

# A PRÁTICA CONSTRUCIONISTA E O PENSAMENTO COMPUTACIONAL COMO ESTRATÉGIAS PARA MANIFESTAÇÕES DO PENSAMENTO ALGÉBRICO

## THE CONTRUCTIONIST PRACTICE AND THE COMPUTATIONAL THINKING AS STRATEGIES FOR MANIFESTATIONS OF ALGEBRAIC THINKING

**Rodrigo Duda**

Instituto Federal do Paraná – Campus Irati, rodrigo.duda@ifpr.edu.br

**Nilceia Aparecida Maciel Pinheiro**

Universidade Tecnológica Federal do Paraná – Campus Ponta Grossa, nilceia@utfpr.edu.br

**Sani de Carvalho Rutz da Silva**

Universidade Tecnológica Federal do Paraná – Campus Ponta Grossa, sani@utfpr.edu.br

### **Resumo**

Nesse artigo é abordada a viabilidade da combinação de estratégias referentes ao construcionismo e pensamento computacional para promoção da educação algébrica. Com base em referencial teórico sobre essas vertentes teóricas, discute-se sobre como ambientes de aprendizagem que contemplem a resolução de problemas matemáticos na perspectiva construcionista podem favorecer a explicitação de aspectos necessários para o desenvolvimento do pensamento algébrico. Por fim, aborda-se como o uso de uma plataforma de programação visual pode auxiliar nesse processo. O cenário exposto permite perspectivar que atividades educativas que contemplam o desenvolvimento de aplicativos de execução de cálculos com a ferramenta App Inventor estimulam a manifestação do pensamento algébrico, justificado pelo fato de contemplar abstração, uso de linguagem simbólica e protagonismo discente de forma combinada.

**Palavras-chave:** Construcionismo, pensamento algébrico, pensamento computacional.

### **Abstract**

In this paper we discuss the feasibility of combining strategies related to constructionism and computational thinking to promote algebraic education. Based on a theoretical framework on these theoretical aspects, it is discussed how learning environments that contemplate the resolution of mathematical problems under the constructionist perspective can favor the explicitation of aspects necessary for the development of algebraic thinking. Finally, it is approached how the use of a visual programming platform can help in this process. The exposed scenario allows to realize that educational activities that contemplate the development of applications to perform calculations with the App Inventor tool stimulate the manifestation of algebraic thinking, justified by the fact of contemplating abstraction, use of symbolic language and student protagonism in a combined manner.

**Key-words:** Constructionism, algebraic thinking, computational thinking.

## **Introdução**

Papert (1988, 2008), pioneiro em estudos sobre como uma linguagem de programação pode auxiliar na construção de conceitos em matemática, contribuiu para que uma forma dinâmica de utilização das tecnologias digitais fosse introduzida na sala de aula. Na atualidade essa perspectiva é revisitada devido à diversidade de ferramentas que contemplam programação em atividades educativas, como o desenvolvimento de aplicativos, jogos e a robótica educacional.

Estudos sobre como linguagens de programação podem contribuir para o desenvolvimento de habilidades educacionais indicam possibilidades para a integração do pensamento computacional no ambiente escolar (CHAO, 2016; FESSAKIS; GOULI; MAVROUDI, 2013; SÁEZ-LÓPEZ; ROMÁN-GONZALEZ; VÁZQUEZ-CANO, 2016; LYE; KOH, 2014; ISRAEL et al., 2015; WILKERSON-JERDE, 2014). Essa possibilidade também ganha ênfase na Base Nacional Comum Curricular, ao ser recomendada a abordagem de problemas que estimulem o desenvolvimento do pensamento computacional via atividades educativas relacionadas ao pensamento algébrico (BRASIL, 2017, 2018).

É por essa razão que neste artigo de cunho teórico, decorrente de uma pesquisa de doutorado sobre como o desenvolvimento de aplicativos de execução de cálculo podem auxiliar a desenvolver habilidades em matemática, são abordados aspectos comuns entre o construcionismo (PAPERT, 1988, 2008), o pensamento computacional (WING, 2006, 2008, 2014) e manifestações do pensamento algébrico.

Buscou-se elencar os pressupostos teóricos dessas três vertentes, com o intuito de discutir relações entre construção de conhecimento, formulação de estratégias para resolver problemas, manifestações de abstração e protagonismo discente, com base nas características dos ambientes de aprendizagem mediados sob a perspectiva do construcionismo de Papert.

### **O construcionismo de Papert e sua origem piagetiana**

Na década de 1960, os primeiros estudos sobre o uso de uma linguagem de programação na aprendizagem de matemática evidenciaram a possibilidade de que a interação humano-computador poderia ser distinta do método instrucional. Embora tenha sido considerada como uma forma utópica de utilização de tecnologias digitais na sala de aula na época em que foi concebida, os resultados das pesquisas iniciais com ambientes de aprendizagem delineados sob a ótica construcionista foram promissores.

É importante destacar que a origem do construcionismo remonta uma mudança de paradigma educacional referente à forma de uso do computador como recurso de ensino. A distinção entre o instrucionismo e o construcionismo é dada pela forma como o computador é utilizado. Enquanto no instrucionismo o computador é uma máquina para ensinar, no construcionismo o computador é uma máquina a ser ensinada (VALENTE, 1993).

Em resumo, é o usuário que indica a ação a ser realizada pela máquina, e não o contrário. Dessa forma, a característica principal dos ambientes de aprendizagem

baseados no construcionismo é a possibilidade de desenvolvimento tanto cognitivo quanto emocional dos educandos por meio do uso da programação, com a construção de significados mediante a interação do educando com o computador (PAPERT, 1988), sendo sugerida como a forma ideal para uso do computador no processo educativo (VALENTE, 1997).

Papert (2008) descreve o construcionismo como sua reconstrução pessoal do construtivismo, com foco na análise de construções mentais. Dessa forma, pode-se afirmar que o foco construcionismo não é a análise dos estágios de desenvolvimento do educando, mas dos aspectos relacionados à cultura com e sem computadores (PAPERT, 1988). No entanto, é claro que similaridades entre construcionismo e construtivismo são evidentes, já que o construcionismo agrega aspectos cognitivistas e desenvolvimentalistas, uma vez que é considerada a interação indivíduo-ambiente para a construção de visão de mundo (CAMPOS, 2009).

Dentre os aspectos comuns dessas perspectivas, elenca-se o uso de um elemento que permita descrever ideias de forma clara e objetiva. Este é o objetivo da linguagem de programação, ao qual é atribuído o papel de elo de comunicação entre indivíduo e máquina, de forma que ações sejam descritas para que tanto máquina quanto usuário compreendam (PAPERT, 1998). Dessa forma, a linguagem de programação utilizada não deve ser um obstáculo para a aprendizagem. De forma metafórica, esses aspectos são descritos por Papert (1998) como similares aos processos primitivos presentes na aprendizagem da fala, descritos por Piaget no estágio pré-operatório de desenvolvimento humano.

Similaridades entre o construcionismo e o construtivismo também são observáveis no processo de elaboração de estratégias para solução de um problema, por contemplar pesquisa, descoberta e reflexão sobre as estratégias utilizadas. Papert (1988) descreve que este processo possibilita o ato de “pensar sobre modos de pensar”. Por essa razão, compara a criança a um epistemólogo, pelo fato de que o educando pode adotar o melhor estilo cognitivo para a solução de um problema. Desta forma, pode descobrir seu “estilo de pensamento”, aprendendo de forma articulada.

No entanto, os principais aspectos que evidenciam os elos entre construcionismo e a teoria de Piaget estão relacionados à reorganização mental, que são os processos de assimilação, de acomodação e de equilíbrio.

Os processos de assimilação e acomodação são responsáveis pelo crescimento cognitivo da criança (MOREIRA, 2015). A assimilação consiste na construção de esquemas assimilatórios pelo indivíduo, decorrente da interação indivíduo-objeto. Conforme Moreira (2015), neste processo a mente da criança não se modifica, ocorre apenas a incorporação da realidade por meio de assimilação. Já o processo de acomodação consiste na modificação dos esquemas de ação para assimilar uma situação, e é por meio da acomodação que ocorre o crescimento cognitivo discente. Dessa forma, não existe um processo de acomodação sem que ocorra o processo de assimilação.

Além disso, um processo de acomodação gera novos processos assimilatórios. Este contexto de adaptação a novas situações e equilíbrio deste processo é

chamado de *equilíbrio majorante*, que é um aspecto essencial para a aprendizagem. Assim, pode-se afirmar que a inteligência é uma função de adaptação, de forma que ocorre uma reestruturação cognitiva do indivíduo. Cabe ao professor criar situações de desequilíbrio, ou seja, que promovam os processos de assimilação e acomodação.

Em geral, problemas criados sob a ótica da prática construcionista são abertos, ou seja, não há indicação dos passos que devem ser seguidos pelo aprendiz. Dessa forma, o processo assimilatório decorre da necessidade de elaborar estratégias para solucionar o problema, ao começar pela análise e compreensão do contexto ao qual está relacionado.

Assim, o principal aspecto que difere o construcionismo do construtivismo é a possibilidade de externalizar processos mentais por meio de um produto “palpável”. Embora o programa gerado por meio da linguagem de programação seja um objeto abstrato, é possível verificar se a forma como as ideias foram estruturadas correspondem às intenções do indivíduo que as ordenou. Este processo contínuo de criação e verificação de resultados é denominado *ciclo construcionista*, cujas etapas são descritas a seguir (VALENTE, 2005):

- a) **Descrição:** descrição da ação que deve ser executada pelo computador, que consiste no traçado de estratégias para solucionar um problema;
- b) **Execução:** execução (pelo computador) do que foi descrito, concomitantemente com a análise do resultado obtido;
- c) **Reflexão:** o aprendiz reflete sobre suas ações, adaptando-as, aprimorando-as ou modificando-as, se necessário. É aqui que reside a abstração reflexiva, necessária para a construção de conhecimento;
- d) **Depuração:** caso o processo anterior contenha algum erro, o aprendiz deve buscar alternativas para corrigi-lo, depurando-os.

Neste processo, diferentes estágios de construção de conhecimento são manifestados, como a abstração empírica, a abstração reflexionante e a pseudo-empírica, conforme ilustrado na Figura 1.

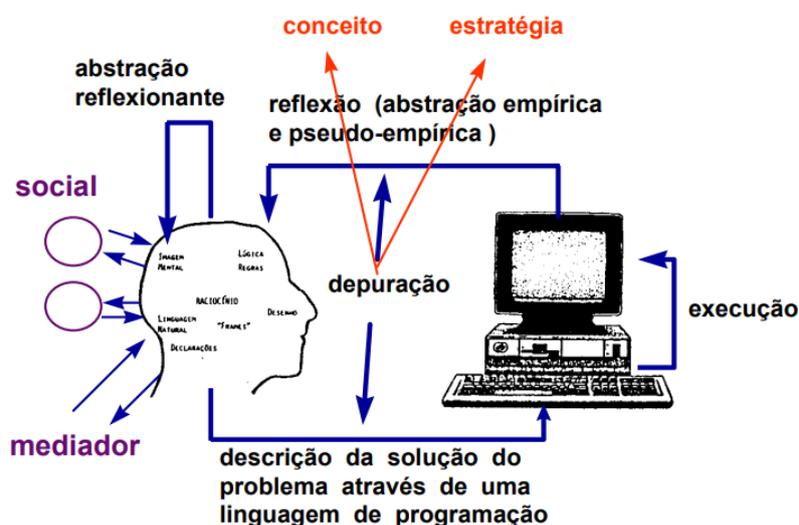


Figura 1 – Ciclo de ações na interação aluno-computador

Fonte: Valente (1999)

Note-se que, além dos aspectos anteriormente citados, o aprendiz pode utilizar-se de outras mediações provenientes da interação com indivíduos de seu meio social, como seus colegas de classe e demais pessoas de seu convívio. (VALENTE, 2002).

Um elemento fundamental no ciclo descrição-execução-reflexão-depuração é o erro (*bug*). É comum que no trabalho com programação não se obtenha êxito nas primeiras tentativas, de forma que a simulação do programa construído nem sempre corresponderá ao intuito delineado pelo estudante. Com isso, torna-se necessário revisar o código ou a estratégia utilizada, com o objetivo de corrigir os eventuais erros ou otimizar o processo. Neste processo de *debugging*, o aprendiz é encorajado a analisar e corrigir o erro, ao invés de simplesmente descartá-lo.

Dessa forma, o desequilíbrio, necessário para que ocorra a aprendizagem, está presente em ambientes de aprendizagem construcionistas. Além disso, estratégias podem ser revistas e otimizadas. Este momento caracteriza-se como um desequilíbrio na estrutura cognitiva do aprendiz, sendo um elemento necessário para que ocorra o ciclo assimilação-adaptação-acomodação.

Este processo pode se repetir diversas vezes, em um processo contínuo, denominado espiral da aprendizagem, conforme ilustrado na Figura 2.

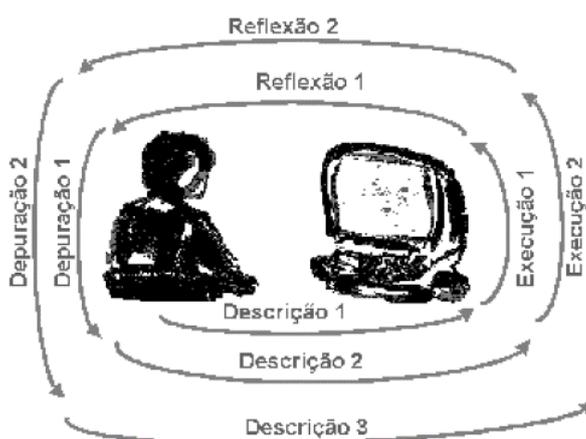


Figura 2 – Espiral da aprendizagem na perspectiva construcionista

Fonte: Valente (2002)

Em cada ciclo, o nível cognitivo do aluno é superior ao nível do ciclo imediatamente anterior, sendo agregados elementos que tornam cada ciclo qualitativamente diferente dos anteriores. Dessa forma, o crescimento da espiral depende da ocorrência do ciclo e as ações de ambos os processos são interdependentes, ou seja,

tanto as ações cíclicas quanto a espiral de aprendizagem estão acontecendo também simultaneamente, uma alimentando a outra. Nesse sentido, a espiral não cresce se o ciclo não acontece. E o ciclo acontece quando o aprendiz está realizando um produto concreto usando as facilidades da informática, segundo a proposta do construcionismo (VALENTE, 2005, p. 72).

Nesse contexto de aprendizagem, tanto o professor quanto o computador tem papel fundamental. O computador tem sua importância devido ao fato de não possuir

capacidade de reproduzir algo que não foi previamente indicado, servindo como âncora para o surgimento dos conflitos cognitivos na estrutura mental do aluno. Neste contexto é essencial o papel do mediador (o professor), sendo imprescindível o acompanhamento do processo pedagógico e psicológico que ocorre, pois conforme Valente (2002), sem a presença do professor o aluno teria que recriar conceitos historicamente convencionados.

Ao professor cabe propiciar meios para que o ciclo descrição-execução-reflexão-depuração seja contínuo, sem rupturas, pois é possível que o aluno não disponha de conhecimento suficiente para continuar a resolução de um problema. (VALENTE, 2002), ou seja, o docente deve desenvolver estratégias para despertar o processo de desequilíbrio descrito por Piaget, mediando o processo de assimilação e acomodação.

Assim, além de possibilitar o desenvolvimento discente, a prática construcionista permite uma melhor compreensão sobre o trabalho docente, pois o professor tem a oportunidade de entender o processo de construção de conhecimento do aluno de forma integral (CAMPOS, 2009). No tocante ao ensino da matemática, a análise do processo de construção de um programa de computador pode propiciar melhor compreensão sobre o pensamento discente acerca de estruturas e estratégias, além de possibilitar a verificação de como o aprendiz se utiliza de diferentes simbologias para se comunicar, como por exemplo, diferentes formas de manifestação algébrica.

## **O pensamento computacional como estratégia para resolução de problemas**

O termo “pensamento computacional”, introduzido por Wing (2006) no cenário educacional, remete-se ao uso de estratégias da ciência da computação para resolver problemas. Trata-se de uma visão sistêmica de resolução de problemas, comparável ao que se descreve na literatura como pensamento complexo, relacionado a abordagens de aprendizagem profunda (JOHNSON *et al*, 2015).

O “pensar computacionalmente” é sugerido como uma habilidade universal, necessária ao contexto de diversas profissões, não apenas às ligadas com a ciência da computação. Não se trata de automatizar processos, mas em pensar de forma recursiva, usando abstração e decomposição para tornar problemas complexos em problemas tratáveis. (WING, 2006).

Dentre as características do pensamento computacional, Wing (2006, 2008) cita que:

- a) Pensar computacionalmente não é, necessariamente, o ato de programar. É uma tarefa que, acima de tudo, consiste em abstração;
- b) Não é uma tarefa mecânica, mas uma habilidade fundamental para qualquer pessoa na atualidade, cujo intuito não é fazer humanos pensar como computadores, pois contempla a forma como o ser humano pensa e resolve problemas;
- c) Integra conhecimentos tanto do pensamento matemático quanto da engenharia para a descrição de sistemas;
- d) É um conjunto de conceitos utilizados para a solução de problemas, e não apenas para gerar artefatos.

Apesar de ser um termo recorrente em diversas pesquisas educacionais nos últimos anos, não há na literatura uma definição precisa e universal sobre o significado de pensamento computacional (VALENTE, 2016). Entretanto, uma frase breve que pode delimitar de forma clara os limites desse modelo de pensamento é dado por Wing (2014, p.1), ao afirmar que “o pensamento computacional descreve a atividade mental na formulação de um problema para admitir uma solução computacional”.

Embora pensar computacionalmente não implique o ato de programar, atividades relacionadas ao uso de programação indicam ser um caminho viável para o desenvolvimento do pensamento computacional discente. Esta possibilidade evidencia, mais uma vez, o potencial de atividades envolvendo programação para o desenvolvimento de habilidades cognitivas, de forma que é possível verificar similaridades do pensamento computacional com o propósito do construcionismo.

Na caracterização de pensamento computacional de Wing (2006, 2008, 2014), a abstração, o protagonismo discente e a resolução sistêmica de problemas estão presentes. Devido a isso, cita-se que “o empoderamento digital é a intersecção entre o Construcionismo e o Pensamento Computacional, permitindo identificar nos erros do passado, estratégias para a popularização da tecnologia na Educação Básica.” (VIEIRA; SANTANA; RAABE, 2017, p. 96).

A síntese de processos referentes ao “pensar computacionalmente” de Brackmann (2017) possibilita uma melhor compreensão do que consiste o tratamento de um problema nessa perspectiva:

- a) Decomposição: refere-se a dividir um problema em unidades menores, com soluções mais simples;
- b) Reconhecimento de padrões: consiste na análise das unidades de decomposição, na busca por interpretações particulares;
- c) Abstração: foco apenas nos dados relevantes do problema, no sentido de abstrair a essência do problema;
- d) Algoritmos: descrição de regras ou passos para tentar resolver o problema.

Não há consenso sobre como desenvolver o pensamento computacional de forma plena (BRACKMANN, 2017). No entanto, recomenda-se ênfase no protagonismo discente, para que não crie uma “computofobia”, decorrente da aversão às atividades envolvendo programação (VIEIRA, SANTANA, RAABE, 2017).

No que diz respeito à disciplina de matemática, o caráter abstrato de determinadas temáticas favorece a exploração de elementos do pensamento computacional. Conforme Valente et al. (2017), um modelo computacional é equivalente a uma abstração lógico-matemática, de forma que para desenvolver um modelo computacional é necessário fazer uso do pensamento matemático. Devido a isso, os autores questionam sobre a ordem de dependência destes dois estilos de pensamento, ou seja, qual precede qual.

É compreensível este questionamento ao analisar o propósito do construcionismo nas primeiras experiências desenvolvidas por Papert e sua equipe, que foi desenvolver habilidades em matemática. Conforme citado na seção anterior, em atividades relacionadas ao construcionismo o aprendiz instrui a máquina, ou seja, antes de indicar o

que o computador deve fazer, o aprendiz deve compreender o contexto do problema a ser resolvido, elaborar estratégias e elencar elementos suficientes para auxiliar na solução e depois combiná-los usando uma linguagem de programação.

Com isso, a linguagem de programação é o elo entre aprendiz e máquina, mas o “pensar matematicamente” e o “pensar computacionalmente” são mesclados neste processo, de forma contínua. O pensamento matemático parece preceder o “pensar computacionalmente”, mas ambos se complementam. Isso é reforçado por Valente et al. (2017), ao sugerir que há uma intersecção entre aspectos do pensamento computacional e do pensamento algébrico, existindo uma conexão natural entre a matemática e o pensamento computacional.

### **Pensamento algébrico e a construção de símbolo**

Embora não exista uma definição consensual da álgebra, ela pode ser compreendida como uma generalização da aritmética (CARRAHER; BRIZUELA; SCHLIEMANN, 2000). Constitui-se como um dos elementos estruturantes do currículo da disciplina de matemática, de forma que seu estudo não consista apenas o uso de simbologia, mas o desenvolvimento de uma habilidade específica: o pensamento algébrico.

Internacionalmente, recomenda-se o desenvolvimento do pensamento algébrico desde as séries iniciais do Ensino Fundamental. No entanto, no Brasil a ênfase a esse processo nos anos iniciais foi incluída nos documentos curriculares apenas na última década (FERREIRA, 2017; RODRIGUES, PIRES, 2017).

Na literatura, há diversas definições para o pensamento algébrico. Para Fiorentini, Miorim e Miguel (1993), o pensamento algébrico é um tipo especial de pensamento, manifestado em diferentes campos da matemática e de outras áreas do conhecimento. Por essa razão, os autores afirmam que

A análise das situações em que esse pensamento pode se manifestar levou-nos, ainda, a concluir que não existe uma única forma de se expressar o pensamento algébrico. Ele pode expressar-se através da linguagem natural, através da linguagem aritmética, através da linguagem geométrica ou através da criação de uma linguagem específica para esse fim, isto é, através de uma linguagem algébrica, de natureza estritamente simbólica. (FIORENTINI, MIORIM, MIGUEL, 1993, p. 88).

Esse ponto de vista é promissor por se perspectivar a exploração de maneiras alternativas de se desenvolver o pensamento algébrico. Dessa forma, estratégias com esse propósito podem ser desenvolvidas desde a educação infantil, caracterizando a educação pré-algébrica.

Outro aspecto importante sobre o pensamento algébrico destacada por Fiorentini, Miorim e Miguel (1993) é a relação entre pensamento e linguagem simbólica. Segundo os autores, essa relação é análoga à relação entre pensamento e linguagem natural, que influencia o desenvolvimento psico-cognitivo, de forma que a linguagem simbólica cumpre o papel de constituição do pensamento abstrato.

Dessa forma, embora existam diversas concepções sobre a constituição do que é o pensamento algébrico, pode-se resumir que este modelo de pensamento diz respeito à utilização de linguagem algébrica para solucionar e descrever problemas, modelando-os e fazendo generalizações (FIORENTINI, MIORIM, MIGUEL, 1993; KIERAN, 2007; CAI; MOYER, 2008; GROENWALD; BECHER, 2010; FERNANDES; SAVIOLI, 2016, FERREIRA, 2017).

Essa concepção contempla as componentes relacionadas à construção do sentido de símbolo de Arcavi (2007), elencadas a seguir:

- a) Simpatia com os símbolos: inclui a compreensão dos símbolos e sobre como e quando podem ser utilizados como forma de generalização, que com uso de outras notações ficariam ocultas;
- b) Capacidade para manipular e interpretar expressões algébricas, como aspectos complementares para a resolução de problemas: trata-se de adotar uma visão global da simbologia, contemplando a atribuição de significados e a razoabilidade na busca por soluções;
- c) Consciência sobre a possibilidade para utilizar relações simbólicas que expressem determinadas informações, como o uso de gráficos, por exemplo;
- d) Capacidade de escolher uma notação simbólica adequada, modificando-a para outra mais adequada, se necessário;
- e) Consciência da necessidade de revisão da simbologia algébrica utilizada em determinadas situações;
- f) Compreender que os símbolos podem desempenhar diferentes sentidos, como variáveis, parâmetros e incógnitas.

No entanto, tal como o pensamento computacional, não há unanimidade sobre a melhor maneira para desenvolver o pensamento algébrico de forma plena e é comum que os alunos das séries finais do ensino fundamental e alunos do ensino médio apresentem dificuldade na compreensão e utilização da álgebra para a solução de problemas, principalmente na manipulação de expressões simbólicas (BONADIMAN, 2012; SUSAC et al., 2014; PEDERSEN, 2015).

Segundo Fiorentini, Miorim e Miguel (1993), a primeira etapa da educação algébrica deve ser embasada no trabalho com situações-problema, para viabilizar a construção significativa da utilização da linguagem simbólica. Já a segunda etapa deve consistir na atribuição de significação à estrutura das expressões algébricas, ou seja, o caminho inverso do que é recomendado na primeira etapa.

Por fim, a terceira etapa consiste no transformismo algébrico, ou seja, na forma como uma expressão algébrica pode ser transformada em outra equivalente, porém mais simples. Embora sejam descritas em três momentos, os autores reforçam que a ordem de trabalho não é fixa, de forma que o trabalho interetapas é relevante para a construção do pensamento algébrico de forma sólida.

Deve-se considerar, também, que o pensamento algébrico não é uma habilidade que se desenvolve de forma imediata. Este processo é gradativo e dividido em etapas.

Godino et al. (2014), com base na teoria antropológica do didático, indica essa divisão no Ensino Fundamental e no Ensino Médio em níveis, conforme descrito a seguir:

- a) Nível 0: uso de linguagens alternativas para representar generalidades (natural, numérica, gestual ou icônica);
- b) Nível 1: uso de simbologia, propriedades do conjunto dos números naturais e uso de equivalências;
- c) Nível 2: uso de representação simbólica, com descrição de problemas no formato  $ax+b=c$ ;
- d) Nível 3: uso de símbolos de forma analítica, sem necessidade de um contexto específico. É nesta fase que o aluno resolve equações de primeiro grau do tipo  $ax+b=cx+d$ , operando com variáveis e incógnitas;
- e) Nível 4: estudo de famílias de equações usando parâmetros e coeficientes;
- f) Nível 5: uso de um ou mais parâmetros, variáveis e incógnitas, no formato analítico;
- g) Nível 6: estudo de estruturas algébricas e suas propriedades estruturais.

Os quatro primeiros níveis são relacionados ao Ensino Fundamental e os dois últimos estão relacionados à estratégias próprias do Ensino Médio, etapa da educação básica onde o aluno tem contato com diferentes estruturas algébricas, como matrizes, por exemplo. No entanto, conforme levantamento efetuado por Rodrigues e Pires (2017), no contexto educacional brasileiro o pensamento algébrico não foi enfatizado de forma satisfatória nos últimos anos. Resultados indicam fragilidade no desempenho discente e existência de poucas alternativas que auxiliem a mudar esse cenário.

Por se tratar de um processo contínuo, que inicia na educação básica, falhas em quaisquer etapas comprometem o sucesso nas etapas posteriores, principalmente no Ensino Médio. O insucesso poderia ser evitado, segundo Cai e Moyer (2008), se álgebra e aritmética não fossem trabalhadas de formas dissociadas.

Desta forma, torna-se necessária a concepção de ambientes de aprendizagem para reconstrução do uso da álgebra. Por essa razão, na seção a seguir são discutidas as similaridades entre a prática construcionista e o pensamento computacional, com o intuito de delinear aspectos que justifiquem a prática construcionista como base para o desenvolvimento de habilidades relacionadas ao pensamento algébrico.

### **App Inventor e desenvolvimento de aplicativos para execução de cálculos: possibilidades para a educação algébrica na educação básica**

Conforme elencado no referencial teórico, o processo de programar ações para execução do computador consiste em oportunidade para que o educando explore sua maneira de pensar. Devido a isso, a prática construcionista propicia um momento singular onde a ação executada pelo computador não consiste apenas na ideia de representação do conhecimento do aluno, mas ilustra a forma como o aluno raciocina, possibilitando que o professor possa analisar o processo de construção de significados e melhor compreendê-lo (MALTEMPI, 2005). Este contexto é totalmente distinto do contexto

presente nos estudos de Piaget, pois os objetos sob os quais ocorria a ação eram estáticos. (VALENTE, 2002).

Na atualidade, possibilidades para aliar o protagonismo discente via pensamento computacional às disciplinas do currículo da educação básica são citadas na Base Nacional Comum Curricular (BNCC). Em relação à disciplina de matemática nas séries finais do Ensino Fundamental, o pensamento computacional é citado como complemento ao desenvolvimento de habilidades relacionadas ao uso de simbologia e resolução de problemas. Visão semelhante é citada na versão preliminar para o Ensino Médio, ao se indicar que

A área de Matemática, no Ensino Fundamental, centra-se no desenvolvimento da compreensão de conceitos e procedimentos em seus diferentes campos, visando à resolução de situações-problema. No Ensino Médio, na área de Matemática e suas Tecnologias, os estudantes devem utilizar conceitos, procedimentos e estratégias não apenas para resolver problemas, mas também para formulá-los, descrever dados, selecionar modelos matemáticos e desenvolver o pensamento computacional, por meio da utilização de diferentes recursos da área. (BRASIL, 2018, p. 470).

A importância dada ao pensamento computacional na BNCC possibilita perspectivá-lo como abordagem não apenas para solução, mas descrição e exploração de problemas, no intuito de promover uma análise aprofundada das temáticas abordadas. Justifica-se essa possibilidade ao combinar a resolução de problemas com uma linguagem de programação que possibilite a descrição de um problema matemático sem que a manipulação dessa linguagem de programação seja um obstáculo de aprendizagem e, ao mesmo tempo, estimule a manifestação de elementos simbólicos.

Um exemplo de linguagem de programação que minimiza as dificuldades operacionais é a linguagem de programação visual. Há inúmeros *softwares* desenvolvidos com esse estilo de programação, cujo intuito é promover a estruturação de comando mediante a combinação de blocos lógicos. As ferramentas *Scratch* e *App Inventor*, ambas atualmente gerenciadas pelo Instituto de Tecnologia de Massachussets (MIT), são exemplos de plataformas de desenvolvimento baseadas em programação visual. Nas figuras 3 e 4 são ilustrados exemplos de comandos estruturados nessas duas plataformas.

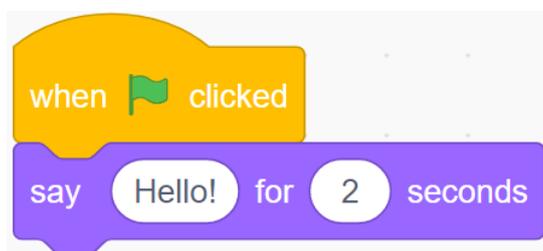


Figura 3 – Exemplo de comando estruturado na plataforma *Scratch*

Fonte: O autor

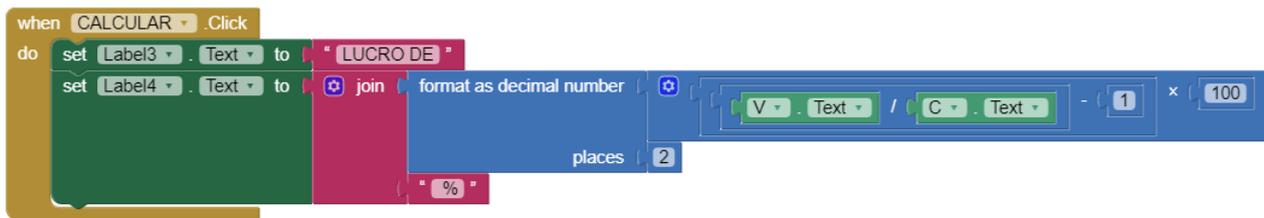


Figura 4 – Exemplo de comando estruturado na plataforma *App Inventor*

Fonte: O autor

Experimentos com base na linguagem de programação visual e robótica foram utilizados por Santos (2018) como forma de explicitação de invariantes operatórios utilizados para a estruturação de um algoritmo. A pesquisa foi realizada com acadêmicos de um curso de licenciatura em matemática e corroboram a possibilidade de uso da linguagem de programação como forma de manifestação de elementos utilizados na resolução de um problema e que, de certa forma, ficam ocultos ao resolvê-lo no modo convencional, usando lápis e papel.

De forma semelhante, o processo de estruturação de aplicativos para execução de cálculos parece favorecer a explicitação de conceitos relacionados à construção de símbolo. O exemplo a seguir apresentado consiste em um experimento realizado com alunos do Ensino Médio usando a ferramenta *App Inventor* para desenvolver aplicativos de execução de cálculos, onde até mesmo a exploração de operações matemáticas básicas favoreceu a manifestação de determinados elementos relacionados ao pensamento algébrico.

Na Figura 5 é apresentado um exemplo de um dos comandos estruturados por um aluno do Ensino Médio ao conceber um aplicativo com o qual é possível efetuar a soma, diferença, o produto e razão entre dois números reais.

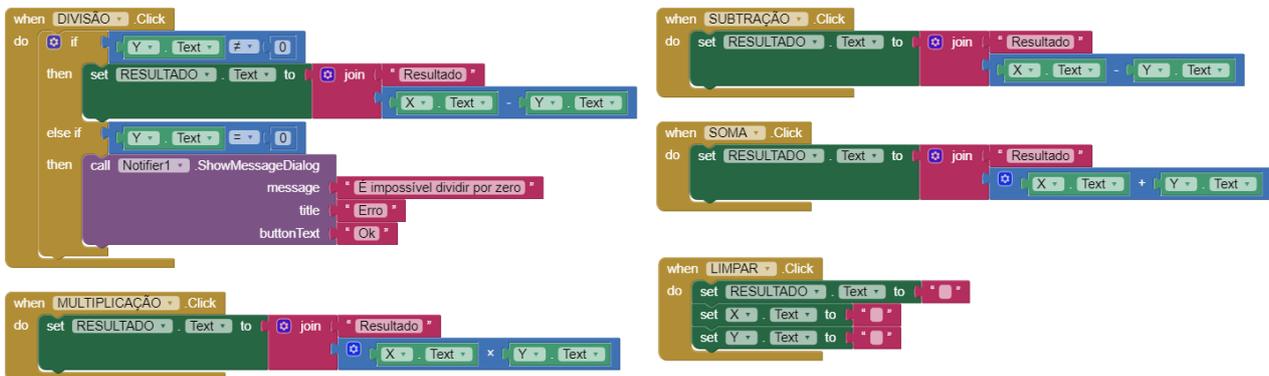


Figura 5 – Exemplo de códigos para operações com números reais

Fonte: O autor

A sequência estruturada pelo aluno ilustra a síntese de processos referente ao “pensar computacionalmente” de Brackmann (2017), conforme descrito a seguir:

- a) Decomposição: o problema foi dividido em partes, de forma que cada bloco se constitui em um comando ativado por um botão;
- b) Reconhecimento de padrões: Uma vez entendida a forma como efetuar um cálculo, o processo é repetido para as demais operações;

- c) Abstração: a essência do problema consiste em efetuar uma operação com base em dados que o usuário digita na tela do celular/*tablet* e disponibilizá-lo para o usuário em outro campo;
- d) Algoritmos: a estruturação dos códigos é organizada de forma objetiva, contemplando os dados necessários e possíveis modificações.

Ao realizar tal operação, contemplam-se algumas das componentes relacionadas à construção de símbolo de Arcavi (2007) demandas pelo problema, como a simpatia com os símbolos, a capacidade para manipular expressões algébricas e o uso de variáveis. Note-se, também, o uso de condicionais para limitar a divisão por zero, indicando a preocupação com um aspecto elementar da divisão, que poderia resultar em um erro ao tentar ser executado.

Embora as operações de soma, subtração, multiplicação e divisão sejam cálculos triviais para alunos das séries finais do Ensino Fundamental e no Ensino Médio, esse exemplo de atividade pode estimular o uso de representação simbólica, ao aliar pensamento computacional sob a perspectiva construcionista.

Dessa forma, a produção discente decorrente dessa atividade educativa se constitui em um importante instrumento de avaliação, onde desdobramentos podem indicar níveis diferentes de abstração e conhecimento sobre a temática abordada.

### **Considerações finais**

Conforme indicado no referencial teórico, há elementos comuns entre a prática construcionista e o pensar computacionalmente, bem como entre o pensamento computacional e o pensamento algébrico. Este cenário permite cogitar a existência de uma interseção entre esses três conjuntos, principalmente ao considerar o propósito inicial do construcionismo descrito por Papert (2008) como uma análise de construções mentais baseada no construtivismo.

Essa possibilidade permite perspectivar o delineamento de estratégias educativas que combinem tais aspectos, com destaque ao estímulo dos processos de assimilação e acomodação. Para isso, é importante contemplar o caráter exploratório, de forma que se dê liberdade ao aprendiz para criar soluções e, no que diz respeito à abordagens construcionista com o intuito de estimular o pensamento abstrato, criação de cenários de exploração conceitual de estruturas algébricas que regem determinados processos.

Cenários de aprendizagem que contemplam a estruturação de aplicativos para a execução de cálculos podem favorecer uma melhor compreensão sobre o modo que o aluno compreende e utiliza conceitos matemáticos em contextos distintos, quer sejam puramente algorítmicos ou relacionados à modelagem de problemas. Com isso, as linhas de código desenvolvidas pelos alunos se constituem em um importante instrumento diagnóstico de manifestações do pensamento algébrico. Além disso, viabilizam a avaliação em caráter qualitativo, no sentido de valorizar a pluralidade de formas de pensamento, em detrimento da resolução convencional de problemas envolvendo números.

Evidentemente, essa prática está relacionada também a aspectos emocionais relevantes no processo de aprendizagem, como desenvolvimento da autonomia, da criatividade e o fato do aprendiz encarar o erro como algo comum no processo de aprender, além de usá-lo como elemento mobilizador para revisão de estratégias para solucionar um problema (processo de *debugging*).

Para tal, é fundamental que atividades educativas que contemplem pensamento algébrico e pensamento computacional sob a ótica construcionista sejam delineadas de forma aberta e, fundamentalmente, contemplem aspectos importantes das temáticas abordadas em sala de aula, de forma que estimulem a reflexão e a (re) construção de significados.

Outro aspecto relevante que deve ser considerado é o ponderamento sobre os limites e potencialidades da ferramenta computacional utilizada, de forma que a linguagem de programação não seja um obstáculo de aprendizagem. Dessa forma, o foco de atividades com tal cunho prevalecerá, efetivamente, na construção e exploração de conceitos, e não na mera reprodução de códigos que não são efetivamente significativos para o aprendiz.

## Referências

ARCAVI, A. El desarrollo y el uso del sentido de los símbolos. Uno: **Revista de didáctica de las matemáticas**. Rioja, n. 44, p. 59-75, 2007. Disponível em: <<https://dialnet.unirioja.es/servlet/articulo?codigo=2240831>>. Acesso em: 10 jan. 2018.

BONADIMAN, A. Álgebra no Ensino Fundamental: produzindo significados para as operações básicas com expressões algébricas. In: BÚRIGO, E. Z. et al. (Orgs). **Matemática na escola: novos conteúdos, novas abordagens**. Porto Alegre: Editora da UFRGS, 2012. p. 99-118.

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: Ensino Fundamental**. Brasília, 2017. Disponível em: [www.basenacionalcomum.mec.gov.br/images/BNCC\\_20dez\\_site.pdf](http://www.basenacionalcomum.mec.gov.br/images/BNCC_20dez_site.pdf). Acesso em: 10 jan. 2018.

BRASIL. Ministério da Educação. **Base Nacional Comum Curricular: Ensino Médio**. Brasília, 2018. Disponível em: <<http://basenacionalcomum.mec.gov.br/bncc-ensino-medio>>. Acesso em: 18 jun. 2018.

CAI, J.; MOYER, J.C. Developing Algebraic Thinking in Earlier Grades: Some Insights from International Comparative Studies. In: GREENES, C.E.; RUBENSTEIN, R. (Eds.). **Algebra and algebraic thinking in school mathematics**. Reston: NCTM, 2008. p.169-180.

CARRAHER, D.; BRIZUELA, B.M.; SCHLIEMANN, A. D. Bringing Out the Algebraic Character of Arithmetic: Instantiating Variables in Addition and Subtraction. In:

CONFERENCE OF THE INTERNATIONAL GROUP FOR THE PSYCHOLOGY OF MATHEMATICS EDUCATION (PME), 24, 2000. Hiroshima. **Anais...** Hiroshima: PME, 2000, p. 145-152.

CAMPOS, F.R. **Diálogo entre Paulo Freire e Seymour Papert: a prática educativa e as tecnologias digitais de informação e comunicação.** 2009. 183 f. Tese (Doutorado em Letras) - Universidade Presbiteriana Mackenzie, São Paulo, 2009.

CHAO, P. Exploring students' computational practice, design and performance of problem-solving through a visual programming environment. **Computers&Education**, v. 95, p. 202-215, 2016.

FERNANDES R. K.; SAVIOLI, A. M. P. D. Características de pensamento algébrico manifestadas por estudantes do 5º ano do ensino fundamental. **Revista Paranaense de Educação Matemática - RPEM**, Campo Mourão, PR, v.5, n.8, p.131-151, jan./jun. 2016. Disponível em: <[www.fecilcam.br/revista/index.php/rpem/article/view/1228](http://www.fecilcam.br/revista/index.php/rpem/article/view/1228)>. Acesso em: 18 jan. 2018.

FERREIRA, M.C.N. Álgebra nos anos iniciais do ensino fundamental: uma análise dos documentos curriculares nacionais. **REnCiMa**, v. 8, n. 5, p.16-34, 2017. Disponível em: <https://doi.org/10.26843/rencima.v8i5.1247>. Acesso em: 06 jun. 2019.

FESSAKIS, G.; GOULI. E.; MAVROUDI, E. Problem solving by 5–6 years old kindergarten children in a computer programming environment: A case study. **Computers&Education**, v. 63, p. 87-97, 2013.

FIORENTINI, D; MIORIN, M. A.; MIGUEL, A. *Contribuição para um Repensar a Educação Algébrica Elementar.* **Pro-Posições**, Campinas, v. 4, n. 1, p. 78 - 91, mar. 1993.

GODINO, J. D. *et. al.* Niveles de algebrización de la actividad matemática escolar. Implicaciones para la formación de maestros. **Enseñanza de las Ciencias**, v. 32, n. 1, p. 199-219, 2014.

GROENWALD, C. L. O.; BECHER, E. L. Características do pensamento algébrico de estudantes do Ensino Médio com equações do 1º grau. **Acta Scientiae**, Canoas, v. 12, n.1, p. 83-94, 2010.

ISRAEL, M.; et al. Supporting all learners in school-wide computational thinking: A cross-case qualitative analysis. **Computers&Education**, v. 82, p. 263-279, 2015.

JOHNSON, L., et al. **NMC HorizonReport: Edição Educação Básica 2015.** Austin, Texas: The New Media Consortium, 2015.

KIERAN, C. Learning and teaching algebra at the middle school through college levels. In F. K. Lester (Ed.). **Second handbook of research on mathematics teaching and learning.** Charlotte, NC: Information Age. 2007, p. 707-762.

LYE, S. Y; KOH, J. H. L. Review on teaching and learning of computational thinking through programming: What is next for K-12? **Computers in Human Behavior**, v. 41, p. 51-61, 2014.

MALTEMPI, M.V. Construcionismo: pano de fundo para pesquisas em informática aplicada à Educação Matemática. In: BICUDO, M.A.V.; BORBA, M. de C. (Orgs).

**Educação Matemática:** pesquisa em movimento. 2. ed. Revisada. São Paulo: Cortez, 2005, p. 264-282.

MOREIRA, M. A. **Teorias de aprendizagem**. 2. ed. ampl. São Paulo: EPU, 2015.

PAPERT, S. **Logo:** Computadores e Educação. Tradução de José Armando Valente, Beatriz Bitelman e Afira Vianna Ripper. 3. ed. São Paulo: Brasiliense, 1988.

PAPERT, S. A máquina das crianças: repensando a escola na era da informática. Tradução de Sandra Costa. Porto Alegre: Artmed, 2008.

PEDERSEN, I. F. What characterizes the algebraic competence of norwegian upper secondary school students? Evidence from TIMSS advanced. **International Journal of Science and Mathematics Education**, v. 13, n. 1, p. S71-S96, 2015.

RODRIGUES, I. C., PIRES, C. M. C. Um mapeamento de teses e dissertações que abordam o ensino e a aprendizagem da álgebra no ensino fundamental no Brasil. **REnCiMa**, v. 8, n. 2, p.162-182, 2017. Disponível em: <https://doi.org/10.26843/rencima.v8i2.1181>. Acesso em: 06 jun. 2019.

SÁEZ-LÓPEZ, J. M.; ROMÁN-GONZALEZ, M.; VÁZQUEZ-CANO, E. Visual programming languages integrated across the curriculum in elementary school: A two year case study using “Scratch” in five schools. **Computers & Education**, v. 97, p. 129-141, 2016.

SANTOS, C. F. R. dos. **A robótica educacional como recurso de mobilização e explicitação de invariantes operatórios na resolução de problemas**. 2018. 189 f. Tese (Doutorado em Ensino de Ciência e Tecnologia) – Programa de Pós-Graduação em Ensino de Ciência e Tecnologia. Universidade Tecnológica Federal do Paraná, Ponta Grossa, 2018.

SUSAC, A. et al. Development of abstract mathematical reasoning: the case of algebra. **Frontiers in Human Neuroscience**, v. 8, set. 2014.

VALENTE, J.A. Diferentes usos do computador na educação. In: **Computadores e conhecimento:** repensando a educação. 1. ed. Campinas, NIED. Unicamp, 1993.

VALENTE, J. A. O uso inteligente do computador na educação. **Pátio Revista Pedagógica**. Editora: Artes Médicas Sul, ano 1, v. 1, p. 19-21, 1997.

VALENTE, J. A. *Análise dos diferentes tipos de softwares usados na educação*. In: \_\_\_\_\_. (Org.). **O computador na sociedade do conhecimento**. Campinas, SP: Gráfica da UNICAMP, 1999, cap. 4. p. 187-208.

VALENTE, J.A. A espiral da aprendizagem e as tecnologias da informação e comunicação: repensando conceitos. In: JOLY, M.C.R.A. (Org.) **A tecnologia no ensino:** implicações para a aprendizagem. São Paulo: Casa do Psicólogo, 2002, cap. 1, p. 16-37.

VALENTE, J.A. **A espiral da espiral da aprendizagem: o processo de compreensão do papel das tecnologias de informação e comunicação na educação**. 2005, 232 f. Tese (Livre Docência). Universidade Estadual de Campinas, Campinas, 2005.

VALENTE, J. A. Integração do Pensamento Computacional no currículo da Educação Básica: diferentes estratégias usadas e questões de formação de professores e avaliação do aluno. **Revista e-Curriculum**, v. 14, n. 3, 2016.

VALENTE, J. A. et al. Alan Turing tinha Pensamento Computacional? Reflexões sobre um campo em construção. **Tecnologias Sociedade e Conhecimento**, Campinas, v. 4, dez. 2017. Disponível em: <<http://www.nied.unicamp.br/ojs/index.php/tsc/article/view/185/191>>. Acesso em: 18 jan. 2018.

VIEIRA, M. F. V.; SANTANA, A.L.M.; RAABE, A.L.A. Do Logo ao Pensamento Computacional: o que se pode aprender com os resultados do uso da linguagem Logo nas escolas brasileiras. **Tecnologias, Sociedade e Conhecimento**, Campinas, v. 4, n. 1, p. 82-106, dez. 2017. Disponível em: <[www.nied.unicamp.br/ojs/index.php/tsc/article/view/181/212](http://www.nied.unicamp.br/ojs/index.php/tsc/article/view/181/212)>. Acesso em: 12 jan. 2018.

WILKERSON-JERDE, M. H. Construction, categorization, and consensus: student generated computational artifacts as a context for disciplinary reflection. **Educational Technology Research and Development**, v. 62, n. 1. p. 99-121, fev. 2014.

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

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

WING, J. M. Computational Thinking Benefits Society. **Social Issues in Computing**, 2014. Disponível em: <<http://socialissues.cs.toronto.edu/2014/01/computational-thinking/>>. Acesso em: 25 out. 2018.