


ARMADILHAS DO PENSAMENTO COMPUTACIONAL: ANÁLISE CRÍTICA

 <https://doi.org/10.56238/arev7n1-121>

Data de submissão: 13/12/2024

Data de publicação: 13/01/2025

Marcelo Magalhães Foohs

Doutor em Informática na Educação, Universidade Federal do Rio Grande do Sul

Antinesca Joana Pissolatto Neta

Mestre em Arquitetura e Urbanismo, Universidade Federal do Rio Grande do Sul

Júlia Andressa Paes dos Santos Krüger

Mestre em Microbiologia Agrícola e do Ambiente, Universidade Federal do Rio Grande do Sul

Randerson Oliveira Melville Rebouças

Mestre em Informática em Saúde, Universidade Federal do Rio Grande do Sul.

RESUMO

Este artigo analisa criticamente a evolução do conceito de pensamento computacional (PC) nos trabalhos de Jeanette Wing, com foco nos artigos publicados entre 2006 e 2017. Argumenta-se que, embora Wing tenha promovido amplamente o PC como uma habilidade essencial para diversas áreas do conhecimento, sua definição apresenta ambiguidades conceituais que comprometem sua delimitação e aplicabilidade. Um dos principais problemas identificados é o uso do operador lógico "OR" na definição inicial do PC, o que permite interpretações excessivamente amplas e a inclusão de processos exclusivamente humanos, não passíveis de execução por máquinas. A revisão dos textos posteriores demonstra que essa indefinição não foi corrigida, resultando em dificuldades na consolidação do PC como um campo formalmente estruturado. Além disso, discute-se a centralidade dos algoritmos, elemento essencial para a identidade do PC, e como sua ausência explícita na formulação de Wing contribuiu para a diluição do conceito. Também são examinadas as implicações educacionais da falta de uma definição precisa, que impactam diretamente o ensino e a implementação curricular do PC. Por fim, argumenta-se que, para garantir a coerência e a eficácia do pensamento computacional, é necessário um maior rigor conceitual, distinguindo claramente seu escopo e sua relação com outras formas de cognição. O artigo propõe uma delimitação mais precisa do PC, enfatizando sua base algorítmica e diferenciando-o de abordagens probabilísticas e modelos baseados em otimização aproximada, a fim de fortalecer sua identidade e aplicabilidade tanto na educação quanto na pesquisa científica.

Palavras-chave: Pensamento Computacional. Jeanette Wing. Algoritmos. Informática na Educação. Análise Crítica.

1 INTRODUÇÃO

Desde a publicação do artigo "*Computational Thinking*" por Jeanette Wing em 2006, o conceito de pensamento computacional (PC) tem ganhado destaque crescente nas discussões sobre educação e ciência da computação. Wing (2006) argumenta que o pensamento computacional não se limita à programação, mas é uma forma de resolver problemas que deveria ser ensinada a todos, desde cedo. Seu trabalho teve um impacto profundo na maneira como o PC é entendido e ensinado ao redor do mundo, influenciando tanto a formulação de currículos quanto as práticas pedagógicas. Contudo, algumas escolhas retóricas e conceituais feitas pela autora, especialmente o uso do conector lógico "OR" em sua definição, introduzem ambiguidades que geram entendimentos problemáticos.

Ao afirmar que "*Computational thinking builds on the power and limits of computing processes, whether they are executed by a human or by a machine*"¹ (Wing, 2006), a autora expande de forma imprecisa o escopo do conceito. O uso do conector "OR" como operador inclusivo implica que qualquer processo dito computacional, seja ele passível de execução somente por humanos, somente por máquinas ou por ambos, pode ser enquadrado como pensamento computacional. Isso leva a uma falsa simetria entre processos humanos e computacionais, ignorando a especificidade de algoritmos e estratégias sistematizáveis que são centrais para o pensamento computacional.

Embora a autora tenha continuado a explorar o pensamento computacional nos anos seguintes, em nenhum momento houve uma revisão explícita dessa definição que delimitasse de forma rigorosa o escopo do conceito. Assim, a indefinição conceitual persiste e compromete a clareza necessária para sua aplicação educacional e científica, permitindo interpretações amplas que podem diluir a especificidade do pensamento computacional.

Essa imprecisão, portanto, é persistente na formulação de Wing e tem implicações diretas para práticas educacionais. Um exemplo prático pode ser observado em atividades educacionais que abordam tarefas triviais do dia a dia como se fossem suficientes para que os alunos desenvolvam habilidades associadas ao pensamento computacional. Nessas atividades, processos exclusivamente humanos, como fluxos de consciência ou resolução intuitiva de problemas, acabam sendo interpretados como parte do escopo do PC. No entanto, tais processos frequentemente carecem da clareza e sistematicidade esperadas no contexto do pensamento computacional, desviando-se de sua essência algorítmica.

Apesar de que estratégias humanas subjetivas de resolução de problemas, como influências sociais ou práticas de meditação, possam ser analisadas ou descritas de maneira aproximada por meio

¹ O pensamento computacional baseia-se no poder e nos limites dos processos de computação, sejam eles executados por um humano OU por uma máquina (Wing, 2006, p. 1).

da linguagem natural, sua complexidade contextual e dependência de nuances subjetivas as afastam do escopo do pensamento computacional. Esse escopo é reservado para processos claros, sistemáticos e potencialmente executáveis por máquinas. Estratégias humanas baseadas em subjetividade não podem ser completamente traduzidas em algoritmos devido à ausência de lógica formal e sistematicidade, características intrínsecas ao pensamento computacional. Assim, essas estratégias permanecem fora do domínio de processos que podem ser representados e operacionalizados de forma precisa.

A ampliação conceitual compromete a formação de habilidades específicas que são centrais ao pensamento computacional, como a abstração e a criatividade voltadas à construção de algoritmos. Embora abstração e criatividade sejam capacidades humanas gerais, presentes em uma ampla variedade de contextos, no pensamento computacional essas capacidades precisam ser aplicadas de maneira direcionada à formulação de sequências claras e executáveis. A diluição do conceito ao incluir processos humanos gerais enfraquece essa especificidade, tornando o ensino e a prática do pensamento computacional menos eficazes. As dificuldades relatadas em relação a ingressantes no ensino superior, como a falta de capacidade de abstração e criatividade nesse contexto (cf. Oliveira e Pereira, 2019), reforçam a importância de um treinamento específico que alinhe essas capacidades às demandas do pensamento computacional, com foco na criação de soluções sistemáticas e algorítmicas.

Para diferenciar algoritmos tradicionais de modelos estatísticos usados em inteligência artificial, o Centro de Inovação para a Educação Brasileira (CIEB) propõe uma definição de algoritmos estruturados, destacando sua clareza e previsibilidade:

A matemática fornece uma linguagem formal e universal. É por meio dela que a Computação constrói modelos chamados de algoritmos. Embora possam ser descritos em diferentes níveis de abstração, com linguagens específicas que focam apenas nos elementos essenciais e relevantes, os algoritmos são sequências finitas de passos que levam a um determinado resultado. Pode haver complicações, como sequências condicionais e laços de repetição, mas todos os passos são definidos e podem ser compreendidos por uma máquina (grifo nosso) (CIEB, Nota Técnica #21, p. 9, 2024).

Em seguida, a Nota Técnica diferencia os algoritmos estruturados dos modelos estatísticos utilizados na inteligência artificial, que operam com base em aprendizado de máquina e adaptação dinâmica a novos dados. Essa distinção é fundamental, pois o pensamento computacional, conforme discutido neste artigo, fundamenta-se na construção de algoritmos estruturados e determinísticos. Já os modelos probabilísticos utilizados na IA seguem uma lógica distinta, ajustando-se dinamicamente a partir de padrões extraídos de dados.

Apesar de que alguns documentos e propostas utilizem o termo "algoritmo" para se referir a esses modelos, essa nomenclatura amplia sem delimitação adequada o escopo do termo e pode gerar confusão conceitual, pois os modelos estatísticos não seguem a estrutura determinística, explícita e sistemática dos algoritmos tradicionais. No contexto da inteligência artificial, algoritmos tradicionais e modelos estatísticos coexistem e se complementam, mas desempenham papéis distintos dentro dos sistemas computacionais.

Tentar enquadrar realidades distintas sob a mesma terminologia de algoritmo, mesmo com uma adjetivação diferenciada, pode comprometer a precisão conceitual do pensamento computacional e dificultar sua aplicação educacional. Essa distinção, no entanto, não reduz a importância dos modelos estatísticos, mas fortalece a clareza conceitual do pensamento computacional, permitindo que ambos os campos se desenvolvam de maneira complementar sem diluir suas identidades.

É importante reconhecer que modelos estatísticos e aprendizado de máquina extrapolam o escopo tradicional do pensamento computacional, pois operam sob princípios distintos, baseados em inferência probabilística e ajuste dinâmico a partir de dados. Isso não significa que essas **metodologias** sejam incompatíveis, mas sim que elas não devem ser indistintamente absorvidas dentro do pensamento computacional sem uma definição operacional rigorosa. Manter essa distinção conceitual fortalece tanto o ensino quanto a aplicabilidade do pensamento computacional, garantindo que ele preserve sua identidade fundamentada na lógica algorítmica e na construção sistemática de soluções computáveis.

Simplificações ou ampliações indiscriminadas distorcem a natureza própria do PC e podem comprometer a formação dos alunos, resultando em deficiências, como dificuldades em resolver problemas computacionais de forma estruturada. Isso ressalta a necessidade de uma reavaliação crítica das ideias de Wing no contexto educacional. Ao longo deste artigo, defendemos que a construção de algoritmos é o fator distintivo do pensamento computacional. Embora conceitos como abstração, decomposição e reconhecimento de padrões sejam essenciais, é sua aplicação no processo de planejar algoritmos que estabelece o pensamento computacional como algo único. Essa centralidade do algoritmo reflete a essência do pensamento computacional como uma ponte entre a cognição humana e os processos automatizáveis.

Diante das discussões sobre a delimitação conceitual do pensamento computacional, propomos o termo Pensamento Computacional Algorítmico-Probabilístico (PCAP) para distinguir sua concepção tradicional, baseada em algoritmos estruturados e determinísticos, daquela que integra metodologias probabilísticas e aprendizado de máquina. Essa distinção evita a fusão indiscriminada de conceitos e possibilita uma análise mais precisa de sua aplicabilidade no ensino e na pesquisa

científica. O PCAP mantém a identidade algorítmica do pensamento computacional tradicional, enquanto interage com modelos estatísticos sem que estes sejam confundidos com a lógica algorítmica. Essa diferenciação não é meramente terminológica, mas central para preservar a especificidade do pensamento computacional e garantir sua aplicação educacional e científica com precisão e coerência.

Mesmo que diversos autores tenham contribuído para o debate sobre o pensamento computacional, este artigo se concentra exclusivamente na obra de Wing, devido à sua influência, analisando seis textos representativos (2006, 2008, 2011, 2014, 2016, 2017). O objetivo é refletir criticamente sobre as bases conceituais do pensamento computacional tradicional, doravante simplesmente pensamento computacional, destacando pontos centrais que podem contribuir para sua consolidação e alinhando-os às demandas contemporâneas da educação e às práticas pedagógicas eficazes.

Apesar de que o pensamento computacional esteja intrinsecamente ligado à educação, este artigo se propõe exclusivamente a esclarecer conceitos fundamentais que definem a natureza do pensamento computacional. Questões relacionadas a teorias de aprendizagem e modelos de implementação pedagógica serão tratadas em trabalhos futuros. Essa delimitação permite um aprofundamento crítico necessário para resolver ambiguidades que impactam diretamente na forma como o pensamento computacional é compreendido e, conseqüentemente, praticado, servindo como base para discussões pedagógicas mais detalhadas em publicações subseqüentes.

2 FUNDAMENTAÇÃO TEÓRICA

Nesta seção, exploraremos os fundamentos teóricos do pensamento computacional, com ênfase nos conceitos de abstração, decomposição e reconhecimento de padrões. Apesar de que esses elementos são essenciais, argumentamos que é na aplicação dessas estratégias à construção de algoritmos que o pensamento computacional encontra sua identidade.

Jeanette Wing (2006) destaca a importância desses conceitos como pilares do pensamento computacional:

Computational thinking is using abstraction and decomposition when attacking a large complex task or designing a large complex system [...] Computational thinking is the ability to think at multiple levels of abstraction [...] An algorithm is an abstraction of a step-by-step procedure for taking input and producing some desired output (WING, 2006).²

² O pensamento computacional consiste em usar abstração e decomposição ao enfrentar uma tarefa grande e complexa ou ao projetar um sistema grande e complexo [...] O pensamento computacional é a habilidade de pensar em múltiplos níveis de abstração [...] Um algoritmo é uma abstração de um procedimento passo a passo para receber entradas e produzir algum resultado desejado (WING, 2006, p. 34).

A visão de Wing sobre abstração ressoa com a teoria de Piaget (1977), especialmente no que diz respeito à progressão de uma abstração empírica para uma reflexionante, aplicada ao contexto da construção de algoritmos. No entanto, embora a abstração, decomposição, reconhecimento de padrões e, implicitamente, a criatividade sejam elementos centrais destacados por Wing, é a aplicação desses elementos na construção de algoritmos que identifica o pensamento computacional. De fato, como destaca a autora, o pensamento computacional não deve ser confundido com programação, que pressupõe o domínio e o uso de uma linguagem de programação com sua sintaxe e recursos próprios. Ao contrário, o pensamento computacional refere-se às estratégias utilizadas para organizar o algoritmo, que, posteriormente, pode ser traduzido para qualquer linguagem de programação. Assim, o processo de planejar e estruturar algoritmos, sequências de passos não ambíguos com o potencial de serem traduzidos e executados por uma máquina, é o elemento que diferencia o pensamento computacional de outras formas de pensamento humano.

2.1 CONTRIBUIÇÕES DE PIAGET PARA O CONCEITO DE ABSTRAÇÃO

Segundo Piaget (1977), a abstração é um dos processos centrais no desenvolvimento cognitivo, permitindo a construção de conhecimentos mais complexos e generalizados a partir de experiências concretas:

A abstração é uma operação intelectual que consiste em isolar certos elementos da realidade, ignorando os demais, para concentrar-se em características particulares que são pertinentes para o objetivo de análise." (PIAGET, 1977, p. 17).

Jean Piaget (1977) descreve dois tipos principais de abstração: a abstração empírica e a abstração reflexionante. Esses tipos de abstração desempenham papéis diferentes no desenvolvimento cognitivo e são fundamentais para a compreensão de como os indivíduos constroem o conhecimento ao longo de suas vidas.

A abstração empírica refere-se ao processo de extração de características dos objetos com base em suas propriedades físicas observáveis. Esse tipo de abstração é diretamente ligado às interações sensoriais e motoras com o mundo exterior: "A abstração empírica consiste na extração de propriedades dos objetos materiais a partir de suas características diretamente perceptíveis, como forma, cor e textura" (PIAGET, 1977, p. 25).

A abstração reflexionante refere-se ao processo de abstrair características que não estão diretamente ligadas à percepção sensorial, mas que emergem da coordenação entre diferentes ações

mentais. Esse tipo de abstração é mais sofisticado e está ligado à capacidade de reflexão e generalização:

A abstração reflexiva é uma forma de abstração que não se refere a características materiais dos objetos, mas às operações mentais que o sujeito pode realizar sobre esses objetos, como as operações de agrupamento, ordenação e classificação" (PIAGET, 1977, p. 32).

Esses conceitos têm implicações profundas para a educação, especialmente na apreensão de conceitos complexos envolvidos nas estratégias relacionadas ao pensamento computacional. A diferenciação dos tipos de abstração fornece uma base para entender como os estudantes podem progredir de uma compreensão concreta para uma compreensão mais abstrata e generalizada, essencial para a construção de algoritmos que organizam soluções em etapas claras e executáveis.

2.2 A PERSPECTIVA ARISTOTÉLICA SOBRE A ABSTRAÇÃO

Além da abordagem psicológica de Piaget (1977, 1950), a filosofia de Aristóteles oferece outra perspectiva relevante sobre a abstração, destacando sua base na cognição humana universal: *The soul never thinks without a mental image... hence it is impossible for an object of thought to exist without the senses, and without an image, we do not learn, or understand anything. (Aristotle, De Anima, Book III, Chapter 7)*³

A imagem mental a que Aristóteles se refere é uma representação interna derivada, inicialmente, da percepção dos sentidos externos em contato com a realidade. Enquanto Piaget (1977) descreve a abstração em termos de desenvolvimento cognitivo, Aristóteles oferece uma visão complementar, posicionando a abstração como uma habilidade universal da mente humana. Em outras palavras, a abstração para o filósofo é uma capacidade intrínseca à natureza humana, ou seja, uma característica transversal da cognição humana, e não algo exclusivo de um tipo de pensamento.

Essa perspectiva reforça que, no pensamento computacional, a abstração está profundamente enraizada em habilidades humanas universais, mas ganha especificidade ao ser aplicada à formulação de algoritmos que organizam o conhecimento para resolução de problemas.

2.3 INFORMAÇÃO E CONHECIMENTO

Essa reflexão sobre a abstração nos leva a explorar como as informações externas podem ser transformadas em conhecimento sistematizado. Embora a abstração seja comum a todas as formas de

³ A alma nunca pensa sem uma imagem mental... portanto, é impossível que um objeto de pensamento exista sem os sentidos, e sem uma imagem, não aprendemos nem entendemos nada (Aristóteles, De Anima, Livro III, Capítulo 7).

cognição humana, ela adquire uma especificidade única no pensamento computacional na construção do algoritmo. Nesse contexto, os elementos captados e organizados são direcionados à criação de algoritmos, que estruturam soluções de maneira lógica e executável. Essa abordagem distingue o pensamento computacional ao traduzir informações dispersas, em sistemas organizados, permitindo a solução de problemas complexos por meio de algoritmos.

O processo de transformação de informação em conhecimento é um fenômeno intrinsecamente humano, fundamentado pela interação dinâmica entre sentidos externos, internos e realidade externa. Como sugere Locke (1996), os sentidos externos são as portas de entrada para as percepções do mundo, fornecendo as bases iniciais para o pensamento. Esses sentidos captam informações específicas do ambiente, que são posteriormente organizadas e processadas pelos sentidos internos. Piaget (1977) destaca que a interação contínua entre os esquemas cognitivos e o ambiente externo possibilita a adaptação progressiva do pensamento, favorecendo o desenvolvimento de estruturas cognitivas mais sofisticadas. Nesse contexto, os sentidos externos desempenham o papel de canais primários de percepção, enquanto os sentidos internos reorganizam essas percepções em conhecimento estruturado, permitindo avanços na cognição humana.

2.4 SENTIDOS EXTERNOS: PORTAS DE ENTRADA DA REALIDADE

Os sentidos externos - visão, audição, tato, olfato e paladar - são os canais primários para a captação de informações provenientes do ambiente. Cada sentido tem um papel específico e limitado no processo de percepção. Como já apontava Aristóteles (350 a.C.) em *De Anima*, "cada sentido percebe uma propriedade específica do objeto sensível", conectando o mundo exterior à mente por meio de canais especializados. Locke (1996) reforça essa perspectiva ao argumentar que os sentidos externos são a fonte primária das ideias na mente:

- Visão: Captura formas, cores, movimentos e profundidade, permitindo a percepção espacial e o reconhecimento de padrões visuais. Marr (1982) propõe um modelo computacional da visão, demonstrando como o sistema visual transforma estímulos sensoriais em representações estruturadas, essenciais para compreender o espaço e os objetos.
- Audição: Capta sons, ritmos e variações tonais, sendo fundamental para a comunicação e a interpretação de padrões auditivos.
- Tato: Percebe texturas, temperaturas e pressões, conectando a experiência corporal ao ambiente físico.

- Olfato e Paladar: Sensíveis a partículas químicas, desempenham um papel importante na identificação de sabores e cheiros. Gibson (1979) argumenta que a percepção ocorre de maneira direta, sem necessidade de inferências cognitivas complexas, e que os sentidos, de modo geral, captam afinidades, informações ambientais que orientam a interação prática com o meio.

Esses sentidos externos atuam como "portas de entrada" para as informações do ambiente, sendo incapazes de armazenar ou processar informações por si sós. Essa percepção inicial é posteriormente transformada pelos sentidos internos, que organizam e sistematizam as informações captadas.

2.5 SENTIDOS INTERNOS: TRANSFORMAÇÃO DA INFORMAÇÃO

Embora a divisão dos sentidos internos em categorias distintas seja uma construção teórica para compreender o processo cognitivo⁴, que é uma interação dinâmica e recursiva entre os sentidos humanos, é importante reconhecer as limitações inerentes a essa abordagem. Ellis (1980), em sua contribuição à psicolinguística, critica a ideia da **caixa preta** como um modelo absoluto para descrever os processos internos da cognição. Ele argumenta que, embora os mecanismos internos da mente sejam complexos e em grande parte inacessíveis diretamente, é possível investigá-los e representá-los através de modelos teóricos que se baseiam em dados empíricos.

Essa visão sustenta a utilização de categorias como memória, inteligência, imaginação, vontade e cogitativa, não como entidades separadas, mas como representações funcionais de um sistema integrado. Tais categorias são ferramentas analíticas que permitem capturar nuances do processo cognitivo, mesmo que na realidade esses elementos atuem de maneira inseparável e dinâmica.

Portanto, a categorização dos sentidos internos no presente artigo não pretende descrever uma realidade fragmentada, mas sim oferecer um modelo interpretativo que facilite a compreensão dos papéis atribuídos a diferentes dimensões do processo cognitivo. Esse posicionamento é consistente com a crítica de Ellis (1980) à caixa preta, pois reconhece que, embora as divisões sejam artificiais, elas fornecem *insights* valiosos sobre o funcionamento do pensamento humano.

A memória opera em diferentes níveis: Baddeley e Hitch (1974) propõem um modelo no qual a memória de trabalho não é um sistema unitário, mas composto por múltiplos componentes, incluindo o executivo central, que coordena a atenção e integra informações do loop fonológico (responsável

⁴ Neste artigo, o termo processo cognitivo é utilizado para descrever a interação dinâmica e recursiva entre os sentidos humanos.

pela retenção de informações verbais) e do esboço visuoespacial (que processa imagens mentais). Tulving (1972) diferencia a memória de longo prazo em memória episódica, que armazena eventos específicos, e memória semântica, que contém conhecimento factual e conceitual.

A teoria da carga cognitiva de Sweller (1994) sustenta que a memória de trabalho tem capacidade limitada e pode processar apenas uma quantidade restrita de informações simultaneamente. Quando a carga cognitiva extrínseca (elementos desnecessários) ou intrínseca (complexidade do material) excede essa capacidade, o aprendizado é comprometido. Em contraste, a memória de longo prazo armazena esquemas organizados, permitindo a redução da carga cognitiva e tornando a recuperação de informações mais eficiente.

No contexto do pensamento computacional, a formulação de algoritmos exige a manipulação simultânea de múltiplas informações, como variáveis, estruturas condicionais e padrões lógicos. Esse processo sobrecarrega a memória de trabalho se não for bem estruturado, dificultando o raciocínio e a solução de problemas. Para mitigar essa sobrecarga, estratégias como decomposição de problemas, uso de pseudocódigo e representação visual de algoritmos ajudam a distribuir a carga cognitiva, permitindo que os aprendizes internalizem conceitos computacionais de maneira mais eficiente. Dessa forma, ao reduzir a demanda sobre a memória de trabalho, a aprendizagem do pensamento computacional torna-se mais acessível e estruturada, facilitando a retenção e aplicação dos conceitos algorítmicos.

A inteligência, segundo Piaget (1950), manifesta-se por meio dos processos de assimilação e acomodação, que permitem ao indivíduo adaptar-se ao ambiente e construir esquemas cognitivos para compreender e interagir com diferentes contextos. A assimilação ocorre quando o indivíduo incorpora novas informações em estruturas cognitivas já existentes, utilizando seus esquemas prévios para interpretar o mundo. Por exemplo, uma criança que aprendeu o conceito de cachorro como sendo um animal de quatro patas pode inicialmente classificar um gato como um cachorro. Essa categorização demonstra o processo de assimilação, no qual a criança tenta acomodar uma nova experiência dentro de esquemas familiares.

Por outro lado, quando a nova informação não pode ser acomodada nos esquemas existentes, ocorre o processo de acomodação. Esse processo exige que o indivíduo modifique ou reorganize seus esquemas cognitivos para lidar com a novidade. No exemplo anterior, à medida que a criança percebe que o gato não compartilha todas as características de um cachorro, como latir ou ser de uma determinada espécie, ela cria um novo esquema para **gato**, ajustando sua compreensão do mundo. Assim, a inteligência não se limita a acumular informações, mas manifesta-se na capacidade de reorganizar e transformar esquemas para melhor se adaptar às complexidades do ambiente.

No contexto do pensamento computacional, esses processos de assimilação e acomodação desempenham um papel central. Por exemplo, quando um aluno aborda o conceito de algoritmos pela primeira vez, ele pode tentar assimilar esse conceito dentro de esquemas já existentes, como instruções passo a passo para realizar uma tarefa cotidiana. Contudo, ao perceber a necessidade de precisão e lógica formal nos algoritmos, ele deve acomodar esse conhecimento, ajustando seu entendimento inicial para incluir elementos específicos do pensamento computacional, como a necessidade de clareza e ausência de ambiguidade.

Além disso, o equilíbrio entre assimilação e acomodação reflete-se diretamente na forma como os estudantes desenvolvem a capacidade de abstração e decomposição no pensamento computacional. Inicialmente, eles tendem a abordar problemas complexos com base em esquemas já conhecidos (assimilação), no entanto, ao se depararem com obstáculos, precisam reorganizar suas estratégias, ajustando seus modelos mentais para incorporar novas formas de resolver a questão (acomodação).

A vontade é o motor que impulsiona a motivação, orientando o esforço intelectual necessário para transformar informações em conhecimento sistematizado. Conforme James (1890), a vontade desempenha um papel essencial na sustentação da atenção e na perseverança diante de desafios cognitivos. Adicionalmente, a Teoria da Autodeterminação (Deci & Ryan, 1985) oferece uma perspectiva contemporânea sobre os fatores que impulsionam a motivação, destacando que a autonomia, a percepção de competência e o senso de pertencimento desempenham um papel fundamental na qualidade do engajamento e da aprendizagem. Esses autores argumentam que quando os indivíduos percebem que têm controle sobre suas ações (autonomia), sentem-se eficazes ao enfrentar tarefas desafiadoras (competência) e experimentam conexões significativas com os outros (relacionamento), sua motivação intrínseca é fortalecida. Esse fortalecimento da motivação intrínseca sustenta o esforço contínuo necessário para a construção de conhecimento sistematizado, sendo, portanto, essencial para o aprendizado e a criação de algoritmos no contexto do pensamento computacional.

A imaginação combina elementos existentes em novas formas, permitindo a inovação e a resolução criativa de problemas. Aristóteles (350 a.C.) posiciona a imaginação como uma capacidade fundamental para o pensamento, vinculando-a à geração de imagens internas que tornam o raciocínio possível. No contexto moderno, Csikszentmihalyi (1990) argumenta que a criatividade emerge em estados de fluxo, nos quais a imaginação desempenha um papel central ao permitir associações inovadoras entre elementos distintos, resultando em novas ideias e soluções.

Sawyer (2012) discute a importância da imaginação na criatividade colaborativa, destacando como interações sociais podem potencializar a geração de ideias inovadoras. Ele argumenta que a

criatividade não ocorre isoladamente, mas é influenciada por práticas culturais e estímulos contextuais. Kaufman e Gregoire (2015), por sua vez, enfatizam a relevância da imaginação na criatividade cotidiana, sugerindo que ela é acessível a todos e pode ser cultivada por meio da abertura a novas experiências e da exploração de desafios intelectuais.

No contexto do pensamento computacional, a imaginação atua como um catalisador para a criação de algoritmos inovadores, permitindo a modelagem de soluções para problemas complexos. Root-Bernstein (2013) destaca a importância da imaginação científica na formulação de hipóteses e experimentos, ao estabelecer pontes entre abstrações teóricas e aplicações práticas. A partir dessa perspectiva, pode-se inferir que a mesma capacidade de transcender o óbvio e gerar novas ideias é fundamental para o pensamento computacional e para a computação criativa, que combina lógica e inovação na resolução de problemas.

A computação criativa é uma abordagem que explora como as ferramentas computacionais podem ser utilizadas para resolver problemas e para a expressão criativa. Essa perspectiva transcende a aplicação convencional de algoritmos para se concentrar na criação de artefatos digitais interativos, como histórias, jogos, animações e músicas. A computação criativa está diretamente ligada ao desenvolvimento do pensamento computacional, pois incentiva a abstração, a experimentação e a iteração como parte de um processo criativo.

Um dos avanços mais significativos na computação criativa foi liderado por Mitchel Resnick e sua equipe no Lifelong Kindergarten Group, do MIT Media Lab. Resnick e seus colaboradores lançaram em 2007 o Scratch, uma plataforma de programação visual projetada para capacitar crianças e jovens a criar e compartilhar projetos digitais. O Scratch foi desenvolvido com o objetivo de tornar a programação acessível e intuitiva, utilizando blocos de código que podem ser arrastados e encaixados, eliminando a necessidade de memorizar sintaxes complexas (cf. Resnick et al., 2009).

Além do Scratch, a computação criativa foi amplamente explorada por pesquisadores como John Maeda (2004), que enfatiza a fusão entre arte, ciência e tecnologia como meio para ampliar a expressão e inovação no design e na mídia digital. Da mesma forma, Reas e Fry (2001) desenvolveram a linguagem de programação *Processing*, voltada para designers e artistas visuais, consolidando-a como uma ferramenta essencial na interseção entre computação e artes visuais. Essas ferramentas e filosofias estabeleceram a computação criativa como uma área interdisciplinar que conecta ciência da computação, design e artes.

No contexto do pensamento computacional, a computação criativa contribui para a construção de algoritmos e modelos computacionais eficientes e para a estimulação da inovação e criatividade. Essa relação destaca que o pensamento computacional vai além de ser uma competência técnica,

apresentando-se também como um enfoque intelectual capaz de enriquecer diversas áreas do conhecimento e da expressão humana.

Por fim, a cogitativa é fundamental para o processamento e a articulação de pensamentos, especialmente por meio da linguagem, que Vygotsky (1986) descreve como o mediador central entre o pensamento e a comunicação. Para ele, o desenvolvimento da linguagem organiza ideias e estrutura o pensamento abstrato, permitindo a criação de narrativas coerentes e argumentações complexas. Estudos contemporâneos, como os de Tomasello (2014), reforçam a visão de que a cognição humana está intimamente ligada ao uso social da linguagem, sendo fundamental para a coordenação de esforços coletivos e o compartilhamento de conhecimento.

Além disso, Pinker (2007) destaca que a linguagem é uma das mais avançadas inovações evolutivas da humanidade, permitindo a transmissão cultural de ideias, tecnologias e valores. A partir dessa perspectiva, pode-se inferir que a linguagem atua como uma ferramenta essencial da cogitativa, auxiliando na formulação e estruturação do pensamento abstrato. No contexto do pensamento computacional, essa função se manifesta na conversão de conceitos abstratos em descrições linguísticas precisas, como pseudocódigos e especificações de algoritmos, que podem ser compartilhados e interpretados tanto por humanos quanto por máquinas.

Ferrari e Sternberg (1998) enfatizam que os processos de autorreflexão e metacognição são fundamentais na formulação de problemas e na resolução criativa, promovendo uma interação contínua entre análise e síntese. Essa interação é essencial no pensamento computacional, pois permite o refinamento de ideias e a transformação de raciocínios complexos em soluções estruturadas e comunicáveis.

Portanto, a cogitativa não se limita à organização interna do pensamento, mas também atua como a ponte que conecta os processos cognitivos à expressão clara e eficaz, tanto na comunicação interpessoal quanto na criação de sistemas computacionais. Sua importância transcende o domínio linguístico, sendo um componente da inovação e da resolução de problemas no mundo contemporâneo, especialmente em um contexto onde o pensamento computacional exige clareza estrutural e precisão conceitual.

Assim, os sentidos internos operam de maneira integrada, transformando as informações captadas pelos sentidos externos em conhecimento estruturado. Cada um desempenha um papel indispensável no processo cognitivo, contribuindo de forma interativa e adaptativa para a construção do saber. A seguir, apresentamos resumidamente os papéis atribuídos a cada sentido interno como uma forma de compreender melhor sua contribuição para esse processo, reconhecendo que essa divisão é uma abstração para facilitar a análise:

Memória (de trabalho e de longo prazo):

- Memória de trabalho: Atua como um espaço transitório para o processamento imediato de informações, permitindo sua conexão com conceitos mais amplos e contextos específicos (Baddeley e Hitch, 1974).
- Memória de longo prazo: Consolida e organiza informações de forma estruturada e acessível, garantindo que possam ser recuperadas e aplicadas em situações futuras (Tulving, 1972).

II. Inteligência:

- Identifica padrões e regularidades nas informações armazenadas, promovendo uma compreensão mais profunda (Piaget, 1950).
- Constrói generalizações e regras aplicáveis a múltiplos contextos, fundamentando-se em processos de dedução e indução.

III. Vontade:

- Atua como o motor da motivação, direcionando esforços cognitivos e emocionais para a consolidação do conhecimento (Ryan e Deci, 2000).
- Sustenta a continuidade no aprendizado, especialmente em contextos que exigem perseverança e dedicação.

IV. Imaginação:

- Produz as "imagens mentais" que tornam o pensamento possível, como descrito por Aristóteles (350 a.C.).
- Combina elementos existentes de maneira inovadora, criando soluções inéditas e contribuindo para a criatividade científica e prática (Csikszentmihalyi, 1990; Kaufman e Beghetto, 2009).

V. Cogitativa:

- Articula pensamentos por meio da linguagem, essencial para a comunicação e a formulação de conceitos complexos (Vygotsky, 1986).
- Integra os resultados da cognição em narrativas coerentes, promovendo o refinamento de argumentos e a resolução de problemas estruturados (Tomasello, 2014; Pinker, 2007).

2.6 O ESCOPO DO PENSAMENTO COMPUTACIONAL: CONSTRUÇÃO DE ALGORITMOS

Ainda que todos esses sentidos internos sejam fundamentais para o processo de transformação de informação em conhecimento, eles são universais e atuam em todas as áreas da cognição humana, portanto, não se confundem com um ou outro tipo de pensamento. Piaget (1950) observa que

mecanismos mentais como a assimilação e acomodação operam de forma generalizada, mas sua aplicação prática é moldada por contextos específicos. Portanto, não é a presença ou o funcionamento isolado de cada um deles que define o pensamento computacional, mas sim o escopo específico em que eles operam.

No contexto do pensamento computacional, a memória retém e estrutura informações relevantes, a inteligência identifica padrões e relações lógicas, a imaginação combina elementos para criar soluções inovadoras, a cogitativa organiza e articula o pensamento em representações formais, e a vontade impulsiona o esforço necessário para a resolução de problemas. Assim, o pensamento computacional distingue-se pela aplicação direcionada desses processos à criação de algoritmos, estruturas lógicas e não ambíguas que organizam informações de maneira executável por máquinas. Ao definir esse escopo, torna-se possível caracterizar o pensamento computacional como uma forma única de cognição humana que não se confunde com outras formas.

2.6.1 Pseudocódigo não é necessariamente algoritmo

Uma confusão possível no ensino de pensamento computacional está na ideia de que pseudocódigo equivale a algoritmo. Embora o pseudocódigo seja uma representação textual útil para expressar a lógica de um algoritmo, ele não possui, necessariamente, as características essenciais que definem um algoritmo.

Um algoritmo deve ser claro, sistemático e potencialmente executável por uma máquina. Para ilustrar essa distinção, considere os seguintes exemplos:

Pseudocódigo que não é um algoritmo: um exemplo exagerado

1. Lembre-se de algo bom que sua mãe te contou.
2. Reflita sobre isso no seu coração.
3. Diga como você se sentiu.

Este pseudocódigo, embora executável por um humano, demonstra como uma sequência de passos vagos e subjetivos falha em atender aos critérios necessários para ser considerado um algoritmo. A dependência de julgamento humano, a ausência de especificidade e a impossibilidade de execução prática por uma máquina destacam sua inadequação no contexto do pensamento computacional.

Pseudocódigo que é um algoritmo

1. Leia um número inteiro n .
2. Se o resto da divisão inteira de n por 2 ($n \% 2$) for igual a 0, exiba "Par".
3. Caso contrário, exiba "Ímpar".

Este segundo exemplo atende aos critérios de clareza, sistematicidade e executabilidade por uma máquina. A lógica está claramente definida, e os passos podem ser implementados diretamente em qualquer linguagem de programação.

A falta de distinção entre pseudocódigo e algoritmo nas práticas educacionais reflete e reforça a ambiguidade presente na definição de pensamento computacional de Wing (2006). Ao aceitar processos gerais como parte do escopo do pensamento computacional, incluindo representações textuais vagas, a definição dilui a especificidade algorítmica central ao conceito.

2.7 OS CONDICIONAMENTOS NO DESENVOLVIMENTO DO PENSAMENTO COMPUTACIONAL

Ainda que este artigo tenha enfatizado os mecanismos cognitivos que fundamentam o pensamento computacional, é primordial reconhecer que esses mecanismos não operam em um vácuo. Eles são profundamente influenciados por uma série de condicionamentos sociais, culturais, biológicos e econômicos, que moldam tanto as oportunidades quanto os desafios enfrentados pelos indivíduos na construção do pensamento computacional.

Reconhecer os condicionamentos que moldam os mecanismos cognitivos é muito importante para uma visão mais abrangente. Essas influências ressaltam a importância de criar políticas educacionais e sociais que mitiguem desigualdades e promovam acesso equitativo ao aprendizado de competências computacionais.

Assim, esta seção serve como um reconhecimento de que, embora os mecanismos cognitivos sejam universais, sua manifestação e desenvolvimento são profundamente condicionados por fatores do contexto. Explorar essas interações é uma tarefa importante para futuras pesquisas e práticas educacionais que visem a inclusão e o desenvolvimento equitativo do pensamento computacional.

2.7.1 O papel dos condicionamentos sociais e culturais

O desenvolvimento cognitivo humano é influenciado por interações sociais e ferramentas culturais. Vygotsky (1986) ressalta que a aprendizagem e o desenvolvimento ocorrem em um processo de mediação entre o indivíduo e o ambiente sociocultural, sendo moldados pela linguagem, pelas práticas culturais e pelas ferramentas disponíveis. No caso do pensamento computacional, desigualdades no acesso a recursos educacionais, tecnologias digitais e ambientes estimulantes podem impactar significativamente seu desenvolvimento.

Contudo, a ausência de infraestrutura tecnológica não é, por si só, uma barreira intransponível para o aprendizado do pensamento computacional. Atividades desplugadas, que não dependem de

computadores ou dispositivos digitais, oferecem caminhos para desenvolver habilidades como abstração, reconhecimento de padrões e decomposição na construção de algoritmos. O que se torna essencial, nesse contexto, é a preparação e formação dos professores para mediar esse processo de forma eficiente. Além disso, diferenças culturais na valorização de distintas formas de raciocínio estruturado podem influenciar o interesse e a motivação dos alunos, destacando a importância de práticas educativas que tornem o pensamento computacional acessível e relevante para diferentes realidades.

2.7.2 Condicionamentos biológicos e individuais

Fatores biológicos também desempenham um papel relevante no desenvolvimento do pensamento computacional. A maturidade cerebral, por exemplo, é um fator determinante para a capacidade de abstração e resolução de problemas complexos. Piaget (1950) descreve que o desenvolvimento cognitivo ocorre em estágios sucessivos, nos quais habilidades mais avançadas emergem à medida que a criança atinge níveis específicos de maturação e interação com o ambiente.

Além disso, diferenças individuais, incluindo variações no funcionamento cognitivo, estilos de aprendizagem e a neurodiversidade, influenciam a maneira como cada pessoa internaliza e aplica conceitos relacionados ao pensamento computacional.

2.7.3 Condicionamentos econômicos

O desenvolvimento cognitivo humano ocorre em constante interação com o meio social e cultural. Vygotsky (1986) ressalta que a aprendizagem é mediada pela linguagem, pelas práticas culturais e pelas ferramentas disponíveis no ambiente. No caso do pensamento computacional, desigualdades no acesso a recursos educacionais e tecnologias digitais podem impactar o aprendizado.

3 EVOLUÇÃO E DESAFIOS: UMA ANÁLISE CRÍTICA DA OBRA DE WING

Desde 2006, Jeanette Wing tem desempenhado um papel fundamental na consolidação do pensamento computacional (PC) como uma habilidade essencial na educação e em áreas interdisciplinares. Por meio de seus artigos, Wing apresentou definições centrais para o conceito e explorou suas aplicações e desafios em diversos contextos. Esta seção analisa criticamente as principais contribuições de Wing em seis artigos-chave (2006, 2008, 2011, 2014, 2016, 2017) à luz de cinco categorias: definição de pensamento computacional, centralidade dos algoritmos, elementos cognitivos universais, impacto educacional e práticas pedagógicas, e ampliação conceitual e ambiguidades.

3.1 DEFINIÇÃO DE PENSAMENTO COMPUTACIONAL

Wing introduziu o pensamento computacional como uma habilidade a ser cultivada em seu artigo seminal de 2006, definindo-o como processos computacionais “executados por um humano ou por uma máquina”. No entanto, o uso do conector lógico “OR” na expressão “*human or machine*” gera ambiguidades conceituais importantes. De acordo com a tabela verdade da lógica proposicional, a operação “OR” é inclusiva, o que significa que os processos referidos por Wing (2006) podem ser executados exclusivamente por humanos, exclusivamente por máquinas ou por ambos. Essa formulação amplia o escopo do pensamento computacional, permitindo que processos puramente humanos, não executáveis por máquinas, sejam considerados parte do conceito.

Nos textos subsequentes, Wing não revisita explicitamente essa definição, nem estabelece critérios rigorosos para diferenciar processos exclusivamente humanos daqueles que são essencialmente computacionais. Em Wing (2008), por exemplo, o pensamento computacional continua a ser apresentado como uma habilidade universal, sem uma delimitação clara que evite a confusão com outros tipos de raciocínio humano. No artigo de 2011, a autora enfatiza que o PC envolve "conceitos fundamentais da ciência da computação" e deve ser tratado como um pilar essencial da educação moderna, mas ainda sem resolver a questão de sua abrangência. Já em Wing (2014), o foco passa a ser a implementação do PC no ensino, porém, mais uma vez, sem uma redefinição que elimine a ambiguidade do escopo conceitual.

Mesmo em seu artigo mais recente analisado (Wing, 2017), a indefinição persiste. A autora reitera que o pensamento computacional pode ser aplicado a todas as áreas do conhecimento e reforça sua importância na formação de cidadãos críticos e criativos. No entanto, essa abordagem, sem uma distinção clara entre processos computacionais e humanos, mantém a possibilidade de interpretações excessivamente amplas que diluem a identidade do pensamento computacional e dificultam sua consolidação como um campo formalmente delimitado.

3.2 CENTRALIDADE DOS ALGORITMOS

Ao longo de sua obra, Wing reconhece a importância dos algoritmos como parte inseparável do pensamento computacional. No entanto, ao incluir no PC a capacidade de “resolver problemas, projetar sistemas e entender o comportamento humano⁵ com base em conceitos fundamentais da ciência da computação” (Wing, 2006), a autora reforça a ambiguidade já presente na definição inicial e expande indevidamente o escopo do conceito. A inclusão do entendimento do comportamento

⁵ Grifo nosso

humano no pensamento computacional sugere um deslocamento do foco técnico para uma cognição mais ampla, sem a exigência de algoritmos claros e sistematizáveis.

Nos textos posteriores, essa questão não é resolvida. Em Wing (2011), por exemplo, há um reforço da ideia de que o pensamento computacional envolve "a formulação e a resolução de problemas de forma que possam ser executados por um agente de processamento de informações⁶", mas a relação entre esse agente e a necessidade de algoritmos estruturados não é aprofundada. Em Wing (2014), quando discute a implementação do PC no currículo escolar, a autora defende a necessidade de ensinar as crianças a pensar computacionalmente desde cedo, mas sem definir com clareza o que distingue essa habilidade de outras formas de pensamento lógico ou criativo.

Além disso, Wing (2017) continua a enfatizar a transversalidade do pensamento computacional em diferentes disciplinas, sem delimitar de maneira rigorosa a centralidade dos algoritmos. Essa ausência de definição específica compromete a clareza do conceito e permite interpretações que afastam o pensamento computacional de sua base algorítmica e sistemática.

3.3 ELEMENTOS COGNITIVOS UNIVERSAIS

Wing destaca consistentemente a importância de elementos cognitivos como abstração, decomposição e reconhecimento de padrões no pensamento computacional. No entanto, sua abordagem não diferencia claramente esses elementos de formas gerais de cognição. Em Wing (2008), por exemplo, a autora afirma que o PC envolve "pensar de forma abstrata", mas não explica como essa abstração se distingue daquela empregada em outras disciplinas, como matemática ou filosofia.

Em Wing (2011), a ênfase na abstração permanece, mas há pouca diferenciação entre a abstração necessária para a formulação de algoritmos e a abstração enquanto processo cognitivo amplo. Já em Wing (2016), a autora reforça a necessidade de desenvolver essas habilidades no ensino, mas sem abordar como sua aplicação específica no pensamento computacional pode ser diferenciada de outras práticas educacionais.

A ausência de um critério claro para delimitar a aplicação dos elementos cognitivos no pensamento computacional contribui para a confusão conceitual. Ao apresentar a abstração, a criatividade e o reconhecimento de padrões como componentes essenciais do PC sem estabelecer sua relação específica com a criação de algoritmos, Wing amplia o conceito a ponto de torná-lo indistinto de outras formas de raciocínio humano.

⁶ Grifo nosso

3.4 IMPACTO EDUCACIONAL E PRÁTICAS PEDAGÓGICAS

Wing (2014) e Wing (2017) enfatizam a necessidade de incluir o pensamento computacional na educação desde os anos iniciais. A autora argumenta que o PC pode ser ensinado por meio de atividades desplugadas e estratégias didáticas interdisciplinares. No entanto, a indefinição conceitual do pensamento computacional impacta diretamente sua implementação educacional.

Por exemplo, em Wing (2014), há uma preocupação com a formação de professores para ensinar pensamento computacional, mas sem uma definição clara do que precisa ser ensinado. Em Wing (2017), a autora reforça a importância da transversalidade do PC, mas não estabelece como diferenciá-lo de outras formas de resolução de problemas. A falta de clareza leva a currículos escolares que podem interpretar erroneamente o pensamento computacional, diluindo sua identidade e resultando em práticas que pouco refletem sua essência algorítmica.

3.5 AMPLIAÇÃO CONCEITUAL E AMBIGUIDADES

Desde sua introdução em 2006, o pensamento computacional tem sido apresentado como uma competência aplicável a todas as áreas do conhecimento. No entanto, Wing (2011) reforça a dificuldade de estabelecer seus limites ao afirmar que "todos devem aprender pensamento computacional". Essa proposta, embora interessante do ponto de vista educacional, não estabelece critérios objetivos que diferenciem o PC de outros processos cognitivos.

Em Wing (2014), há uma tentativa de conectar o pensamento computacional a diferentes disciplinas, mas sem definir exatamente quais elementos são exclusivos dessa forma de pensamento. Wing (2017) amplia ainda mais essa visão, sugerindo que o PC pode ser um modelo mental geral aplicável à ciência, à engenharia e até mesmo às humanidades. Essa abordagem pode ser interpretada de maneira excessivamente abrangente, tornando o pensamento computacional indistinguível de outras formas de raciocínio analítico e de resolução de problemas.

Dessa forma, a questão não está na interdisciplinaridade do pensamento computacional, mas na ausência de um modelo operacional que estabeleça seus limites conceituais. Para que sua aplicação seja eficaz, é essencial definir seus princípios fundamentais, especialmente no que diz respeito à centralidade dos algoritmos e à sistematicidade de suas soluções.

4 CONSIDERAÇÕES FINAIS

A análise detalhada das publicações de Jeanette Wing ao longo de mais de uma década revelou contribuições significativas para a disseminação do conceito de pensamento computacional, mas também evidenciou ambiguidades que precisam ser abordadas para garantir sua aplicação teórica e

prática. Desde sua formulação inicial, o conceito de pensamento computacional foi apresentado como uma competência ampla e essencial, mas com limites conceituais pouco claros, como a inclusão de processos exclusivamente humanos que extrapolam o escopo computacional. O uso do operador lógico "OR" na definição "*Computational thinking builds on the power and limits of computing processes, whether they are executed by a human OR by a machine*" (Wing, 2006) gera ambiguidades que dificultam a delimitação do pensamento computacional em relação a outras formas de cognição. Embora Wing tenha expandido a discussão sobre o pensamento computacional em trabalhos posteriores, a ausência de uma revisão explícita da definição inicial manteve a ambiguidade conceitual presente em sua formulação original.

Além disso, a revisão de sua obra evidencia uma persistente falta de centralidade no papel dos algoritmos, elemento que define a especificidade do pensamento computacional. Conquanto conceitos como abstração, decomposição e reconhecimento de padrões sejam essenciais, é sua aplicação no processo de construção de algoritmos que estabelece o pensamento computacional como algo único. A diluição conceitual do PC resulta em dificuldades educacionais, uma vez que a indefinição daquilo que deve ser ensinado compromete sua implementação eficaz.

Além da indefinição conceitual identificada na obra de Wing, outra questão contemporânea que exige maior clareza na delimitação do pensamento computacional diz respeito à relação entre modelos probabilísticos e a estrutura algorítmica tradicional. Embora modelos probabilísticos possam ser ferramentas auxiliares poderosas, eles não eliminam a necessidade de algoritmos estruturados, que continuam sendo a base operacional do PC. O pensamento computacional e os modelos probabilísticos não são concorrentes, mas sim complementares, uma vez que o PC fornece a estrutura lógica e formal sobre a qual os modelos probabilísticos podem operar para lidar com incertezas.

Enquanto o pensamento computacional enfatiza a construção de soluções estruturadas, determinísticas e sistemáticas, os modelos probabilísticos lidam com incerteza, variabilidade e padrões emergentes. Em muitos casos, algoritmos determinísticos e técnicas estatísticas trabalham juntos para otimizar processos, como na filtragem de dados, aprendizado de máquina e inteligência artificial. Por exemplo, um sistema de recomendação pode combinar estratégias algorítmicas tradicionais para organizar e recuperar informações, ao mesmo tempo que modelos probabilísticos refinam as sugestões com base em padrões estatísticos e inferências adaptativas.

Essa relação não descaracteriza o pensamento computacional, mas sim o complementa, oferecendo novas camadas de complexidade e eficiência. No entanto, é preciso ressaltar que a lógica algorítmica permanece essencial, pois modelos probabilísticos e aprendizado profundo funcionam como ferramentas analíticas complementares dentro de um *framework* algorítmico mais amplo. Em

última instância, o pensamento computacional fornece a estrutura lógica necessária para interpretar, validar e integrar modelos probabilísticos de maneira confiável e compreensível.

Assim, ao invés de substituir os algoritmos, os modelos probabilísticos se tornam um elemento adicional dentro de um ecossistema computacional mais amplo, no qual o pensamento computacional continua sendo o alicerce fundamental. Essa interação entre lógica determinística e inferência estatística ilustra a necessidade de uma definição clara do PC, que reconheça sua capacidade de integrar múltiplas concepções sem perder sua identidade algorítmica central.

Dessa forma, a crítica à indefinição conceitual do pensamento computacional permanece pertinente. Se qualquer estratégia de resolução de problemas for indistintamente classificada como PC, o conceito se torna difuso e perde seu caráter distintivo, comprometendo sua aplicabilidade e sua relevância acadêmica e pedagógica. A lógica algorítmica deve continuar sendo a base do pensamento computacional, pois é a estrutura capaz de garantir clareza, previsibilidade e sistematicidade na resolução de problemas computáveis.

Para garantir o avanço do pensamento computacional como pilar da educação contemporânea, é necessário alinhar teoria e prática. Isso requer um esforço coordenado para revisar definições conceituais, fortalecer a formação docente e implementar atividades pedagógicas que reflitam a centralidade dos algoritmos.

Definir claramente os limites do pensamento computacional é essencial para fortalecer sua identidade conceitual, garantindo que sua implementação na educação e na pesquisa científica seja precisa e eficaz.

REFERÊNCIAS

- ARISTÓTELES. De Anima. Tradução de W. S. Hett. Harvard University Press, 1986. (Original publicado em 350 a.C.).
- BADDELEY, A.; HITCH, G. Working memory. *Psychology of Learning and Motivation*, v. 8, p. 47-89, 1974.
- CENTRO DE INOVAÇÃO PARA A EDUCAÇÃO BRASILEIRA. CIEB: Notas Técnicas #21 Inteligência artificial na educação básica: novas aplicações e tendências para o futuro. São Paulo: CIEB, 2024. E-book em PDF. Disponível em: https://cieb.net.br/wp-content/uploads/2024/06/Inteligencia-Artificial-na-Educacao-Basica_2024.pdf. Acesso em: 26 ago. 2024.
- CSIKSZENTMIHALYI, M. *Flow: The Psychology of Optimal Experience*. New York: Harper & Row, 1990.
- DECI, E. L.; RYAN, R. M. *Intrinsic Motivation and Self-Determination in Human Behavior*. Springer Science & Business Media, 1985.
- ELLIS, N. C. *The Study of Second Language Acquisition*. Oxford: Pergamon, 1980.
- FERRARI, M.; STERNBERG, R. J. *Self-awareness: Its Nature and Development*. Guilford Press, 1998.
- GIBSON, J. J. *The Ecological Approach to Visual Perception*. Boston: Houghton Mifflin, 1979.
- JAMES, W. *The Principles of Psychology (Vols. 1 & 2)*. New York: Henry Holt and Company, 1890.
- KAUFMAN, S. B.; GREGOIRE, C. *Wired to Create: Unraveling the Mysteries of the Creative Mind*. Perigee Books, 2015.
- LOCKE, J. *An Essay Concerning Human Understanding*. Editado por Peter H. Nidditch. Oxford: Clarendon Press, 1996.
- MAEDA, J. *Creative Code: Aesthetics and Computation*. Cambridge, MA: MIT Press, 2004.
- MARR, D. *Vision: A Computational Investigation into the Human Representation and Processing of Visual Information*. San Francisco: W. H. Freeman, 1982.
- OLIVEIRA, C. M.; PEREIRA, R. Desenvolvimento do Pensamento Computacional no Ensino Superior em Ciência da Computação. In: VIII Congresso Brasileiro de Informática na Educação (CBIE 2019). Anais dos Workshops do VIII Congresso Brasileiro de Informática na Educação. Curitiba: Universidade Federal do Paraná (UFPR), 2019. DOI: 10.5753/cbie.wcbie.2019.1502.
- PINKER, S. *The Stuff of Thought: Language as a Window into Human Nature*. Viking, 2007.
- PIAGET, J. *The Development of Thought: Equilibration of Cognitive Structures*. Cambridge, MA: Harvard University Press, 1977.

PIAGET, J. The Psychology of Intelligence. London: Routledge and Kegan Paul, 1950. [Original em francês: La Psychologie de l'Intelligence (1947)].

REAS, C.; FRY, B. Processing: A flexible software sketchbook and a language for learning how to code within the context of the visual arts. MIT Media Lab, 2001.

RESNICK, M.; et al. Scratch: Programming for All. Communications of the ACM, v. 52, n. 11, p. 60-67, 2009. doi:10.1145/1592761.1592779. Disponível em: <https://dl.acm.org/doi/10.1145/1592761.1592779>. Acesso em: 27 dez. 2024.

ROOT-BERNSTEIN, R. S. Sparks of Genius: The 13 Thinking Tools of the World's Most Creative People. Mariner Books, 2013.

SAWYER, R. K. Explaining Creativity: The Science of Human Innovation. Oxford University Press, 2012.

SWELLER, J. Cognitive Load Theory, Learning Difficulty, and Instructional Design. Learning and Instruction, v. 4, n. 4, p. 295-312, 1994.

TOMASELLO, M. A Natural History of Human Thinking. Harvard University Press, 2014.

TULVING, E. Episodic and semantic memory. In: TULVING, E.; DONALDSON, W. (Eds.). Organization of Memory. New York: Academic Press, 1972. p. 381-403.

VYGOTSKY, L. S. Thought and Language. Tradução de Alex Kozulin. Cambridge, MA: MIT Press, 1986. (Original em russo publicado em 1934).

WING, J. M. Computational Thinking. Communications of the ACM, v. 49, n. 3, p. 33-35, 2006. doi:10.1145/1118178.1118215. Disponível em: <https://www.cs.cmu.edu/afs/cs/usr/wing/www/publications/Wing06.pdf>. Acesso em: 26 ago. 2024.

WING, J. M. Five deep questions in computing. Communications of the ACM, v. 51, n. 1, p. 58-60, 2008. doi:10.1145/1327452.1327479. Disponível em: <http://www.cs.cmu.edu/afs/cs/usr/wing/www/publications/Wing08.pdf>. Acesso em: 26 ago. 2024.

WING, J. M. Computational thinking: What and why. The Link, Carnegie Mellon University, março 2011. Disponível em: <http://www.cs.cmu.edu/link/research-notebook-computational-thinking-what-and-why>. Acesso em: 26 ago. 2024.

WING, J. M. Computational thinking benefits society. Microsoft Research Blog, 10 jan. 2014. Disponível em: <http://www.cs.cmu.edu/afs/cs/usr/wing/www/publications/Wing14.pdf>. Acesso em: 26 ago. 2024.

WING, J. M. Computational thinking, 10 years later. Communications of the ACM, março 2016. Disponível em: <https://www.acm.org/media/news/2016/wings-computational-thinking>. Acesso em: 26 ago. 2024.

WING, J. M. Computational thinking's influence on research and education for all. Italian Journal of Educational Technology, v. 25, n. 2, p. 7-14, 2017. doi:10.17471/2499-4324/922. Disponível em: <https://ijet.itd.cnr.it/article/view/922>. Acesso em: 26 ago. 2024.