

PENSAMENTO COMPUTACIONAL NA FORMAÇÃO INICIAL DE PROFESSORES DE MATEMÁTICA: Um Estudo de Caso sob a Perspectiva da Teoria de Robbie Case

Ana Paula Canal¹
Vanilde Bisognin²
Sílvia Maria de Aguiar Isaia³

RESUMO

O Pensamento Computacional tem sido inserido no ensino como um componente curricular específico da Computação ou em conjunto a outros componentes curriculares. Há necessidade de desenvolver as habilidades do Pensamento Computacional nos estudantes e, conseqüentemente, nos professores. Como forma de inserir o Pensamento Computacional com o componente curricular da Matemática, este artigo traz um estudo de caso realizado com os licenciandos em Matemática em uma universidade privada. Para tal, foi ofertada uma disciplina extracurricular, trabalhando o conteúdo de padrões e regularidades, com o Pensamento Computacional, por meio da linguagem de programação Python. A pesquisa é qualitativa, do tipo estudo de caso, em que foram analisadas duas atividades desenvolvidas pelos licenciandos sob a perspectiva da teoria neopiagetiana de Robbie Case. Com a investigação, identificaram-se as relações entre o Pensamento Computacional e o Pensamento Algébrico e foi proposto um caminho para inclusão do Pensamento Computacional no ensino para formação inicial de professores de Matemática.

Palavras-chave: Computação. Ensino. Padrões e regularidades. Pensamento algébrico.

COMPUTATIONAL THINKING IN THE INITIAL TRAINING OF MATHEMATICS TEACHERS: A CASE STUDY FROM THE PERSPECTIVE OF ROBBIE CASE'S THEORY

ABSTRACT

Computational Thinking teaching has been developed as a specific subject or together with other curricular subjects. There is a need to develop Computational Thinking skills in students and, especially, in teachers. As a way to teach computational thinking with Mathematical curricular subjects, this paper brings a case study with college math students, at a private university. An extracurricular discipline was offered, about patterns and regularities, with Computational Thinking, using Python programming language. This research is qualitative, is a case study and two activities developed by the college math students are analyzed. The Robbie Case's neo piagetian theory was used to analyze the data. The results bring the relationships between Computational Thinking and Algebraic Thinking and a proposal to include Computational Thinking in teaching, for initial teacher training.

Keywords: Computation. Teaching. Patterns and regularities. Algebraic thinking.

Recebido em: 17/11/2020

Aceito em: 4/3/2021

¹ Autora correspondente. Universidade Franciscana – UFN. R. dos Andradas, 1614 – Centro. CEP 97010-030. Santa Maria/RS, Brasil. <http://lattes.cnpq.br/9715593348031501>. <https://orcid.org/0000-0001-6360-1688>. apc@ufn.edu.br

² Universidade Franciscana – UFN. Santa Maria/RS, Brasil. <http://lattes.cnpq.br/1245060231128858>. <https://orcid.org/0000-0001-5718-4777>.

³ Universidade Franciscana – UFN. Santa Maria/RS, Brasil. <http://lattes.cnpq.br/0954246692473735>. <https://orcid.org/0000-0002-9987-7931>.

Muitas pesquisas para inclusão do Pensamento Computacional no Ensino Fundamental e Médio, no Brasil, têm sido realizadas, como apontam os trabalhos de revisão sistemática da literatura (AVILA *et al.*, 2016, 2017; SANTOS; ARAUJO; BITTENCOURT, 2018). Internacionalmente, há também pesquisas sobre essa temática, como mostram as revisões de literatura (BARCELOS *et al.*, 2015; WEINTROP *et al.*, 2016; SHUTE; SUN; ASBELL-CLARKE, 2017; CANSU; CANSU, 2019).

No Brasil, até há pouco tempo, não havia definição legal quanto à incorporação do Pensamento Computacional no ensino. Existiam esforços de grupos de pesquisa, da iniciativa privada e de sociedades, como a Sociedade Brasileira de Computação, para tal. Com a Base Nacional Comum Curricular (BNCC), de 2018 (BRASIL, 2018), o ensino da computação passa a ser contemplado, inclusive, na área da Matemática.

Para a Sociedade Brasileira de Computação (SBC), o Brasil deve acompanhar as constantes mudanças da Sociedade Mundial (RAABE *et al.*, 2017). Reconhecer e compreender aspectos da computação no cotidiano é necessidade crescente neste cenário, em que produtos, processos e serviços artificiais estão presentes e orientam as formas de trabalho e relações sociais. Somado ao uso das tecnologias, é preciso compreender seu funcionamento e conhecer o que está envolvido para empregá-las de forma adequada.

O Pensamento Computacional é um campo emergente (VALENTE *et al.*, 2017) e influencia outras áreas do conhecimento. Na Educação Básica, têm-se duas maneiras de trabalhá-lo: componente específico ou integrado a outros componentes curriculares, esta última em menor escala. A temática deste artigo envolve o Pensamento Computacional e o ensino de Matemática, abordando, especificamente, a formação inicial de professores de Matemática a partir da teoria de aprendizagem de Robbie Case, teórico neopiagetiano.

O ensino de Matemática necessita adequar o trabalho escolar à realidade dos alunos, na qual a Matemática é presente em diversos campos (BRASIL, 2018) e, nos dias de hoje, deve-se refletir sobre temas relacionados à Computação. Temas que oportunizam uma maneira diferente de pensar e resolver problemas e são distintos do uso das Tecnologias de Informação e Comunicação. Diante disto, é importante formar o professor para que consiga incluir conceitos da computação em suas práticas, trabalhando o Pensamento Computacional com seus alunos, em conjunto aos conteúdos de sua área fim.

Pela incipiência do Pensamento Computacional, os professores não têm familiaridade com a área e apresentam dificuldades de estabelecer conexões entre estas habilidades e seu componente curricular (SHUTE; SUN; ASBELL-CLARKE, 2017). No Brasil, a área também é emergente e, em razão disto, há poucas iniciativas na formação docente (BARCELOS; BORTOLETTO; ANDRIOLI, 2016; TAVARES; SALVADOR; VIOLA, 2017; AVILA *et al.*, 2016; PINHO *et al.*, 2016).

Robbie Case, em sua teoria, preservou aspectos centrais da epistemologia de Piaget e acrescentou outros: desenvolvimento social e emocional do indivíduo e questões educacionais. Para ele, os indivíduos são solucionadores de problemas, e, conforme crescem, vão se tornando mais competentes em resolver problemas, pois sua coleção de estratégias de resolução aumenta, ampliando sua estrutura cognitiva (CASE, 1989).

A resolução de problemas, para Case, é amparada por estruturas conceituais centrais, gradativamente construídas pelo indivíduo em diferentes domínios, e influenciada por sua cultura e suas interações sociais, trazendo a influência de Vigotsky (CASE, 1992).

Neste artigo são discutidas as relações entre o Pensamento Computacional e a Matemática no estudo de caso desenvolvido em uma turma de licenciandos em Matemática de uma Instituição privada de Ensino Superior, a partir da análise de duas atividades propostas sobre Padrões e Regularidades em conjunto com o Pensamento Computacional. Para tal, aspectos do aporte teórico da pesquisa e o caminho metodológico percorrido são apresentados. As atividades com o Pensamento Computacional e o conteúdo Matemático desenvolvido, no estudo de caso, sob a perspectiva da Teoria de Case, são analisadas e discutidas, bem como as perspectivas futuras da investigação.

O APORTE TEÓRICO: Teoria de Robbie Case e o Pensamento Computacional na Licenciatura em Matemática

Há diferentes definições ao termo Pensamento Computacional. A maioria delas faz referência à Resolução de Problemas. Para a Sociedade Brasileira de Computação (SBC):

O Pensamento Computacional se refere à capacidade de sistematizar, representar, analisar e resolver problemas. Apesar de ser um termo recente, vem sendo considerado como um dos pilares fundamentais do intelecto humano, junto com leitura, escrita e aritmética, pois como estes, serve para descrever, explicar e modelar o universo e seus processos complexos (RAABE *et al.*, 2017, p. 3).

Neste sentido, traz-se a teoria neopiagetiana de Robbie Case: na resolução de problemas é que se encontra a essência do comportamento inteligente (CASE, 1989). Ao resolver problemas, o indivíduo modifica e amplia suas estratégias de solução, as quais promovem o seu desenvolvimento. A resolução dos problemas é influenciada pela cultura em que ele está inserido. Para problemas particulares, a cultura pode prover maneiras comuns de solucioná-los ou limitar o sucesso de sua solução (CASE, 1998).

A Estrutura de Controle Executivo (ECE), identificada por Case, configura a situação-problema concreto ante a qual o sujeito está, os objetivos (aquilo que ele deseja alcançar) e as estratégias aplicadas, para encontrar a solução e atingir o objetivo. A ECE representa o modo como o sujeito constrói a solução do problema, e permite descrever processos estratégicos como sequências de passos mentais (CASE, 1989).

À medida que acontece a transição entre os estágios de desenvolvimento, a mudança mais importante é nas estruturas de controle executivo que se integram. Os indivíduos são dotados de processos reguladores desde cedo, e esses processos promovem a integração das estruturas. Os processos reguladores são Resolução de Problemas, Exploração, Imitação e Regulação Mútua (CASE, 1989).

O processo regulador chamado Resolução de Problemas é aquele em que o sujeito, diante de um objetivo que não pode ser alcançado imediatamente por uma sequência operacional existente, procura uma nova sequência operacional a fim de alcançá-lo (CASE, 1989). O processo regulador Exploração é aquele em que o indivíduo, ante a uma

situação em que se pode aplicar uma estratégia (já conhecida), não sabe prever os resultados. Ao ser aplicada a estratégia, uma nova situação é alcançada, sem mesmo ser esperada. O processo regulador Imitação consiste na imitação da ação ou conduta, por parte do indivíduo, com estrutura de controle executivo de ordem inferior, ao observar a ação ou conduta de outro indivíduo mais experiente (com estrutura de controle executivo de ordem superior) (CASE, 1989).

O processo regulador Regulação Mútua também acontece por meio da interação social, como no processo de Imitação. A Regulação Mútua é a adaptação ativa do indivíduo e de outro ser humano, aos sentimentos, cognições ou condutas de cada um. Cada membro do grupo exerce uma influência sobre o outro e também é influenciado pelo outro (CASE, 1989). O processo de ensino é um exemplo do processo regulador Regulação Mútua.

A teoria explica que o desempenho dos indivíduos está fortemente relacionado à complexidade representacional das tarefas. Conforme a etapa de desenvolvimento, há uma representação conceitual geral mais complexa do mundo (a estrutura conceitual central) e que habilita o indivíduo a elaborar uma variedade de estruturas de controle executivo para encontrar os requisitos de uma tarefa particular.

A Estrutura Conceitual Central (ECC) é definida como “[...] uma rede de nodos e relações semânticas que desempenham um papel central na mediação do desempenho da criança, através de uma ampla gama de tarefas (embora não todas), e que também ocupa um papel central em seu desenvolvimento” (CASE, 1992, p. 273, tradução nossa).

As ECCs aparecem para constituir um ponto de interconexão que favorece o encontro da biologia e da cultura e circundar a compreensão do mundo pelo indivíduo. O conteúdo semântico de tais estruturas é dependente da cultura, de seu sistema simbólico e as instituições pelas quais estes sistemas são adquiridos e utilizados. Por outro lado, os limites gerais aos quais as estruturas estão sujeitas são fatores biológicos e neurológicos. A teoria de Robbie Case embasou o desenvolvimento do estudo de caso realizado nesta investigação, na Licenciatura em Matemática, ao alicerçar o planejamento das aulas e a análise dos dados, como será abordado nas próximas seções.

A formação de professores é um processo contínuo de desenvolvimento de competências para o exercício da docência. Ao futuro professor é necessário ter conhecimentos sobre seu campo e conseguir mobilizá-los, transformando-os em ação a fim de atender às demandas de sua profissão. “[...] a construção da docência envolve simultaneamente os conhecimentos pedagógicos e os conhecimentos da área específica” (BOLZAN; ISAIA; MACIEL, 2013, p. 55). A capacidade de articular esses conhecimentos adequadamente, relacionando-os à realidade e às diferentes situações vivenciadas na sala de aula, se constrói no decorrer da trajetória do próprio professor.

As diretrizes para formação inicial têm como referência a Base Nacional Comum Curricular (BNCC) da Educação Básica. Conforme o artigo 2º, o licenciando deve desenvolver as competências previstas na Base (BRASIL, 2019). Há competências gerais da formação docente e competências específicas. Assim, dos fundamentos pedagógicos à formação inicial docente, em nível superior, salienta-se os itens do artigo 8º:

II – o compromisso com as metodologias inovadoras e com outras dinâmicas formativas que propiciem ao futuro professor aprendizagens significativas e contextualizadas em uma abordagem didático metodológica alinhada com a BNCC, visando ao desenvolvimento da autonomia, da capacidade de resolução de problemas, dos processos investigativos e criativos, do exercício do trabalho coletivo e interdisciplinar, da análise dos desafios da vida cotidiana e em sociedade e das possibilidades de suas soluções práticas;

[...]

IV – emprego pedagógico das inovações e linguagens digitais como recurso para o desenvolvimento, pelos professores em formação, de competências sintonizadas com as previstas na BNCC e com o mundo contemporâneo (BRASIL, 2019, p. 5).

No item II consta o desenvolvimento da capacidade de resolução de problemas e de processos investigativos e criativos e o trabalho interdisciplinar, aos quais entende-se que o Pensamento Computacional vai ao encontro. Além disso, o Pensamento Computacional proporciona uma outra linguagem no mundo contemporâneo, como previsto no item IV e que serão elucidados na descrição das atividades desenvolvidas no estudo de caso.

Os cursos de Licenciatura devem estar organizados, segundo as Diretrizes Curriculares Nacionais. Nessas, um dos grupos de conhecimentos, o Grupo I, abrange conhecimentos científicos, educacionais, pedagógicos e fundamenta a educação e suas articulações com os sistemas, as escolas e as práticas educacionais (BRASIL, 2019). No artigo 12, Parágrafo único, no item II, Didática e seus fundamentos, há referência ao Pensamento Computacional, a saber: “f) compreensão básica dos fenômenos digitais e do pensamento computacional, bem como de suas implicações nos processos de ensino-aprendizagem na contemporaneidade” (BRASIL, 2019, p. 6). As diretrizes para formação de professores determinam, portanto, a compreensão básica do Pensamento Computacional, atrelado aos processos de ensino e de aprendizagem.

Os cursos de formação de professores em áreas específicas, como a Matemática, devem proporcionar uma formação também por meio de vivências interdisciplinares, trabalhando a ideia de que as diferentes áreas do conhecimento se interconectam. Devem ser organizados a fim de desenvolver Competências e Habilidades, como as que estão relacionadas com a investigação: “c) capacidade de compreender, criticar e utilizar novas idéias e tecnologias para a resolução de problemas. [...] f) estabelecer relações entre a Matemática e outras áreas do conhecimento” (BRASIL, 2002, p. 4).

Entende-se que a inserção do Pensamento Computacional no ensino para a formação de professores de Matemática pode promover o desenvolvimento das competências elencadas. As diretrizes curriculares são genéricas e deixam liberdade para que os sistemas de ensino definam os conteúdos e disciplinas específicas (MELLO, 2000). O trabalho na formação inicial, por meio da Simetria Invertida, proporciona preparar o licenciando para atuar em uma realidade em constante mudança, em um ambiente similar ao que se construiu como professor, dadas as devidas diferenças de conjuntura.

A Simetria Invertida considera que a experiência desenvolvida, enquanto aluno em formação, constitui o futuro professor. Ele experimenta, no processo de sua formação, aquilo que pode vir a incorporar a suas práticas pedagógicas futuras. “Ninguém

facilita o desenvolvimento daquilo que não teve oportunidade de aprimorar em si mesmo. Ninguém promove a aprendizagem de conteúdos que não domina, a constituição de significados que não compreende nem a autonomia que não pode construir” (MELLO, 2000, p. 102).

Há, portanto, necessidade de incentivar ações sobre o Pensamento Computacional na formação de professores nas suas áreas de atuação. Integrar o Pensamento Computacional na Licenciatura de Matemática possibilitará desenvolvê-lo para que os futuros docentes consigam trabalhar estas habilidades em sala de aula.

No ensino da Matemática os problemas envolvem situações e conhecimentos matemáticos e, para serem resolvidos, exigem a elaboração de estratégias mentais adequadas à sua resolução (SILVER, 2006). Na elaboração de estratégias mentais à resolução de problemas é que o Pensamento Computacional pode contribuir, pois, conforme Wing (2006), é uma forma de resolução de problemas, usando algoritmos, métodos e fundamentos da computação aplicados a outras áreas do conhecimento.

Relacionado às questões matemáticas que podem surgir na sala de aula, a oportunidade ao futuro professor em vivenciar, experimentar e propor suas próprias práticas por meio do Pensamento Computacional, poderão ir ao encontro de habilidades citadas por Silver (2006), como: escolher a representação matemática mais apropriada ao momento de aprendizagem; perceber e aceitar ideias diferentes propostas pelos alunos para encontrar a solução; e determinar os dados de um problema.

O enfoque educacional deve ser no desenvolvimento da capacidade de resolução de problemas de forma ampla, em áreas temáticas ou domínios inseridos na cultura específica (CASE, 1989). “Quando se tem um bom ensino, percebe-se que este processo na realidade é totalmente recíproco” (CASE, 1989, p. 328). Nos processos de ensino e de aprendizagem o professor também aprende e, com isto, pode aprimorar suas práticas.

A aprendizagem de padrões, conteúdo de álgebra, é uma forma de desenvolver habilidades matemáticas ao propiciar ao aprendiz o desenvolvimento da generalização e o estabelecimento de conjecturas. O raciocínio algébrico é uma forma de pensar que supõe o estabelecimento de generalizações e regularidades em diversas situações matemáticas. É um processo no qual os estudantes generalizam ideias matemáticas a partir de um conjunto particular de instâncias, estabelecendo-a por meio de uma argumentação e a expressando formalmente, conforme as características cognitivas de sua idade (BLANTON; KAPUT, 2005). “Este tipo de raciocínio está no coração da matemática concebida como a ciência dos padrões” (GODINO; FONT, 2003, p. 774).

A procura de padrões e regularidades e a formulação de generalizações em diversas situações, devem ser fomentadas desde os primeiros anos da educação básica (NCTM, 2000). Generalizar é o processo pelo qual, a partir de um conjunto de casos particulares, o raciocínio continua além deste conjunto de casos, identificando a regularidade entre eles, fazendo a generalização da ideia por meio do discurso e/ou da expressão formal (BLANTON; KAPUT, 2005).

A Base Nacional Comum Curricular (BNCC) de 2018 prevê, na etapa do Ensino Fundamental, na área da Matemática, que os processos matemáticos de resolução de problemas e de investigação são ricos para o desenvolvimento de competências para

o letramento matemático e para o desenvolvimento do Pensamento Computacional (BRASIL, 2018). Das unidades temáticas definidas na BNCC, na Matemática para o Ensino Fundamental a Álgebra visa a desenvolver o pensamento algébrico, e, para tal:

[...] é necessário que os alunos identifiquem regularidades e padrões de sequências numéricas e não numéricas, estabeleçam leis matemáticas que expressem a relação de interdependência entre grandezas em diferentes contextos, bem como criar, interpretar e transitar entre as diversas representações gráficas e simbólicas, para resolver problemas por meio de equações e inequações, com compreensão dos procedimentos utilizados (BRASIL, 2018, p. 270).

Na etapa do Ensino Médio, a BNCC traz diferentes itinerários formativos, e um deles é a Matemática e suas Tecnologias:

[...] os estudantes devem consolidar os conhecimentos desenvolvidos na etapa anterior e agregar novos, ampliando o leque de recursos para resolver problemas mais complexos, que exijam maior reflexão e abstração. Também devem construir uma visão mais integrada da Matemática, da Matemática com outras áreas do conhecimento e da aplicação da Matemática à realidade (BRASIL, 2018, p. 471).

Neste contexto, o desenvolvimento do Pensamento Computacional, na formação inicial de professores de matemática, situou-se nesta investigação, procurando a integração dessas áreas do conhecimento.

Para verificar como este tema tem sido previsto na formação inicial do professor de Matemática atualmente, foi realizado um levantamento dos cursos de Licenciatura em Matemática do Estado do Rio Grande do Sul. Ao todo foram 29 cursos presenciais listados na plataforma do e-MEC⁴. Buscaram-se os Projetos Políticos Pedagógicos (PPC) dos cursos nos portais e, para aqueles que não o disponibilizam na íntegra, as informações sobre os cursos em seus portais, como a matriz curricular, disciplinas, perfil e objetivos.

Na investigação realizada, a articulação do Pensamento Computacional com o ensino na formação inicial de professores de Matemática não foi encontrada em qualquer um dos cursos. Em virtude da publicação da Resolução n. 2, de 20 de dezembro de 2019, com as Diretrizes Curriculares Nacionais para a Formação Inicial de Professores para a Educação Básica e que institui Base Nacional Comum para a Formação Inicial de Professores da Educação Básica (BRASIL, 2019), que prevê a inserção do Pensamento Computacional às licenciaturas, possivelmente em breve, haverá a adequação dos cursos às diretrizes do Ministério da Educação.

O CAMINHO PERCORRIDO NA PESQUISA

A pesquisa desenvolvida tem abordagem qualitativa, com delineamento do tipo estudo de caso. O estudo de caso parte do desejo da compreensão detalhada de um pequeno número de sujeitos. “[...] o ‘caso’ também pode ser algum evento ou entidade,

⁴ Cadastro Nacional de Cursos e Instituições de Ensino Superior – Cadastro e-Mec. Disponível em: <https://emec.mec.gov.br/>.

além de um único indivíduo. Os estudos de caso têm sido realizados sobre uma ampla variedade de tópicos, incluindo pequenos grupos, comunidades, decisões, programas, mudança organizacional e eventos específicos” (YIN, 2015, p. 33).

O caso, nesta pesquisa, compõe-se pelo grupo de quatro licenciandos em Matemática, de uma Universidade privada, que cursaram a disciplina “Pensamento Computacional e o Ensino de Matemática: uma abordagem sobre padrões”, ofertada de forma extracurricular no segundo semestre do ano de 2019. Os acadêmicos incluídos no grupo foram os que tiveram interesse em cursar a disciplina voluntariamente. O Parecer número 3.565.731, emitido pelo Comitê de Ética em Pesquisa da Universidade, aprovou a presente investigação.

A realização do estudo de caso compreende a existência de múltiplas fontes de evidência (fonte de dados) (YIN, 2015, p. 33). As fontes de evidência compuseram-se por questionários e entrevistas individuais, observação participante, diário de bordo e artefatos produzidos pelos estudantes.

A disciplina teve carga horária de 44 horas presenciais em sala de aula com computadores. A estratégia para trabalhar as habilidades do Pensamento Computacional com conteúdos de Matemática, na formação inicial de professores, foi por meio da linguagem de programação Python. Python foi criada por Guido Van Rossum, um cientista da computação e matemático holandês, e publicamente difundida em 1991. Python foi escolhida, pois não foram identificados trabalhos com essa linguagem na formação inicial de professores e buscou-se investigar como essa articulação poderia acontecer.

Python é uma linguagem de programação de alto nível, ou seja, detalhes da arquitetura do computador não precisam ser considerados para o seu uso, proporcionando uma maior abstração. É considerada uma linguagem de programação de fácil aprendizagem e tem sido utilizada em aplicações para o desenvolvimento de sistemas, interfaces gráficas, jogos, robótica, ciência de dados, bancos de dados, entre outras (ROMANO, 2015). Os licenciandos que participaram da disciplina não possuíam experiência prévia com o Pensamento Computacional ou a linguagem Python.

A análise dos dados coletados, a partir das diferentes fontes de evidência, foi realizada com a teoria de Robbie Case. A Estrutura de Controle Executivo (CASE, 1989) é um dos elementos centrais para representar como os licenciandos elaboraram as estratégias de resolução de problemas em conjunto a um conteúdo da Matemática. Os Processos Reguladores Gerais da Teoria de Robbie Case (Resolução de Problemas, Exploração, Imitação e Regulamentação Mútua) propiciaram o planejamento das aulas e a busca pela compreensão das estratégias de resolução. Outro elemento importante da teoria, para a análise, é a Estrutura Conceitual Central (CASE, 1992), que traz as relações entre os significados envolvidos na resolução de problemas, as relações semânticas entre os diferentes nodos que representam os conceitos gerais e suas representações.

A ARTICULAÇÃO DO PENSAMENTO COMPUTACIONAL NO ENSINO NA FORMAÇÃO INICIAL DE PROFESSORES DE MATEMÁTICA

Na disciplina desenvolvida com os licenciandos em Matemática foram trabalhados aspectos do Pensamento Computacional juntamente com os Padrões e Regularidades, conteúdo de Álgebra. As atividades foram agrupadas em Padrões Numéricos, Números Figurados e Padrões Repetitivos. Neste artigo é descrita a análise de duas atividades sobre Padrões Numéricos.

A Atividade 1 foi sobre os números ímpares que, conforme os pitagóricos, são os números que podem ser divididos em duas partes desiguais (GUNDLACH, 1992). Os estudantes foram instigados a identificar o padrão da sequência a partir da observação da quantidade de vértices da Figura 1, e, nessa atividade, percebeu-se, inicialmente, o processo regulador de Resolução de Problemas de Case, pois cada estudante teve o objetivo de identificar o padrão e precisou traçar estratégias para alcançá-lo.

Figura 1 – Representação visual do padrão da Atividade 1



Fonte: Elaborada pelas autoras.

Como os estudantes organizaram-se em duplas para o desenvolvimento da Atividade 1, discutiram para encontrar a solução e a construíram conjuntamente nas duplas. Também foi observada a ocorrência do processo Regulação Mútua. No prosseguimento da Atividade os estudantes foram questionados: “Há uma equação que define esta sequência numérica?”. As respostas obtidas aconteceram em forma de função, como pode ser observado na Figura 2, que contém a resposta de cada grupo. Ambos os grupos, prontamente constataram que a diferença entre os termos da sequência era o número dois. No planejamento havia sido apontado o cálculo do n -ésimo elemento da sequência, a partir de sua posição n na sequência, com a equação do termo geral $n + n - 1$. Os grupos encontraram outra forma de solução relacionada à Progressão Aritmética.

Figura 2 – Equações para definir a sequência – Atividade 1

$$f(x) = 2 \cdot x + 1$$

$$a_n = \{1, 3, 5, 7, \dots\}$$

(a) Resposta do Grupo A

$$a_m = 2m + 1$$

$$P / m \geq 0$$

$$a_0 = 1 \quad a_2 = 5 \quad a_4 = 9$$

$$a_1 = 3 \quad a_3 = 7$$

(b) Resposta do Grupo B

Fonte: Dados da pesquisa.

Os estudantes foram instigados a desenvolver a Equação de Diferenças que representasse o padrão. Os grupos obtiveram o mesmo resultado: $A_n = A_{n-1} + 2$, para $n \geq 0$ sendo $A_0 = 1$ e cuja resposta está na Figura 3. O padrão da atividade foi identificado naturalmente pelo acréscimo de duas unidades a partir do primeiro elemento.

Figura 3 – Equações para definir a sequência – Atividade 1

$$r = 2$$

$$a_n - a_{n-1} = 2 \forall n$$

(a) Resposta do Grupo A

$$a_n = a_{n-1} + 2 \text{ (equação da diferença)}$$

$$a_m = a_{m-1} + 2$$

$$P / m \geq 0$$

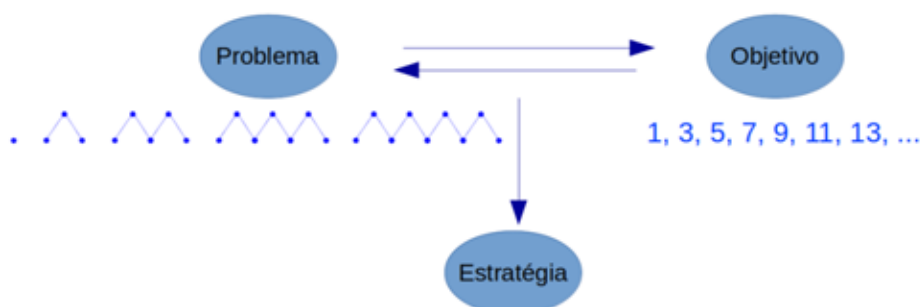
(b) Resposta do Grupo B

Fonte: Dados da pesquisa.

A integração do Pensamento Computacional e do Pensamento Matemático, aconteceu com a equação do termo geral, introduzida para produzir a sequência dos números ímpares em Python. Para caracterizar a estratégia de solução, esta é constituída como a Estrutura de Controle Executivo (ECE) (CASE, 1989) para cada grupo.

Na Atividade 1 o problema na ECE representa a situação concreta ante a qual o sujeito encontra-se, como o padrão ilustrado na Figura 1. O objetivo é a sequência numérica que representa o padrão. A estratégia da ECE é o modo como a solução do problema foi construída, mostrando a sequência de passos e sua aplicação para a solução. A representação da estratégia desenvolvida pelo grupo A está na Figura 4.

Figura 4 – Estratégia desenvolvida pelo Grupo A na Atividade 1

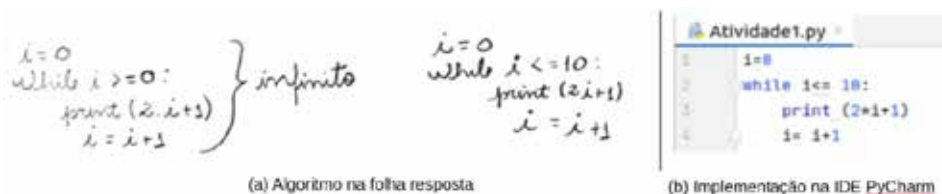


- 1) Identificaram o padrão representado, os números ímpares.
- 2) Associaram a uma Progressão Aritmética e identificaram a razão 2.
- 3) Definiram a equação $f(x) = 2x + 1$, como o termo geral.
- 4) Definiram a variável de controle para o algoritmo, variável i .
- 5) Escolheram o `while` para a estrutura de repetição.
- 6) O critério de parada do laço foi $i >= 10$, para a produção infinita de elementos, ou $i <= 10$ para a produção finita de elementos da sequência.
- 7) Em cada iteração, do laço de repetição, mostraram o elemento com `print(2*i+1)`, em que usaram a equação definida no item 3).
- 8) Incrementaram a variável de controle i , para seguir as iterações no laço de repetição.

Fonte: Elaborada pelas autoras.

O grupo A produziu a sequência dos números ímpares a partir do primeiro termo, o elemento 1, infinitamente. Usaram uma variável de controle para produzir cada elemento em um laço de repetição do tipo `while`. A cada iteração do laço mostraram o elemento na tela, por meio da equação do termo geral (Figura 2), escrita como $2 * i + 1$ e inserida no comando: `print()`. A Figura 5 tem a solução na folha resposta e no ambiente do PyCharm⁵.

Figura 5 – Código instituído em Python pelo Grupo A na Atividade 1



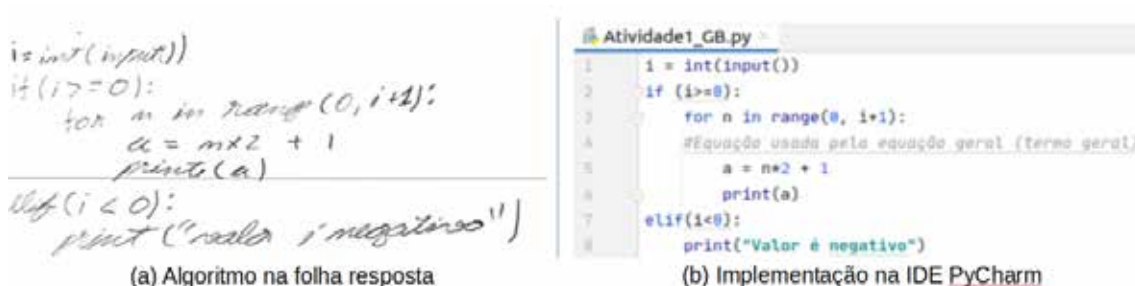
Fonte: Dados da Pesquisa.

⁵ PyCharm é o ambiente de desenvolvimento integrado (*Integrated Development Environment – IDE*) usado para programar na linguagem Python.

Na entrevista, após o desenvolvimento de toda a atividade, o grupo A, ao ser questionado se a sequência era gerada e se terminava, percebeu que o laço de repetição estava sendo executado infinitamente. Então, começaram a procurar o que poderiam fazer e concluíram que para o caso de a sequência finalizar, deveriam alterar o critério de parada do laço para algum valor específico. Na resposta alteraram para $i \leq 10$.

O grupo B produziu a sequência numérica dos números ímpares, também a partir do primeiro termo, o elemento 1, conforme a solução que está na Figura 6. Nesta, solicitaram ao usuário quantos elementos da sequência seriam produzidos (comando $i = \text{int}(\text{input}())$). Para o controle do término da sequência, usaram a função *range* do Python, que permite trabalhar com uma faixa de valores definida em um intervalo, como no código *range* (0, $i+1$). A cada iteração do laço de repetição executaram a equação do termo geral definida para calcular e mostrar o termo da sequência.

Figura 6 – Código instituído em Python pelo Grupo B na Atividade 1



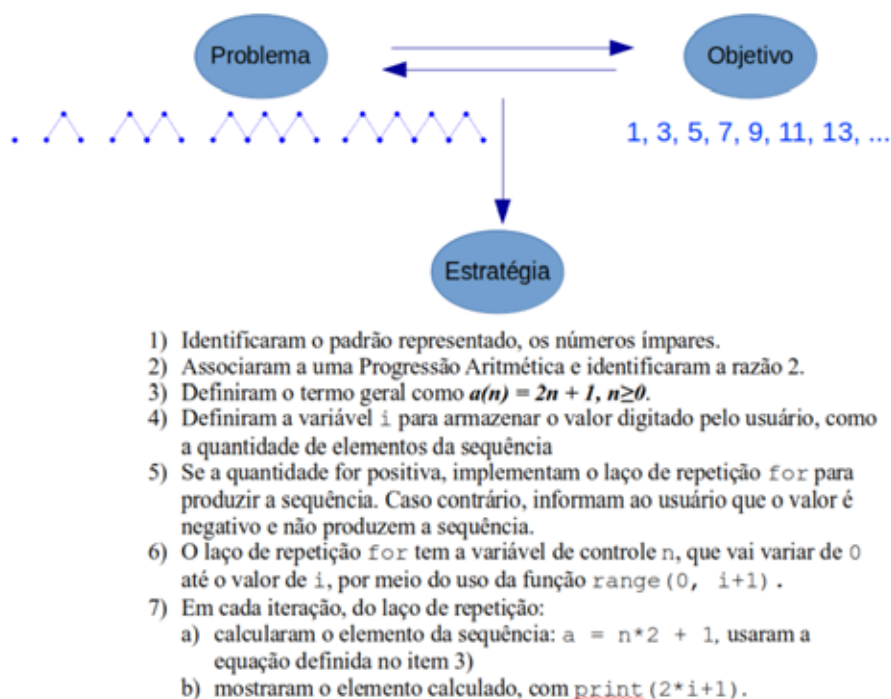
Fonte: Dados da Pesquisa.

Ao serem questionados “*Como fizeram?*”, responderam, apontando para o código desenvolvido: “*Ler o i . Colocar no for para fazer a equação. Colocar condição para ≥ 0* ”. No laço de repetição, comentaram o código relatando: “*Equação usada pela equação geral (termo geral)*”. Na função *range*, colocaram como valor final $i + 1$, pois perceberam que a faixa de valores (o término) é um intervalo aberto. Por fim, ao serem questionados sobre o uso de uma estrutura condicional, consideraram: “*Usamos o if e else por questões de segurança*”, fazendo referência à necessidade de serem inseridos apenas valores positivos para indicar a posição do elemento na sequência. A estratégia empregada para a solução, pelo grupo B, na forma da Estrutura de Controle Executivo, de Case, está na Figura 7.

Os algoritmos desenvolvidos pelos grupos e introduzidos na linguagem Python produzem a sequência numérica dos números ímpares: 1, 3, 5, 7, 9, 11, 13, ... O grupo A predefiniu a quantidade de números da sequência, fazendo o laço de repetição executar 11 iterações, de zero a 10 inclusive, portanto a saída do código produziu os 11 primeiros números ímpares. Inicialmente produziram infinitamente os elementos, pois a condição do laço de repetição estava $i \geq 0$. O grupo B desenvolveu o algoritmo de forma a solicitar ao usuário a quantidade de números a serem produzidos (foi inserido o comando para ler esta informação, porém não foi emitida qualquer mensagem ao usuário). Além disso, o grupo validou, no programa, se o número informado era positivo, caso contrário, mostravam que o valor era negativo e a sequência não poderia ser produzida.

Os grupos usaram estrutura de repetição para criar. O primeiro grupo foi o laço *while*, e o segundo o *for*. Sobre o pensamento algébrico nas soluções encontradas pelos grupos às questões propostas, percebeu-se que foi similar. As estratégias desenvolvidas foram diferentes e alcançaram a solução do problema.

Figura 7 – Estratégia desenvolvida pelo Grupo B na Atividade 1

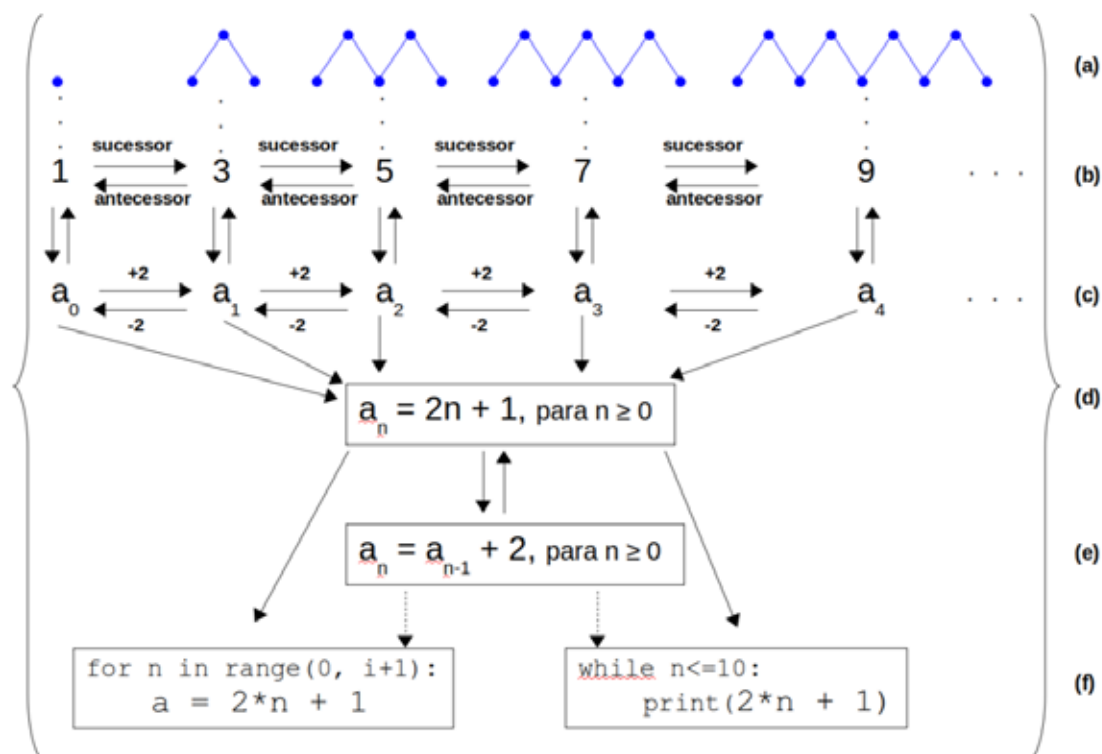


Fonte: Dados da Pesquisa.

Com o desenvolvimento da Atividade 1, pelos estudantes, foi delineada a rede de nodos e seus relacionamentos semânticos, semelhante à Estrutura Conceitual Central (CASE, 1992). A abordagem na rede de nodos foi elaborada pelas pesquisadoras, conforme a construção da Atividade 1 dos estudantes, e está ilustrada na Figura 8.

Na Figura 8, linha (a), tem-se a representação visual da sequência apresentada aos estudantes, a fim de que, considerando o número de vértices, encontrassem o padrão. A linha (b) contém a representação numérica da sequência e os números ímpares escritos com os algarismos arábicos. Na linha (c) está a representação de cada termo da sequência na linguagem simbólica matemática. Na linha (d), da mesma figura, ao invés da representação de cada termo, tem-se a generalização da sequência por meio da equação do termo geral; os estudantes definiram o primeiro termo como sendo o elemento de índice zero. Da mesma forma, a generalização encontra-se na linha (e) com a equação de diferenças, e, em razão disso, há um caminho de duas vias entre as linhas (d) e (e).

Figura 8 – Rede de nodos na Atividade 1



Fonte: Dados da Pesquisa.

Em virtude da abstração realizada, chegaram às instruções na linguagem Python para a produção da sequência. As duas formas encontradas pelos estudantes estão na rede de nodos, na linha (f). Para a efetivação foi usada a equação da linha (d), mas poderia ter sido usada a equação da linha (e), por isso as setas pontilhadas da linha (e) à linha (f), indicando que isto pode ser feito.

As setas horizontais entre os elementos de cada linha, na Figura 8, indicam o crescimento da sequência e também o caminho do decrescimento, bem como a diferença entre os elementos (+2 ou -2). Há um caminho, de duas vias, entre as linhas (b) e (c), pois, dada a sequência dos ímpares, pode ser que o estudante diretamente a represente na linguagem simbólica.

Há semelhança da linguagem simbólica matemática com a linguagem de programação Python na expressão aritmética para produção do termo da sequência em relação à equação do termo geral, assim como a variação do n na sequência, em que, na generalização, tem-se para maior ou igual a zero, e na linguagem de programação definido no laço de repetição `for i n in range (0, 10)`. As chaves externas trazem a ideia da estrutura de nodos como um todo, constituindo a integração dos pensamentos algébrico e computacional.

Na Atividade 2 estavam presentes três estudantes e cada um optou por trabalhar em seu computador e interagir com os colegas para trocar ideias (processo regulador Regulação Mútua). Em vários momentos precisaram experimentar, por meio da instalação do programa, estratégias para resolver, a fim de alcançar a solução do problema. O

processo Regulação Mútua aconteceu em conjunto com o processo Resolução de Problemas e também com o processo Exploração, na experimentação de situações diferentes no código em Python e verificação do resultado obtido.

Nesta atividade, a sequência numérica 2, 6, 12, 20, 30... foi apresentada aos estudantes e não sua representação visual, como na Atividade 1. A partir da observação da sequência, foi solicitado “Qual o padrão para formação da sequência de números?” O estudante E1 comentou: “a sequência é 1×2 , 2×3 , 3×4 , 4×5 , 5×6 ...”. Os estudantes discutiram sobre a questão e concluíram que não devia ser isto, porém não perceberam que poderia ser definida a equação do termo geral a partir da conjectura do estudante E1. Houve interação entre os três estudantes. O estudante E2 responde: “outra forma: é sempre o 4 mais $2 \times n$. É o anterior mais o 4” e continuam neste sentido, encontrando a equação das diferenças, como está na Figura 9, que contém a resposta do estudante E1 e que foi construída conjuntamente pelos estudantes.

Figura 9 – Equação que define o padrão da Atividade 2

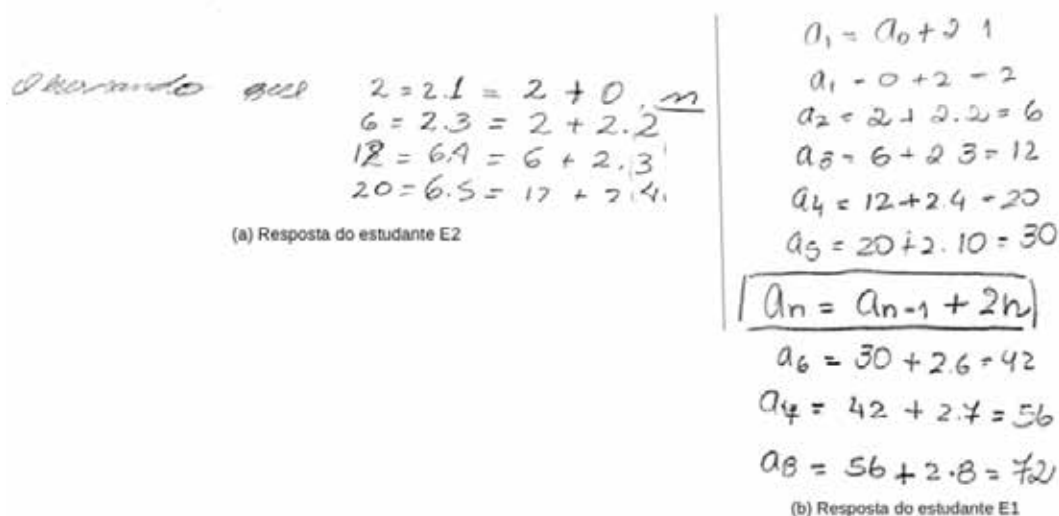
$$a_n = a_{n-1} + 2 \cdot n, n \geq 1$$

Fonte: Dados da Pesquisa.

A definição da equação de diferenças, da Figura 9, não foi feita de imediato pelos estudantes. Houve discussão e exploração das possibilidades para compor a sequência numérica até ser possível a generalização. A resposta dos estudantes à questão “Como obtiveram os resultados?”, mostra o processo percorrido até chegar à generalização.

Observou-se que o E1, em sua resposta, iniciou buscando identificar a formação dos primeiros elementos a_1, a_2, \dots, a_n , assim como o estudante E2, até alcançar a generalização (Figura 10). E1 também explicou a produção dos elementos da sequência como na sua primeira conjectura (e que estava correta): “Observando o crescimento: $a_1 = 1 \times 2$, $a_2 = 2 \times 3$, $a_3 = 3 \times 4$, $a_4 = 4 \times 5$, $a_5 = 5 \times 6$... onde o termo da sequência é multiplicado pelo seu sucessor...”, de forma que a obtenção de cada termo da sequência poderia ter sido expressa por $n * (n + 1)$, em que n indica a posição do termo na sequência, chegando, assim, à equação do termo geral. O E1 não escreveu a equação do termo geral algebricamente na questão A, mesmo tendo sido esta a sua primeira conjectura, pois, com a interação com os colegas, tomaram outro caminho para a solução, chegando à equação das diferenças. O processo descrito pelo E2 foi semelhante, pois considerou, inicialmente, o $2 = 2 \times 1 = 2 + 0 \times n$; $6 = 2 \times 3 = 2 + 2 \times 2$; no elemento de valor 12 houve um equívoco, pois, ao invés de ser 3×4 estava 6×4 , porém o passo para a equação de diferenças está correto: $6 + 2 \times 3$. Esse mesmo equívoco aconteceu para o elemento de valor 20. Mesmo assim, a generalização foi obtida.

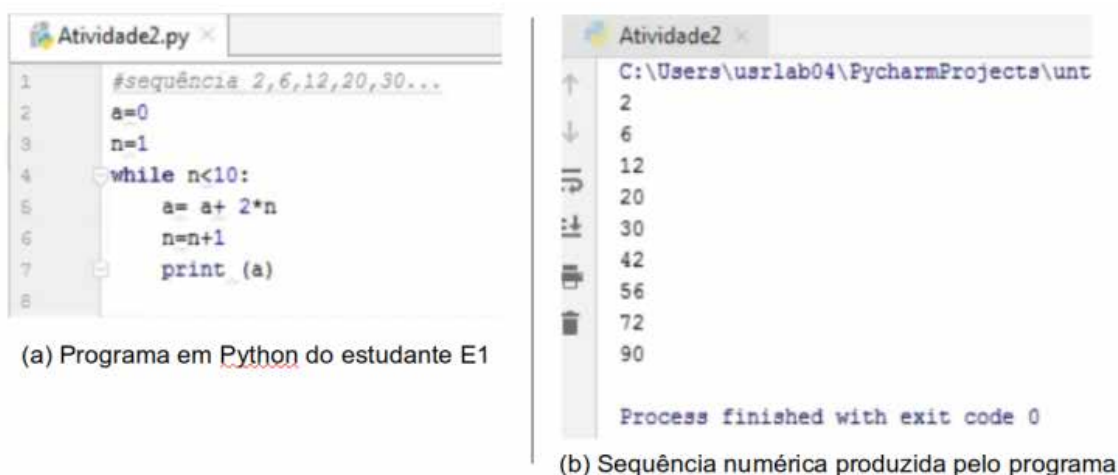
Figura 10 – Processo de generalização desenvolvido pelos estudantes na Atividade 2



Fonte: Dados da Pesquisa.

A equação de diferenças foi usada no desenvolvimento do algoritmo para produzir o padrão numérico, como nas respostas da questão “Como produzir esta sequência em Python?” O estudante E1 desenvolveu o algoritmo em Python da Figura 11(a), e a produção da sequência numérica, pelo programa, está no item (b) da mesma Figura.

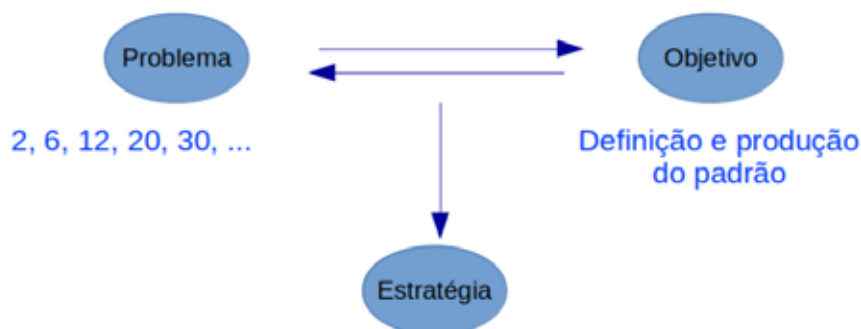
Figura 11 – Produção da sequência da Atividade 2 desenvolvida pelo estudante E1



Fonte: Dados da Pesquisa.

O estudante E1 comentou o uso da equação de diferenças ao ser questionado se a definição da sequência foi empregada e como foi utilizada; ele respondeu: “Sim, com a variação do ‘a’.” Assim, houve relação da equação de diferenças com a expressão aritmética $a = a + 2 \times n$, usada no programa em Python, em que a variável a , inicialmente, tem o valor 0 e a cada iteração é usado o seu valor para calcular o valor do n -ésimo termo da sequência. A estratégia para solução e produção da sequência do padrão da Atividade 2 está na Figura 12, ilustrada de forma semelhante à ECE de Case (1989).

Figura 12 – Estratégia desenvolvida pelo estudante E1 na Atividade 2



- 1) A partir dos casos particulares: 1×2 , 2×3 , 3×4 , 4×5 , $5 \times 6 \dots$ ou seja, 2, 6, 12, 20, 30...
- 2) Construiu a generalização da sequência: $a_n = a_{n-1} + 2n$
- 3) Criou a variável a inicializada com *zero* e que no laço de repetição recebe o valor calculado do termo na sequência
- 4) Criou a variável n para controlar as iterações e consequentemente, o número do termo na sequência
- 5) Escolheu o laço de repetição `while`, com a condição de parada $n \leq 10$.
- 6) Em cada iteração, do laço de repetição, calculou o n -ésimo termo da sequência com a expressão $a = a + 2 * n$, usou a equação definida no item 2).
- 7) Incrementou a variável n , para seguir as iterações no laço de repetição, enquanto $n \leq 10$.
- 8) Mostrou o valor de a (o termo da sequência), com comando `print (a)`.

Fonte: Dados da Pesquisa.

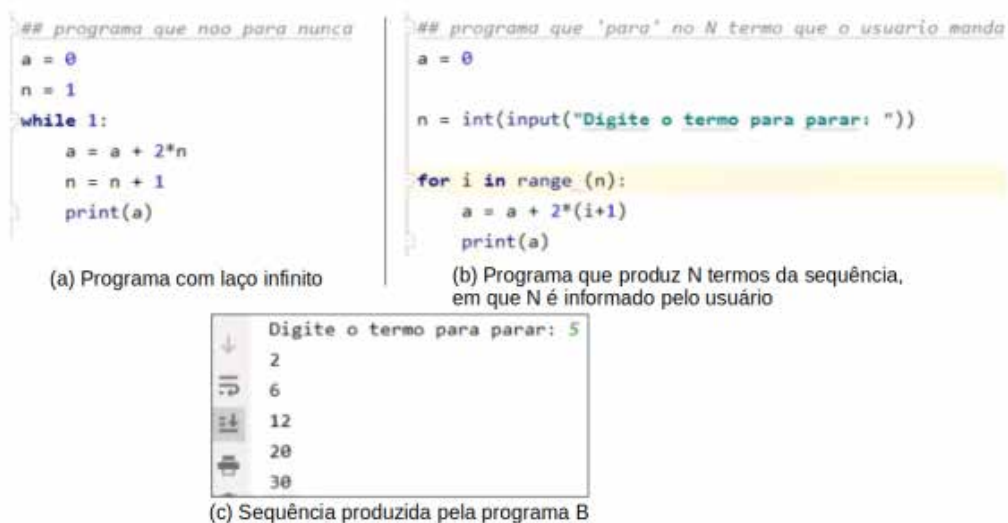
O estudante E2, ao ser perguntado, afirmou usar a equação que define a sequência para o desenvolvimento do programa em Python. Ele elaborou duas versões para produção. A primeira com o laço `while`, em que o algoritmo produziu infinitamente a sequência, ou seja, não houve um número de repetições preestabelecido, ocasionando a execução infinita do programa. Após perceber isto, refletiu juntamente com o E4 e começaram a experimentar outras formas de solução. Modificaram o programa para solicitar ao usuário quantos termos gostaria de produzir, então a repetição ficou finita até o número de termos informado pelo usuário. A segunda versão foi com o laço de repetição `for`.

Em razão das duas versões elaboradas, na resposta do estudante E2 à questão “No desenvolvimento do programa na linguagem Python, a equação da questão E foi utilizada? Se sim, como?”, ele relacionou às versões: “Para o programa `while` sim, sem nenhuma modificação. Em quanto que no que o usuário informa o no. de termos, deve-se modificar o i , pois ele inicia em zero.”⁶. A transferência da representação simbólica para o programa desenvolvido na linguagem Python foi direta na primeira versão e houve habilidade para modificá-la, mantendo a sequência gerada e alterada a forma de interação com o usuário, instituindo a segunda versão do programa.

Os programas desenvolvidos estão na Figura 13: a versão com laço infinito (E2) e a versão que solicita ao usuário até qual termo deseja produzir a sequência (E2 e E4). A sequência produzida pela segunda versão do programa está na Figura 13(c).

⁶ A transcrição da resposta do estudante é fiel ao que foi expresso por ele na folha da Atividade 2.

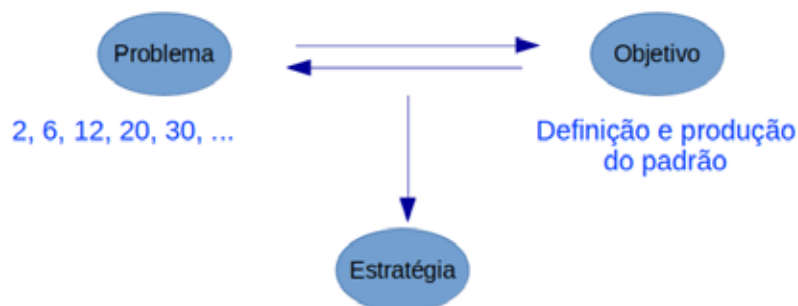
Figura 13 – Produção da sequência da Atividade 2 desenvolvida pelos estudantes E2 e E4



Fonte: Dados da Pesquisa.

Percebeu-se no desenvolvimento do algoritmo para gerar a sequência, que a generalização foi empregada no laço de repetição *while*. Na primeira versão o estudante E2 deixou o laço infinito, pois o critério de parada nunca atingiria o resultado lógico falso. A estratégia de E2 está na Figura 14.

Figura 14 – Estratégia desenvolvida pelo estudante E2 na Atividade 2

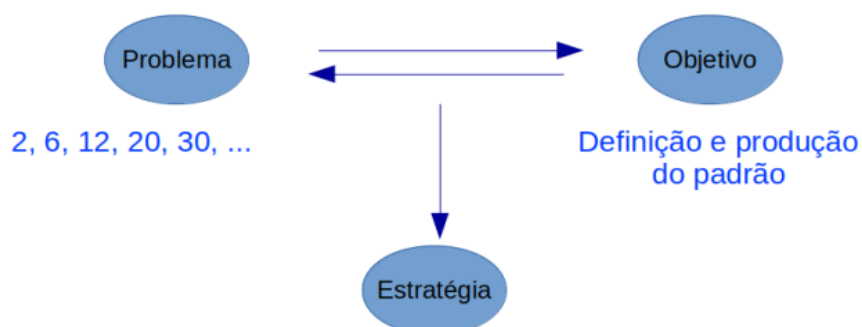


- 1) A partir dos casos particulares:
 - $2 = 2 \times 1 = 2 + 2 \times 1$
 - $6 = 2 \times 3 = 2 + 2 \times 2$
 - $12 = 3 \times 4 = 6 + 2 \times 3$
 - $20 = 4 \times 5 = 12 + 2 \times 4$
- 2) Percebeu que os números destacados em negrito, acima, correspondem à posição do termo na sequência, o que possibilitou generalizar.
- 3) Generalização da sequência: $a_n = a_{n-1} + 2n$
- 4) Criou a variável *a* inicializada com *zero* e que no laço de repetição recebe o valor calculado do termo na sequência
- 5) Criou a variável *n* para contar o número do termo na sequência
- 6) Escolheu o laço de repetição *while*, com a condição de parada *1*, ou seja, o laço repetirá infinitamente.
- 7) Em cada iteração, do laço de repetição, calculou o *n*-ésimo termo da sequência com a expressão $a = a + 2 * n$, usou a equação definida no item 3).
- 8) Incrementou a variável *n*, para seguir a produção dos termos
- 9) Mostrou o valor de *a* (o termo da sequência), com comando `print(a)`.

Fonte: Dados da Pesquisa.

A segunda versão desenvolvida pelos estudantes E2 e E4 incluiu uma quantidade fixa de repetições, que é informada pelo usuário do programa para produzir um número determinado de termos da sequência. A estratégia para esta solução encontra-se na Figura 15. Foi empregado o laço de repetição *for*, por eles considerado apropriado, uma vez que pode fazer uso da função *range(n)*, definindo um intervalo de valores, de 1 até $(n - 1)$. Esta solução pode ser observada na Figura 13(b).

Figura 15 – Segunda estratégia desenvolvida pelos estudantes E2 e E4 na Atividade 2



- 1) Com a generalização da sequência definida: $a_n = a_{n-1} + 2n$
- 2) Criou a variável *a* inicializada com *zero* e que no laço de repetição recebe o valor calculado do termo na sequência
- 3) Criou a variável *n* para armazenar a quantidade de itens informada pelo usuário
- 4) Escolheu o laço de repetição *for*, fazendo a variável *i* percorrer o intervalo de 0 até *n*, com o uso da função *range*, com a instrução `for i in range(n)`:
- 5) Em cada iteração, do laço de repetição, calculou o *n*-ésimo termo da sequência com a expressão $a = a + 2 * (i+1)$, usou a equação definida no item 1).
- 6) Mostrou o valor de *a* (o termo da sequência), com comando `print(a)`.

Fonte: Dados da Pesquisa.

Para finalizar a Atividade 2 os estudantes foram desafiados a modificar o algoritmo desenvolvido, para que a sequência fosse ilustrada ou mostrada, visualmente, de alguma forma. Para resolver, os estudantes começaram a multiplicar a quantidade representada pelo termo na sequência pelo seu próprio nome e viram que seu nome aparecia *n* vezes na tela, o que tornou a descoberta lúdica. Nesse processo de exploração trocaram o seu nome por um símbolo específico. A alteração na estratégia de solução, pelo estudante E1, no código em Python, foi somente na linha do `print(a)`, em que foi incluída a multiplicação pelo símbolo '@', obtendo o comando `print(a * '@')`. O estudante E2 usou o símbolo '#' e o estudante E4 optou pelo símbolo '\$', fazendo com que em que cada linha da tela, houvesse a quantidade de símbolos de cada elemento da sequência.

Na Atividade 2, na integração do Pensamento Algébrico e do Pensamento Computacional, observou-se a abstração, a generalização, o desenvolvimento do algoritmo e sua efetivação, além da articulação de diferentes formas de representação do padrão: numérica, algébrica, algorítmica e visual.

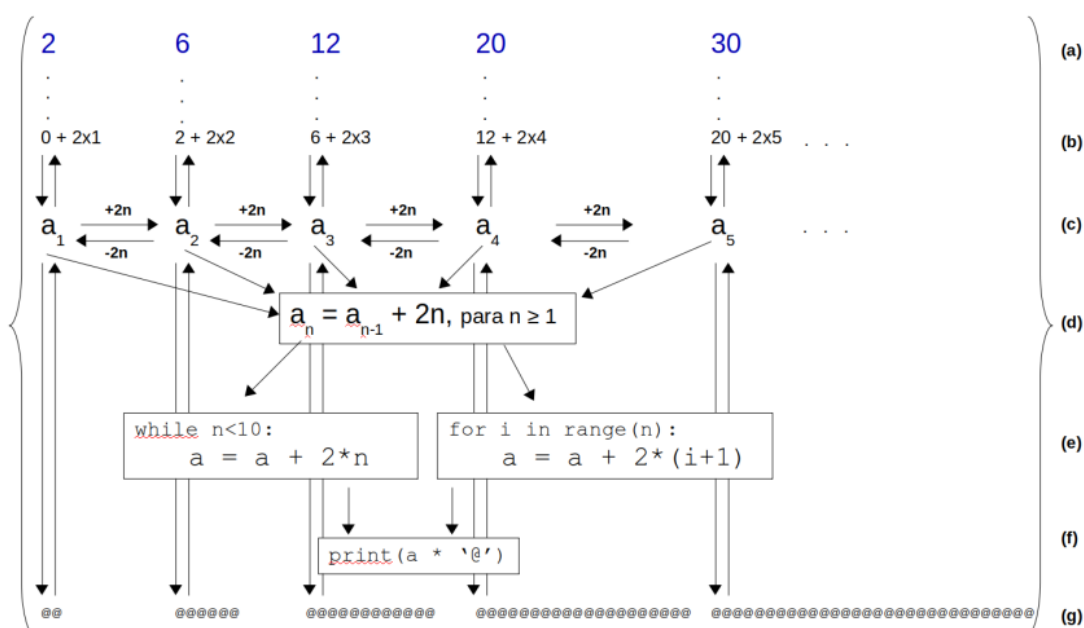
Uma rede de nodos foi elaborada, pelas pesquisadoras, conforme a construção da Atividade 2 feita pelos estudantes, e está representada na forma da Estrutura de Controle Conceitual (CASE, 1992). Na linha (a), da Figura 16, que contém a rede de nodos, está a sequência numérica apresentada aos estudantes. Na linha (b) está a decomposição dos números da sequência, conforme elaborado pelos estudantes, representando

os casos particulares até se chegar à generalização expressa pela equação de diferenças da linha (d). Antes disso, na linha (c) está a representação algébrica de cada elemento da sequência. Há setas verticais de duas vias entre as linhas (b) e (c) e também entre as linhas (c) e (g), indicando as diferentes formas de representar o elemento: numérica, algébrica e visual.

Na linha (c), da mesma figura, há setas horizontais indicando a transição entre os elementos da sequência e sua diferença $+2n$ para o caso de ascender na sequência ou $-2n$, ao descender. A linha (e) contempla as duas formas de execução realizadas pelos estudantes, com a equação da linha (d). Na linha (f) está a linha do código em Python para produção da quantidade de símbolos conforme o valor do elemento da sequência. Na linha (g) está a representação visual de cada elemento da sequência produzida pelo programa executado.

Por fim, as chaves externas, na rede de nodos, trazem a ideia da compreensão das relações, como um todo, das diferentes representações da sequência da Atividade 2 e as associações entre elas, construídas conforme as evidências na resolução da atividade.

Figura 16 – Rede de nodos na Atividade 2

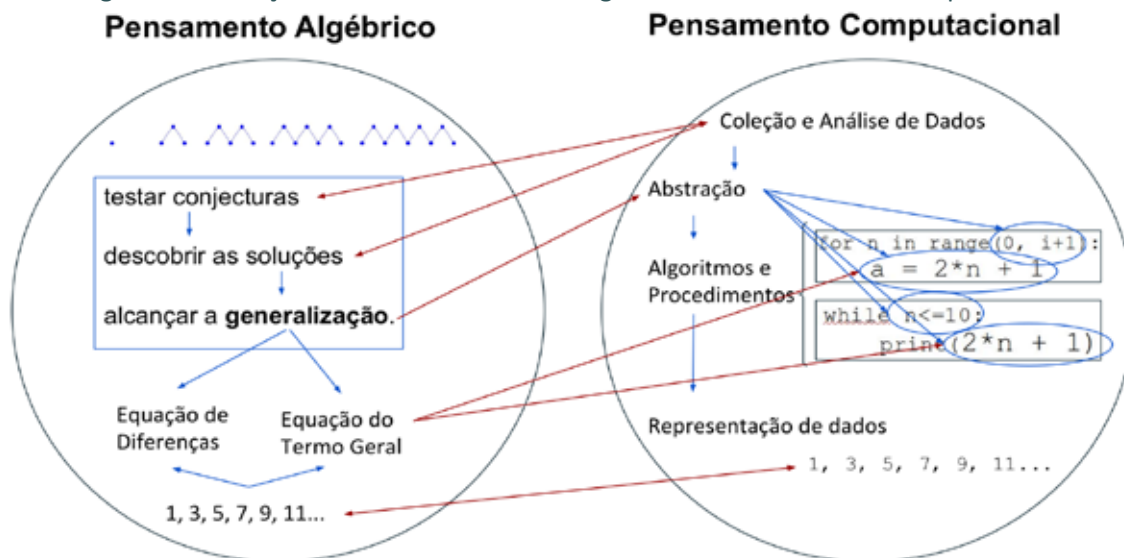


Fonte: Elaborada pelas autoras.

As habilidades do Pensamento Computacional evidenciadas nas atividades foram abstração, algoritmos e procedimentos e coleção, análise e representação de dados.

Entende-se por abstração a estratégia de simplificação, a fim de reduzir a complexidade para definir a ideia principal, podendo encapsular detalhes sem perder a generalidade. Os algoritmos e procedimentos compõem-se pela sequência de passos encadeados e ordenados para resolução de um problema. A coleção de dados envolve a coleta dos dados para a resolução, enquanto a análise de dados consiste em encontrar sentido para os dados coletados e padrões, e a representação de dados abrange as diferentes formas adequadas em que os dados podem ser representados (por exemplo, gráficos, imagens, diagramas, tabelas, símbolos, entre outros) (CSTA; ISTE; NSF, 2011).

Figura 17 – Relações entre o Pensamento Algébrico e o Pensamento Computacional



Fonte: Elaborada pelas autoras.

A Articulação do Pensamento Algébrico e do Pensamento Computacional nas atividades proporcionou aos estudantes, ao testar conjecturas e ir descobrindo as soluções algébricas, trabalharem com habilidades como a coleção e análise de dados. Ao alcançarem a generalização a abstração esteve presente, assim como ao definirem o algoritmo iterativo para produção dos padrões, e a abstração esteve presente na estrutura dos laços de repetição como critério de parada desses laços.

A transferência da representação algébrica da generalização para a representação na linguagem de programação Python, a fim de calcular cada elemento da sequência, por meio das equações do termo geral ou das diferenças definidas pelos estudantes, evidenciou o emprego da abstração. A representação de dados foi observada nas diferentes formas de apresentar o padrão: numérica, algébrica, visual. A representação simbólica do Pensamento Matemático para a linguagem Python é praticamente direta, como observa-se na Figura 17, em que estão as equações e o respectivo comando na linguagem de programação.

CONSIDERAÇÕES FINAIS

Com a investigação realizada procurou-se integrar o Pensamento Computacional no ensino da Matemática, embasado na resolução de problemas conforme a teoria de Robbie Case. Com o estudo de caso na formação inicial, foi possibilitado aos licenciandos em Matemática experimentar, resolvendo problemas de padrões e regularidades, o Pensamento Computacional.

Os padrões e regularidades permeiam a Matemática, estão presentes nas diferentes etapas de ensino e, nesta investigação, o Pensamento Computacional foi trabalhado junto a esse conteúdo de Álgebra, como forma de prover os processos de investigação e resolução de problemas, como referidos na BNCC. Ressalta-se que o Pensamento Computacional também pode ser associado a outros conteúdos matemáticos.

Os processos reguladores gerais da teoria de Robbie Case proporcionaram ir em busca da compreensão da resolução de problemas desenvolvida pelos acadêmicos com o Pensamento Computacional. A identificação das estratégias utilizadas e a rede de conceitos, representações e significados nas resoluções, foram expressos, respectivamente, por meio da estrutura de controle executivo e da estrutura conceitual central de Case.

As relações estabelecidas entre o Pensamento Computacional e o Pensamento Algébrico, nas duas atividades analisadas, trazem a abstração, a generalização, a manipulação de dados e diferentes formas de representá-los. Na continuidade do trabalho haverá a análise de outras atividades da disciplina ofertada, com a resolução de problemas e também a criação destes pelos licenciandos. Identificou-se, até o momento, como limitação da investigação, um maior espaço para as discussões e reflexões entre os estudantes, o que possibilitaria maiores trocas de experiências.

O ensino de Matemática articulado ao Pensamento Computacional é uma abordagem necessária à formação de professores, dadas as constantes mudanças na sociedade. Conforme a simetria invertida, as experiências vivenciadas pelo futuro professor, em sua formação, servem a ele como referência para suas práticas. A presente abordagem traz um caminho para trabalhar o Pensamento Computacional no ensino, de maneira transversal com a Matemática. O Pensamento Computacional é uma habilidade essencial, como a leitura e a escrita, no cenário atual, pois tarefas rotineiras e mecânicas no trabalho tendem a dar espaço à resolução de problemas cada vez mais complexos.

REFERÊNCIAS

- AVILA, C. *et al.* Desdobramentos do pensamento computacional no Brasil. In: SIMPÓSIO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO, 27., 2016. Uberlândia. *Anais [...]*. Uberlândia: Sociedade Brasileira de Computação, 2016.
- AVILA, C. *et al.* O pensamento computacional por meio da robótica no ensino básico – uma revisão sistemática. In: CONGRESSO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO, 6., 2017. Recife. *Anais [...]*. Recife: Sociedade Brasileira de Computação, 2017.
- BARCELOS, T. *et al.* Relações entre o pensamento computacional e a matemática: uma revisão sistemática da literatura. In: CONGRESSO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO, 4., 2015. Maceió. *Anais [...]*. Maceió: Sociedade Brasileira de Computação, 2015.
- BARCELOS, T.; BORTOLETTO, R.; ANDRIOLI, M. G. Formação online para o desenvolvimento do pensamento computacional em professores de matemática. In: CONGRESSO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO, 5., 2016. Uberlândia. *Anais [...]*. Uberlândia: Sociedade Brasileira de Computação, 2016.
- BLANTON, M. L.; KAPUT, J. J. Characterising a classroom practice that promotes algebraic reasoning. *Journal for Research in Mathematics Education*, v. 36, n. 5, p. 412-446, 2005.
- BOLZAN, D. P. V.; ISAIA, S. M. de A.; MACIEL, A. M. da R. Formação de professores: a construção da docência e da atividade pedagógica na educação superior. *Revista Diálogo Educacional*, v. 13, n. 38, p. 49-68, jan./abr. 2013.
- BRASIL. *Diretrizes Curriculares Nacionais para os cursos de Matemática, Bacharelado e Licenciatura* – Parecer CNE/CES 1.302/2001. Brasília: MEC/CES, 2002. Disponível em: <http://portal.mec.gov.br/cne/arquivos/pdf/CES13022.pdf>. Acesso em: 5 maio 2019.
- BRASIL. *Base Nacional Comum Curricular: educação é a base*. Brasília: MEC, 2018. Disponível em: http://basenacionalcomum.mec.gov.br/images/BNCC_EI_EF_110518_versaofinal_site.pdf. Acesso em: 10 fev. 2019.
- BRASIL. *Resolução n.2, de 20 de dezembro de 2019*. Brasília: MEC, 2019. Disponível em: <http://portal.mec.gov.br/docman/dezembro-2019-pdf/135951-rcp002-19/file>. Acesso em: 2 maio 2020.
- CANSU, F. K.; CANSU, S. K. An overview of computational thinking. *International Journal of Computer Science Education in Schools*, v. 3, p. 3, 2019.
- CASE, R. *El desarrollo intelectual: del nacimiento a la edad madura*. Barcelona: Paidós, 1989.

- CASE, R. *The Mind's Staircase: Exploring the Conceptual Underpinnings of Children's Thought and Knowledge*. Hillsdale: Lawrence Erlbaum Associates, 1992.
- CASE, R. The development of conceptual structures. In: *Handbook of Child Psychology: cognition, perception and language*. 5. ed. New Jersey: Wiley, 1998.
- CSTA; ISTE; NSF. Computer Science Teachers Association; International Society for Technology in Education; National Science Foundation. *Computational Thinking Teacher Resource*. 2011. Disponível em: http://csta.acm.org/Curriculum/sub/CurrFiles/472.11CTTeacherResources_2ed-SP-vF.pdf. Acesso em: 3 set. 2018.
- GODINO, J. D.; FONT, V. *Razonamiento algebraico y su didáctica para maestros*. Granada: Universidad de Granada, 2003.
- GUNDLACH, B. H. *História dos números e numerais*. São Paulo: Atual, 1992.
- MELLO, G. N. Formação inicial de professores para a educação básica: uma (re)visão radical. *São Paulo em Perspectiva*, v. 14, n. 1, p. 98-110, 2000.
- PINHO, G. et al. Pensamento computacional no ensino fundamental: Relato de atividade de introdução a algoritmos. In: WORKSHOP DE INFORMÁTICA NA ESCOLA, 22., 2016. Uberlândia. *Anais [...]*. Uberlândia: Sociedade Brasileira de Computação, 2016. p. 261-270.
- NCTM. Council of Teachers of Mathematics. *Principles and Standards for School Mathematics*. Reston, VA: The National Council of Teachers of Mathematics, 2000.
- RAABE, A. L. A. et al. *Referenciais de formação em computação: educação básica*. 2017. Disponível em: <http://www.sbc.org.br/files/ComputacaoEducacaoBasica-versaofinal-julho2017.pdf>. Acesso em: 10 nov. 2018.
- ROMANO, F. *Learning Python*. Birmingham: PACKT Publishing, 2015.
- SANTOS, P. S. C.; ARAUJO, L. G. J.; BITTENCOURT, R. A. A mapping study of computational thinking and programming in brazilian k-12 education. In: *IEEE Frontiers in Education Conference, FIE2018*. San Jose, CA, USA: IEEE, 2018. p. 1-8.
- SHUTE, V. J.; SUN, C.; ASBELL-CLARKE, J. Demystifying computational thinking. *Educational Research Review*, v. 22, p. 142-158, 2017.
- SILVER, E. A. Formação de professores de matemática: desafios e direções. *Boletim de Educação Matemática*, Rio Claro, SP, v. 19, n. 26, p. 125-152, 2006.
- TAVARES, C. A. G.; SALVADOR, L. N.; VIOLA, D. N. O raciocínio computacional para a educação básica: considerações sobre o ensino de análise combinatória e probabilidade. In: WORKSHOP DE INFORMÁTICA NA ESCOLA, 23., 2017. Recife. *Anais [...]*. Recife: Sociedade Brasileira de Computação, 2017.
- VALENTE, J. A. et al. Alan turing tinha pensamento computacional? reflexões sobre um campo em construção. *Tecnologias, Sociedade e Conhecimento*, v. 4, n. 1, p. 7-22, 2017.
- WEINTROP, D. et al. Defining computational thinking for mathematics and science classrooms. *Journal of Science Education and Technology*, v. 25, p. 127-147, 2016.
- WING, J. M. Computational thinking. *Communications of the ACM*, ACM, New York, NY, USA, v. 49, n. 3, p. 33-35, mar. 2006.
- YIN, R. K. *Estudo de caso: planejamento e métodos*. 5. ed. Porto Alegre: Bookman, 2015.