmulas e regras matemática de forma ativa e criativa. Esse processo do aluno de identificar padrões, estabelecer relações entre propriedades e generalizar conceitos propicia-lhe ter novos patamares de compreensão e de elaboração de conhecimentos. Tais habilidades, já mencionadas por autores citados neste artigo (WING, 2006; RAMOS; ESPADEIRO, 2014; VALENTE, 2016), por exemplo, estão relacionadas ao pensamento computacional, que se centra na resolução de problemas, envolvendo abstração, lógica e representação de dados, assim como a construção da linguagem algorítmica, a qual tem pontos em comum com a linguagem algébrica para a criação de novos artefatos e conhecimentos.

Finalizando e, ao mesmo tempo, iniciando nossa reflexão sobre o pensamento computacional, constatamos que essa temática apresenta grandes desafios, principalmente para

educadores. Isso porque ela envolve desde a compreensão epistemológica e prática desse conceito até o ato de repensar a escola, o currículo, a prática do professor e a expectativa dos alunos.

Um segundo ponto que destacamos é a formação do professor, pois, como vimos na retomada histórica, o ingresso dos computadores nas escolas e o uso da linguagem de programação *Logo* aconteceram de maneira pontual e restrita; ou seja, a experiência em situações mesmo exitosa não chegou a ser consolidada. Daí surge nossa preocupação: o professor está preparado para integrar em suas práticas atividades que podem propiciar o desenvolvimento computacional? É claro que, neste momento, o cenário social demanda, com urgência, que a formação do aluno esteja voltada para o desenvolvimento de competências para um novo perfil profissional, mas os desafios continuam quando consideramos a realidade da escola e a formação tanto continuada como inicial de docentes.

Portanto, caberá aos gestores do sistema educacional repensar estratégias para a implementação da BNCC, contemplando o enfoque do pensamento computacional. As questões e as decisões são complexas e requerem compromisso, pesquisa, articulação de diferentes saberes para reconstruir os espaços de ensino e aprendizagem para as crianças e os jovens da sociedade digital.

REFERÊNCIAS

- ALMEIDA, M. E. B. *O computador na escola: contextualização e formação de professores*. 2000. 281 f. Tese (Doutorado em Educação: Currículo) – Pontifícia Universidade Católica, São Paulo, 2000.
- BARANAUSKAS, M. C. Procedimento, função, objeto ou lógica? Linguagens de Programação vistas pelos seus paradigmas. In: VALENTE, J. A. (org.). *Computadores e conhecimento: repensando a educação*. 2. ed. Campinas: UNICAMP/NIED, 1998. p. 56-78.
- BLANTON, M.; KAPUT, J. Characterizing a classroom practice that promotes algebraic reasoning. *Journal for Research in Mathematics Education*, Boston, v. 36, n. 5, p. 412-446, 2005.
- BLIKSTEIN, P. O pensamento computacional e a reinvenção do computador na educação. In: _____. *Paulo Blikstein*. Stanford, 22 dez. 2008. Disponível em: http://www.blikstein.com/paulo/documents/online/ol_pensamento_computacional.html. Acesso em: 25 jun. 2018.
- BRACKMANN, C. P. *Desenvolvimento do pensamento computacional através de atividades desplugadas na educação básica*. 2017. 226 f. Tese (Doutorado em Informática na Educação) – Universidade Federal do Rio Grande do Sul, Porto Alegre, 2017.
- BRASIL. Ministério da Educação. *Base Nacional Comum Curricular: Educação é a Base: Ensino Fundamental*. Brasília: MEC, 2018.

_____. Ministério da Educação. Secretaria de Educação Fundamental. *Parâmetros curriculares nacionais: matemática* (1.º e 2.º ciclos do ensino fundamental). Brasília: SEF/MEC, 1997.

DOUGHERTY, D. The maker mindset. In: HONEY, M.; KANTER, D. E. (org.). *Design, make, play: Growing the next generation of STEM innovators*. New York: Routledge, 2013. p. 7-16.

DUDA, R.; PINHEIRO, N. A. M.; SILVIA, S. de C. R. A prática construcionista e o pensamento computacional como estratégias para manifestações do pensamento algébrico. *REnCiMa*, São Paulo, v. 10, n. 4, p. 39-55, 2019.

FAGUNDES, L. da C.; ARAGÓN, R.; BASSO, M. V. de A.; MRASCHIN, C. Laboratório de Estudos Cognitivos: percursos de pesquisa, formação e criação. *Informática na Educação: teoria & prática*, Porto Alegre, v. 22, n. 2, p. 242-257, 2019.

INTERNATIONAL SOCIETY FOR TECHNOLOGY IN EDUCATION; COMPUTER SCIENCE TEACHERS ASSOCIATION. *Computational Thinking Teacher Resource*. 2. ed. [S. l.]: ISTE: CSTA, 2011.

LOBO DA COSTA, N. M.; PRADO, M. E. B. B. A integração das tecnologias digitais ao ensino de Matemática: desafio constante no cotidiano escolar do professor. *Perspectivas da Educação Matemática*, Campo Grande, v. 8, n. 16, p. 99-119, 2015.

MALTEMPI, M. V. Construcionismo: pano de fundo para pesquisas em informática aplicada à educação matemática. In: BICUDO, M. A. V.; BORBA, M. C. (org.) *Educação Matemática: pesquisa em movimento*. São Paulo: Cortez, 2004. p. 264-282.

PAPERT, S. *Logo: computadores e educação*. São Paulo: Brasiliense, 1985.

PAPERT, S.; HAREL, I. *Constructionism*. Norwood: Ablex Publishing, 1991.

PIAGET, J. *A equilibração das estruturas cognitivas: problema central do desenvolvimento*. Rio de Janeiro: Zahar, 1976.

PRADO, M. E. B. B. *Educação a Distância e formação do professor: redimensionando concepções de aprendizagem*. 2003. 291 f. Tese (Doutorado em Educação: Currículo) – Pontifícia Universidade Católica, São Paulo, 2003.

PRADO, M. E. B. B.; VALENTE, J. A. A formação na ação do professor: uma abordagem na e para uma nova prática pedagógica. In: VALENTE, J. A. (org.). *Formação de Educadores para o uso da Informática na Educação*. Campinas: UNICAMP/NIED, 2003. p. 21-38.

RAMOS, J. L.; ESPADEIRO, R.G. Os futuros professores e os professores do futuro. Os desafios da introdução ao pensamento computacional na escola, no currículo e na aprendizagem. *Educação, Formação & Tecnologias*, Lisboa, v. 7, n. 2, p. 4-25, 2014.

RESNICK, M. *et al.* Scratch: programming for all. *Communications of the ACM*, New York, v. 52, n. 11, p. 60-67, 2009.

SANDHOTZ, J.H.; RINGSTAFF, C.; DWYER, D.C. *Ensinando com Tecnologia – Criando salas de aulas centradas nos alunos*. Porto Alegre: Artes Médicas, 1997.

VALENTE, J. A. A Espiral da Aprendizagem e as Tecnologias da Informação e Comunicação: repensando conceitos. In: JOLY, M. C. (org.). *Tecnologia no ensino: implicações para a aprendizagem*. São Paulo: Casa do Psicólogo, 2002. p. 15-37.

_____. Integração do pensamento computacional no currículo da educação básica: diferentes estratégias usadas e questões de formação de professores e avaliação do aluno. *Revista e-Curriculum*, São Paulo, v. 14, n. 3, p. 864-897, 2016.

_____. *O computador na sociedade do conhecimento*. Campinas: Gráfica da Unicamp, 1999.

_____. (org.). *O professor no ambiente Logo: formação e atuação*. Campinas: Nield/Unicamp, 1996.

_____. Pensamento computacional, letramento computacional ou competência digital? Novos desafios da educação. *Revista Educação e Cultura Contemporânea*, Rio de Janeiro, v. 16, n. 43, p. 147-168, 2019.

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

Submetido em 19 de outubro de 2019.
Aprovado em 26 de janeiro de 2020.