

Recomendações para Introdução do Pensamento Computacional na Educação Básica

André Raabe^{1,2}, Marli V. Vieira², André L.M. Santana¹, Filipe Gonçalves¹, Julia Bathke³

¹Mestrado em Computação Aplicada
²Programa de Pós Graduação em Educação
³Graduação em Ciência da Computação
Universidade do Vale do Itajaí - UNIVALI
raabe@univali.br

***Abstract:** This paper proposes the challenge of building recommendations for Computational Thinking to become a regular knowledge in basic education. Focusing on workload increase provided by the expansion of full-time schools, as preconized by the National Education Plan (Brasil, 2014), the paper suggests the creation of a discussion community for building guidelines and models of educational activities contextualized in Brazilian reality.*

***Resumo:** Este artigo propõe como desafio a construção de recomendações para Inserção do Pensamento Computacional como conhecimento regular na educação básica. Para tal propõe que sejam formada uma comunidade de discussão tendo como meta construir diretrizes e modelos de atividades educacionais contextualizados a realidade brasileira, focalizando no aumento da carga horária proporcionado pela ampliação das escolas em tempo integral conforme proposto pelo Plano Nacional de Educação (Brasil, 2014).*

1. Introdução

No Brasil, iniciativas de introdução ao Pensamento Computacional têm sido multiplicadas a cada ano e publicadas nos veículos da Comissão especial de Informática na Educação da SBC e também no Workshop de Educação em Computação. Muitas apontam resultados promissores relacionados ao engajamento dos estudantes e o potencial de interdisciplinaridade envolvida. Este cenário, impulsionado também por iniciativas de empresas e de outros países, em especial os Estados Unidos da América, tem tornado a possibilidade de inserção de conhecimentos relativos à computação como candidatos e serem aderidos pela Educação Básica.

A Meta 6 no Plano Nacional de Educação (Brasil, 2014) é: “oferecer educação em tempo integral em, no mínimo, 50% (cinquenta por cento) das escolas públicas, de forma a atender, pelo menos, 25% (vinte e cinco por cento) dos (as) alunos (as) da educação básica”. O plano propõe como estratégia, relacionada a esta meta, a promoção da oferta de educação básica pública em tempo integral, por meio de atividades de acompanhamento pedagógico e multidisciplinares, inclusive culturais e esportivas, de forma que o tempo de permanência dos (as) alunos (as) na escola, ou sob sua responsabilidade, passe a ser igual ou superior a 7 (sete) horas.

Entende-se que desta forma, as escolas de educação integral terão a possibilidade de oferecer aos estudantes atividades inovadoras e que ampliem a gama de conceitos abordados de uma forma não disciplinar. Vislumbra-se aí um cenário

propenso a proposição de atividades relacionadas ao desenvolvimento do pensamento computacional em suas mais diferentes modalidades como: utilização jogos de programar; construção de aplicações pelos estudantes com ambientes de programação; a robótica educacional, a configuração de ambientes integrados onde a tecnologia possui um papel central como *FabLabs*, *MakerSpaces* e outros.

Considera-se fundamental nesta discussão o resgate e o fortalecimento da perspectiva construcionista introduzida por Papert (1980) e trazida a o Brasil no século passado pelos precursores da área de Informática na Educação. Em especial a experiência advinda da introdução da linguagem Logo nas escolas Brasileiras necessita ser consultada a fim de não repetir equívocos e aprender com os acertos do passado.

Neste sentido este trabalho propõe discutir estratégias para a definição de parâmetros que possibilitem a inclusão do pensamento computacional na Educação Básica tendo como premissa os fundamentos do construcionismo. Entende-se que o principal caminho a ser seguido é apoiar-se no exemplo entidades de outros países (como a ISTE e CSTA) que se organizaram coletivamente com o fim de criar e disponibilizar programas de atividades sugeridas e materiais de orientações aos professores. No entanto, a simples tradução destes materiais não será suficiente, é necessário que se construa uma proposta aderente às práticas e a realidade educacional brasileira.

O artigo está organizado como segue: a seção 2 discute as principais iniciativas de introdução ao pensamento computacional realizadas em outros países; a seção 3 analisa os materiais disponíveis e mais utilizados para introdução ao Pensamento Computacional; a seção 4 busca defender a abordagem do pensamento computacional em uma perspectiva construcionista; a seção 5 sugere ações para articulação de uma rede de pesquisadores e uma agenda de pesquisa voltada ao compartilhamento das experiências e a construção de modelos de boas práticas.

2. A Disseminação do Pensamento Computacional

Surpreende o impacto do artigo de Wing (2006) que introduz o termo pensamento computacional. O artigo é um artigo curto (três páginas) e expressa a opinião da autora, não estando fundamentado em resultados de pesquisa. A partir dele o termo pensamento computacional, ainda que tenha sido mencionado anteriormente por outros, passou a ser amplamente discutido. A popularidade do artigo de certa forma reflete a ânsia que existia na comunidade de pesquisadores de Educação em Computação, Tecnologia Educacional, Informática na Educação e áreas afins pela conquista de espaço e de argumentos que pudessem renovar e aprofundar o uso de tecnologia nas escolas.

No entanto o termo pensamento computacional não é um constructo definido em termos precisos. Existem muitas diferentes definições e compreensões para o termo. Ao analisar os resultados de um Workshop realizado pelo *National Research Council* dos Estados Unidos para discutir o Escopo a Natureza do pensamento computacional (NRC, 2015) é possível perceber uma diversidade de visões que ampliam muito o escopo do termo. No documento, as sobreposições com outros termos já cunhados anteriormente como o pensamento matemático e pensamento de engenharia (*engineering thinking*) são apontadas, mas não solucionadas. Não é apresentada nenhuma argumentação na direção de esclarecer se é um novo tipo de pensamento, ou uma combinação de vários existentes e ainda se é adequado realmente chamar de “pensamento”. Esta característica gera uma dificuldade para discussão sobre a forma de proporcionar o desenvolvimento do

pensamento computacional, sobre como mensurá-lo ou avaliá-lo e conseqüentemente como difundi-lo de forma consistente para outras áreas, em especial a educação.

Dentre as muitas definições de Pensamento Computacional que surgiram após o termo ter sido disseminado pelo artigo de Wing (2006), a definição construída pela *International Society for Technology in Education* (ISTE) em conjunto com a *Computer Science Teachers Association* (CSTA) é a que melhor possibilita a discussão pautada em elementos objetivos.

Nesta definição Pensamento computacional é um processo de resolução de problemas que inclui (não somente) as seguintes características: (i) Formulação de problemas de forma que computadores e outras ferramentas possam ajudar a resolvê-los; (ii) Organização lógica e análise de dados; (iii) Representação de dados através de abstrações como modelos e simulações; (iv) Automatização de soluções através do pensamento algorítmico; (v) Identificação, análise e implementação de soluções visando a combinação mais eficiente e eficaz de etapas e recursos (vi) Generalização e transferência de soluções para uma ampla gama de problemas (CSTA, 2015).

Wing (2006) defende que o pensamento computacional é uma habilidade fundamental para qualquer um, não apenas para cientistas da computação. Juntamente com a leitura, a escrita e a aritmética, devíamos adicionar o pensamento computacional na habilidade analítica de cada criança.

No Brasil, iniciativas de introdução ao Pensamento Computacional como (BARCELOS E SILVEIRA, 2012; ANDRADE, 2013; FRANÇA e AMARAL, 2013; VIEL, RAABE e ZEFERINO, 2014) apontam resultados promissores relacionados ao engajamento dos estudantes e o potencial de interdisciplinaridade envolvidos. Outras iniciativas como os CodeClubs¹, a criação de ambientes de programação em português como Portugol Studio (NOSCHANG, et al. 2014) e iniciativas de incentivo ao ensino de programação como o Programa² tem cada dia alcançado mais adeptos.

A programação de computadores é uma atividade que integra muitas das características do pensamento computacional. Blikstein (2013) afirma que muito trabalho tem sido feito para tornar a programação de computadores fácil de aprender. Ferramentas como Scratch (Resnick et al., 2009) e iniciativas como a Hora do Código do Code.org³ atingiram um índice de popularidade sem precedentes, o que tornou a ideia de ensinar programação nas escolas muito menos impactante do era muitos anos antes, quando Papert (1980) propôs o mesmo com a linguagem Logo.

Grandes empresas estão desenvolvendo produtos e projetos que apoiam o desenvolvimento do Pensamento Computacional, como é o caso da Microsoft, desenvolvedora do software Kodu⁴ e a empresa Lego que já a muitos anos criou o kit de robótica Lego MindStorms e recentemente o We Do. O Google desenvolve os projetos *Exploring Computational Thinking*⁵, *CS First*⁶, *Code-In*⁷, *Computer Science for High*

¹ <http://www.codeclubbrasil.org>

² <http://www.programae.org.br>

³ <http://www.code.org>

⁴ <http://www.kodugamelab.com/>

⁵ <http://www.google.com/edu/resources/programs/exploring-computational-thinking/>

⁶ <http://www.cs-first.com/>

⁷ <https://developers.google.com/open-source/gci/>

*School*⁸ voltados principalmente ao público em idade escolar do ensino fundamental e médio.

Universidades e organizações não governamentais também estão produzindo ferramentas e materiais para auxiliar o desenvolvimento do Pensamento Computacional. O MIT Media Lab, desenvolvedor da ferramenta Scratch (Resnik et al., 2009) que é a mais amplamente utilizada em atividades do pensamento computacional. A Carnegie Mellon University, que distribui a IDE Alice, um ambiente de programação 3D voltado a aprendizes de programação, e mantém um centro de estudos do Pensamento Computacional⁹. O *CollegeBoard* que em parceria com a *National Science Foundation* dos Estados Unidos da América, desenvolveram o curso *CS Principles*¹⁰ que além de definir um conjunto de práticas para o desenvolvimento do Pensamento Computacional, oferece um currículo de atividades para os professores poderem aplicar em sala de aula.

A maioria das iniciativas está sendo desenvolvida nos Estados Unidos da América, caracterizando-se como o país com mais atividade na área do desenvolvimento do Pensamento Computacional. O segundo país com mais atividades na área é a Inglaterra, provavelmente devido ao conjunto de recomendações publicado pela Royal Society (2012) para a renovação do currículo escolar na área da Ciência da Computação, outros países onde existem relatos de experiências com o Pensamento Computacional são Brasil, Qatar, Holanda, Finlândia, Turquia, Itália e Índia.

Muitas das ferramentas e projetos citados são livres, podendo ser utilizados e disseminados sem qualquer custo em escolas ou qualquer outro ambiente, seja público ou privado.

3. Principais Recursos de apoio a Introdução ao Pensamento Computacional

Algumas iniciativas mencionadas na seção anterior possuem materiais de apoio à introdução ao pensamento computacional na forma de documentos, modelos, exemplos, sugestões de atividades e etc. Destaca-se a seguir materiais que são amplamente utilizados e que foram concebidos para apoio a introdução ao pensamento computacional. Este destaque é dado, pois entende-se que compreendem ações mais abrangentes e que apoiam de forma mais fundamentada o planejamento do docente e do gestor escolar interessado em abordar o tema.

ISTE/CSTA - Computational Thinking Leadership Toolkit

O documento *Computational Thinking Leadership Toolkit*, construído pela ISTE e CSTA com suporte da *National Science Foundation* (NFS) foi publicado em 2011. Tem por objetivo auxiliar docentes e gestores educacionais a compreenderem e se tornarem agentes de implantação do pensamento computacional de forma sistêmica na educação básica (k-12). Busca servir como modelo para impulsionar o desenvolvimento de uma geração de pensadores criativos, que saibam aproveitar o poder da computação para resolução de problemas.

O documento descreve porque o pensamento computacional é importante para os estudantes na atualidade e detalha os seguintes recursos para sua introdução no ambiente escolar: (i) definição operacional de pensamento computacional (ii) vocabulário dos termos chaves envolvidos (iii) modelo para uma mudança sistemática

⁸ <http://www.cs4hs.com/>

⁹ <http://www.cs.cmu.edu/~CompThink/>

¹⁰ <http://www.csprinciples.org/>

(iv) um guia para implementação das estratégias e (v) aspectos a serem discutidos com diferentes atores envolvidos.

A definição operacional, já apresentada na seção anterior, permite uma compreensão e discussão mais direcionadas sobre o tema. Na mesma linha, o vocabulário de termos chaves como Coleta de dados (*Data Collection*), Análise de Dados (*Data Analysis*), Representação de Dados (*Data Representation*), Decomposição de Problemas (*Problem Decomposition*), Abstração (*Abstraction*) e etc. auxilia a construção do diálogo entre os envolvidos. Para cada termo são apresentadas também formas de abordá-lo em cada faixa etária.

O modelo para uma mudança sistemática se apoia em quatro 4 passos, são eles 1-Guiar (*Lead*), 2-Construir (*Build*), 3-Conectar (*Connect*) e 4-Praticar (*Practice*). Guiar (1) diz respeito a assumir uma postura de liderança e de conduta para tomada de decisão para então Construir (2) a consciência de pensamento computacional entre os demais professores e líderes. Conectar (3) refere-se ao potencial interdisciplinar de ligar o pensamento computacional com iniciativas educacionais inovadoras que construirão valores e conhecimento, incluindo a preparação de materiais e recursos para cada faixa etária. Praticar (4) refere-se a realização de atividades de promoção introdução de pensamento computacional nas salas de aula acompanhadas.

Como estratégias de implementação o documento apresenta sugestões de atividades segmentadas por faixa etária, com objetivos e pontos que deverão ser desenvolvidos e abordados.

Computação Desplugada (Computer Science Unplugged)

A computação desplugada é um documento que visa abordar os fundamentos da computação de forma lúdica, sem o uso de computadores e sem detalhes técnicos em demasia (Vieira et al, 2013). Pode ser utilizada com estudantes desde o ensino fundamental até o ensino superior. Entre os objetivos está a popularização da Computação levando seus conceitos a um público mais amplo.

O documento é voltado ao professor e fornece exemplos de atividades que possibilitam relacionar conceitos de computação com problemas do cotidiano. Aborda conceitos de computação como números binários, representação de imagens, compressão de texto, detecção e correção de erros, algoritmos de ordenação, criptografia e outros. As atividades são descritas na forma de um plano de aula e são acompanhadas de materiais que podem ser impressos para auxiliar a sua realização.

Computação Criativa

O guia de computação criativa, desenvolvido pelo MIT busca criar o engajamento com a computação com base na criatividade, imaginação e interesses pessoais. Constitui um programa de introdução à computação criativa com uso do Scratch utilizando uma abordagem de aprendizagem baseada no conceito de design.

As atividades do guia são concebidas para explorar conceitos de pensamento computacional (sequência, ciclos, execução em paralelo, eventos, condições, operadores, dados), práticas (trabalhando de forma iterativa e incremental, testando, corrigindo e depurando, reutilizando e refazendo, abstraindo e modulando), e perspectivas (expressando, ligando, questionando).

Possui 20 sessões de 60 minutos distribuídas da seguinte maneira: Introdução (2), Artes (3), Histórias (3), Jogos(3) e Projeto Final (8). É voltado a professores que desejem introduzir a computação de forma interdisciplinar envolvendo conceitos de criatividade e arte.

Hora do Código (Code.org)

A hora do código, dentre os recursos detalhados nesta seção é o único voltado mais diretamente aos estudantes. Propõe um sequencia de atividades de introdução a programação com duração estimada de uma hora, voltada ao público infanto-juvenil. As atividades são esteticamente bem construídas, usam personagens de desenhos animados e usam a notação de programação com blocos das ferramentas scratch e blockly. Entre uma atividade e outra, vídeos motivacionais e explicativos com personalidades de destaque no mundo da tecnologia (como Bill Gates, Mark Zulkemberg e outros) são exibidos.

Em novembro de 2014 ocorreu a 2ª edição da Semana da Educação em Computação (*Computer Science Education Week*) que disseminou a utilização da hora do código em uma escala sem precedentes. Conforme (Hour of Code Stats, 2015) mais de 60 milhões de estudantes acessaram a atividade.

Além da Hora do Código o site Code.org oferece recursos para que professores e gestores se engajem da campanha de ampliar o ensino de computação nas escolas. Ainda que existam informações em outros idiomas, os dados e a motivação para o projeto são exclusivamente fundamentados na realidade do mercado e escolas dos Estados Unidos.

4. Por um Pensamento Computacional Construcionista

Ao analisar os recursos detalhados na seção anterior, percebe-se que diferentes perspectivas políticas e filosóficas sustentam as iniciativas. Enquanto que os recursos Computação Desplugada e Computação Criativa se aproximam de uma conotação de disseminar o conhecimento sobre Computação e a fluência no uso de tecnologia, os recursos da Hora do Código estão mais voltados ao ensino de habilidades de programação e técnicas computacionais de resolução de problemas. Já o material da CSTA transita em ambas as perspectivas, tendendo mais para a primeira.

Em especial a Hora do Código, que atingiu um número expressivo de estudantes, possui sua argumentação muito voltada necessidade de preencher vagas de trabalho na área de tecnologia e as perspectivas positivas para a carreira de programador. Considera-se que este enfoque não permite que o potencial do conceito de Pensamento Computacional seja explorado em sua plenitude.

Blikstein (2013) menciona que relatórios do NRC (*National Research Council*) nos Estados Unidos indicam que décadas já foram desperdiçadas ao ensinar os estudantes nesta abordagem. O foco deve estar em proporcionar a fluência no uso da tecnologia, possibilitando que o estudante crie habilidades fundamentais e adaptáveis que empoderem-no a manipular o meio a seu favor com uso da tecnologia. A habilidade de programar é apenas um componente neste processo.

A natureza do Pensamento Computacional e, principalmente, os benefícios de promover o desenvolvimento do Pensamento Computacional pelos estudantes se

aproximam do que foi defendido por Papert (2008) quando propôs a Linguagem Logo e fundamentou o Construcionismo.

Conforme Valente (1993) Papert denominou de construcionista a abordagem em que o aprendiz constrói, com uso do computador, o seu próprio conhecimento. Ele usou esse termo para mostrar um outro nível de construção do conhecimento: a construção do conhecimento que acontece quando o aluno constrói um objeto de seu interesse, como uma obra de arte, um relato de experiência ou um programa de computador.

Na noção de construcionismo de Papert existem duas ideias que contribuem para que esse tipo de construção do conhecimento seja diferente do construtivismo de Piaget. Primeiro, o aprendiz constrói alguma coisa, ou seja, é o aprendizado por meio do fazer, do "colocar a mão na massa". Segundo, o fato de o aprendiz estar construindo algo do seu interesse e para o qual ele está bastante motivado. O envolvimento afetivo torna a aprendizagem mais significativa (VALENTE, 1993).

Desta forma, defende-se neste artigo que o Pensamento Computacional terá maior amplitude e possibilitará promover mudanças mais profundas no desenvolvimento dos estudantes, e conseqüentemente maior impacto na sociedade, se estiver alinhado à abordagem construcionista.

Ao defender tal premissa, torna-se fundamental resgatar as aprendizagens decorrentes do maior empreendimento construcionista realizado no Brasil, quando da introdução da filosofia Logo nas escolas de educação básica. Porém, os trabalhos realizados com o Logo, por serem na sua maioria da década de 1980 e 1990, foram publicados em formatos não digitais. Sendo assim, a estratégia de revisão sistemática com base em mecanismos de busca não permitirá reconstruir esta história completamente. Cabe a realização de um empreendimento mais amplo envolvendo a recuperação de documentos textuais, a realização e entrevistas com pesquisadores e professores envolvidos a fim de concretizar este levantamento.

Entende-se que o olhar para a experiência do passado, em um momento que o contexto impulsiona para uma nova onda de uso de tecnologia com um novo rótulo (pensamento computacional), e em um período onde a tecnologia está mais presente e com mais potencialidades que na época do Logo, permitirá não apenas refletir sobre os acertos e erros, mas também vislumbrar novos cenários mais alinhados a realidade político educacional das escolas brasileiras.

5. Por um Plano de Ação

Não há a menor pretensão de propor soluções ou recomendações pontuais para o uso do pensamento computacional neste artigo, principalmente por entender-se que não é uma iniciativa que possa ser realizada por um pequeno grupo. É um tema que necessita ser discutido amplamente e neste sentido as iniciativas de ação propostas a seguir buscam criar espaços para discussão e enfatizam a importância de que sejam construídas diretrizes alinhadas a realidade brasileira.

- Criação de uma Comunidade de Discussão: um ponto de partida será organizar as pessoas interessadas, pesquisadores, professores, gestores educacionais para dialogarem sobre o assunto. Decorrente desta ação podem estar:
 - A criação de eventos temáticos por nível de ensino;
 - A articulação com comunidades científicas da área de Educação e de Educação em Computação;

- A criação de mecanismos de comunicação como listas e wikis;
 - A proposição de um fórum para discussão de alternativas para a educação integral.
- Definição de estratégias para impactar as escolas: retoma-se aqui o elemento que motivou esta reflexão. A possibilidade de as escolas de educação integral oferecerem atividades interdisciplinares que possam ser alinhadas as iniciativas do pensamento computacional. Neste sentido, estratégias de ação devem ser traçadas. A elaboração de documentos de orientação como os da CSTA/ISTE é certamente uma possibilidade. Em paralelo podem ser realizadas ações voltadas a estimular a criação de políticas educacionais e ações acadêmico científicas como as mencionadas a seguir.
 - Definição de uma Agenda de Pesquisa: As pesquisas mais frequentes na área são relatos de experiência de uso de tecnologias na educação básica, normalmente abordando atividades de programação e ou robótica. A partir da formação de uma comunidade de discussão a identificação de lacunas e a proposição de uma agenda de pesquisa para supri-las torna-se possível.
 - Produção e disseminação de material de orientação: já existe muito material de apoio ao pensamento computacional, muitos no idioma inglês, mas bons matérias em português. Orientações sobre onde encontra-los e como usálos podem ser construídas tendo como público alvo professores e gestores escolares. A criação de um repositório de boas práticas pode também colaborar na mesma direção

6. Considerações Finais

Para tornar a educação integral um avanço significativo de qualidade na formação dos estudantes é importante que se investiguem e discutam alternativas que explorem o potencial da interdisciplinaridade tal como defendido neste artigo, o desenvolvimento do pensamento computacional em uma perspectiva construcionista.

Certamente o pensamento computacional, ou mais amplamente o construcionismo são temas que podem impactar positivamente as escolas, mas não são os únicos. Cabe a comunidade de pesquisadores envolvidos com o assunto a definição de parâmetros que possam orientar e impulsionar sua adoção pelos ambientes educacionais. Teme-se que na ausência de alternativas e diretrizes, o provável é que as escolas reproduzam o que lhes é natural, ou seja, segmentem o tempo adicional em matérias e preencham-nas de conteúdos no formato tradicional que cada vez mais a dificultam a produção de significados pelo estudante.

7. Referencias

ANDRADE, D. et al. Proposta de Atividades para o Desenvolvimento do Pensamento Computacional no Ensino Fundamental. Anais do Workshop de Informática na Escola, Congresso Brasileiro de Informática na Educação, 2013.

BARCELOS, T. S., SILVEIRA, I. F. (2012). Pensamento Computacional e Educação Matemática: Relações para o Ensino de Computação na Educação Básica. In XX Workshop sobre Educação em Computação.

BLIKSTEIN, P. (2013a). Digital Fabrication and 'Making' in Education: The Democratization of Invention. In J. Walter-Herrmann & C. Büching (Eds.), FabLabs: Of Machines, Makers and Inventors. Bielefeld: Transcript Publishers. (To appear)

BRASIL, Plano Nacional de Educação 2014-2014, Disponível em: http://www.planalto.gov.br/CCIVIL_03/_Ato2011-2014/2014/Lei/L13005.htm, Acessado Abril de 2015.

CREATIVE COMPUTING. Disponível em: <http://scratched.gse.harvard.edu/guide/> Acessado: Junho de 2015.

CS UNPLUGGED. Disponível em: <http://csunplugged.org/> Acessado: Junho de 2015.

CSTA Computational Thinking Task Force, Computational Think Flyer. Disponível em: <http://csta.acm.org/Curriculum/sub/CurrFiles/CompThinkingFlyer.pdf>, Acessado Abril de 2015.

FRANÇA, Rozelma; AMARAL, Haroldo. Proposta Metodológica de Ensino e Avaliação para o Desenvolvimento do Pensamento Computacional com o Uso do Scratch. Anais do Workshop de Informática na Escola, Congresso Brasileiro de Informática na Educação, 2013.

HOUR OF CODE STATS. Disponível em: <https://hourofcode.com/us/resources/stats> Acessado: Junho de 2015.

NOSCHANG, L. F. ; DE JESUS, E. A. ; PELZ, Fillipi ; RAABE, André Luís Alice . Portugol Studio: Uma IDE para Iniciantes em Programação. In: Workshop sobre Educação em Informática, 2014, Brasília. Anais do Congresso Anual da Sociedade Brasileira de Computação. Porto Alegre: SBC, 2014. v. 1. p. 535-545.

PAPERT, S. Mindstorms : children, computers, and powerful ideas. New York: Basic Books, 1980.

PAPERT, Seymour. A máquina das crianças: Repensando a escola na era da Informática. Artes Médicas: Porto Alegre. 2008.

RESNICK, M., Maloney, J., Monroy-Hernández, A., Rusk, N., Eastmond, E., Brennan, K., . . . Silverman, B. Scratch: programming for all. Communications of the ACM, 52(11), 60-67. 2009.

THE ROYAL SOCIETY. Shut down or restart? The way forward for computing in UK schools. Technology, 2012.

VALENTE, José Armando. Diferentes usos do computador na educação. In:

Computadores e conhecimento: repensando a educação. 1ª ed. Campinas, NIED Unicamp, 1993.

VIEIRA, A.; PASSOS, O.; BARRETO, R. Relato de Experiência do Uso da Técnica Computação Desplugada. In: Workshop sobre Educação em Informática, 2013.

VIEL, Felipe ; RAABE, André Luís Alice ; ZEFERINO, Cesar Albenes . Introdução a Programação e à Implementação de Processadores por Estudantes do Ensino Médio. In: Workshop de Informática na Escola, 2014, Dourados. Anais do III Congresso Brasileiro de Informática na Educação. Porto Alegre: SBC, 2014. v. 1. p. 248-257.

WING, J. Computational thinking. Communications of ACM, v. 49, n. 3, p. 33-36, 2006.