

## Desafios e oportunidades ao ensino do pensamento computacional na educação básica no Brasil

Rozelma Soares de França<sup>1</sup>, Patrícia Cabral de Azevedo Restelli Tesdeco<sup>1</sup>

<sup>1</sup>Centro de Informática – Universidade Federal de Pernambuco (UFPE)  
Recife – PE – Brasil

{rsf2, pcart}@cin.ufpe.br

**Resumo.** Atualmente é requerido dos estudantes desenvolver diversas habilidades, dentre elas o pensamento computacional. Contudo, no Brasil o ensino de tal habilidade não integra o currículo escolar. Nesse sentido, este artigo discute desafios ao ensino do pensamento computacional na educação básica brasileira, apresentando oportunidades de pesquisa na área. Além disso, é apresentada uma proposta para minimizar alguns dos problemas apontados, a qual demonstrou contribuir com a aprendizagem de estudantes do nível médio durante um curso de desenvolvimento de jogos digitais.

### 1. Introdução

Com o rápido crescimento computacional e tecnológico ocorrido no mundo moderno surge a necessidade de ensinar, desde a educação básica, conceitos fundamentais da Ciência da Computação como forma de melhorar o aprendizado escolar dos indivíduos e possibilitar o uso mais eficaz dessas tecnologias em benefício da sociedade. O *ACM Model Curriculum for K-12 Computer Science* (CSTA, 2011) enfatiza tal necessidade, destacando que o ensino de Computação na escola pode desenvolver a capacidade de resolver problemas pelos estudantes, além de apoiar e relacionar-se com outras ciências.

Ao falar-se de Computação enquanto saber necessário na educação básica deve-se ressaltar que seu ensino passa por noções de modelos computacionais, algoritmos, complexidade, autômatos, entre outros conteúdos (NUNES, 2008). Assim, o ensino à manipulação de *softwares*, tais como editores de texto, não é suficiente na atualidade. A habilidade em questão tem sido chamada de *pensamento computacional* (WING, 2006) e deve integrar a formação básica dos cidadãos do século XXI possibilitando que futuros jornalistas, músicos, advogados, dentre outros, resolvam problemas de suas respectivas áreas de atuação.

Percebendo a relevância do tema, os Estados Unidos e países da Europa têm buscado promover o ensino de Computação em suas escolas (CSTA, 2005). No contexto brasileiro, no entanto, tal disciplina é ofertada apenas em cursos técnicos, de graduação e de pós-graduação na área, sendo observadas algumas iniciativas envolvendo estudantes dos níveis fundamental e médio.

Ensinar habilidades computacionais na educação básica no Brasil pode, portanto, configurar-se um desafio e apresentar-se como um cenário repleto de oportunidades aos educadores, pesquisadores e comunidade escolar. Nesse cenário, este trabalho discute desafios enfrentados no ensino do pensamento computacional na educação básica brasileira apontando para oportunidades de pesquisa na área. Além disso, é apresentada uma proposta passível de ser integrada na sala de aula que promove

a aprendizagem do pensamento computacional pela prática de programação no contexto do ensino médio.

O restante do artigo está organizado como segue: na seção 2 são apresentadas experiências de ensino do pensamento computacional na educação básica brasileira e na seção 3 são apontados desafios ao ensino de tal habilidade, considerando-se o atual currículo escolar. Ainda, na seção 4 é apresentado um caminho para superar alguns dos desafios descritos e, por fim, na seção 5 são feitas algumas considerações finais.

## **2. Pensamento computacional em contextos escolares brasileiros**

O interesse pela disseminação do pensamento computacional no ambiente escolar é crescente tendo envolvido a participação de pesquisadores e educadores de diferentes países, incluindo o Brasil.

No cenário nacional, Scaico *et al.* (2013) relatam a experiência de uma olimpíada realizada na Paraíba que objetivou apresentar aos estudantes as principais estruturas de uma linguagem de programação e praticar técnicas utilizadas na construção de algoritmos, como é o caso do uso da abstração, depuração de erros, testes e melhoria de algoritmos simples.

Em Pernambuco, França *et al.* (2012) e França *et al.* (2014) relatam experiências vivenciadas por licenciandos em Computação na busca pela disseminação do pensamento computacional na escola, bem como na divulgação dos cursos de Licenciatura em Computação e na desmistificação do papel desses profissionais na sociedade. Já Gomes e Melo (2013) descrevem um curso de lógica de programação ofertado a estudantes do ensino médio no qual foi utilizado o ambiente de programação visual *App Inventor* ([www.appinventor.mit.edu](http://www.appinventor.mit.edu)) para criar aplicativos para dispositivos Android.

Na Bahia, Machado *et al.* (2010) discorrem sobre o desenvolvimento de um projeto que visou incentivar e descobrir jovens talentos para a área de Computação em escolas do ensino fundamental e médio. Também era objetivo do trabalho incentivar a participação feminina no projeto e dos estudantes na Olimpíada Brasileira de Informática (OBI). Segundo os autores, a pesquisa teve como base trabalhar nos educandos a capacidade de raciocínio lógico-matemático e introduzir o raciocínio algorítmico, de maneira que os estudantes pudessem compreender a utilização destes modelos de raciocínio na solução de problemas.

No Rio Grande do Sul, Andrade *et al.* (2013) propuseram atividades direcionadas ao ensino fundamental que envolvem os princípios do pensamento computacional. Já Vieira *et al.* (2014) realizaram peças teatrais no Amazonas possibilitando aos participantes compreender os fundamentos da Computação de forma clara e divertida. Por outro lado, de Souza *et al.* (2014) têm utilizado o *AgentSheets* ([www.agentsheets.com](http://www.agentsheets.com)) em suas pesquisas em escolas do Rio de Janeiro possibilitando que os educandos expressem seus interesses e capacidades de comunicação por meio da programação de jogos e simulações.

Já no trabalho de Barcelos e Silveira (2013) os autores investigaram, em uma instituição de São Paulo, de que forma as competências relacionadas à Matemática são mobilizadas por estudantes do ensino médio no processo de desenvolvimento do pensamento computacional através da construção de jogos digitais. No mesmo Estado,

Garcia *et al.* (2008) apresentam a experiência de execução de um projeto que visou proporcionar a estudantes do ensino médio a oportunidade de adquirir competências relacionadas ao desenvolvimento do raciocínio lógico, especificamente na resolução de problemas por meio de algoritmos e estruturas de dados; e, como efeito secundário, desejava-se motivar os participantes a continuar seus estudos na área de Computação. Os resultados apontam para o interesse dos estudantes nos conteúdos do curso, havendo ainda participação deles em atividades da OBI e o ingresso em cursos superiores da área de Computação.

No trabalho de Carvalho *et al.* (2013) os autores apresentam uma iniciativa para a inserção do ensino do pensamento computacional e de conceitos básicos de tecnologia da informação no ensino médio de escolas de Minas Gerais. O projeto, voltado à empregabilidade e que envolve diversas áreas incluindo Tecnologia da Informação, foi implementado em 2012 em onze escolas da zona norte de Belo Horizonte, abrangendo um total de 5.979 alunos. A expectativa era que em 2014 todas as 2.167 escolas estaduais fizessem parte do projeto, alcançando os 678.684 estudantes do ensino médio estadual.

### **3. Oportunidades e desafios para o Brasil**

As atividades desenvolvidas, descritas na seção anterior, são, em sua maioria, decorrentes de projetos de pesquisa, ensino e extensão executados por graduandos e pós-graduandos de cursos de Computação. Considerando-se os benefícios que o pensamento computacional pode trazer para a formação dos estudantes da atualidade, discutem-se alguns desafios a serem enfrentados na busca pelo seu ensino na educação básica no Brasil.

#### **3.1. Currículo escolar**

Uma das principais questões a serem consideradas no ensino do pensamento computacional é o currículo escolar. Nesse contexto, enxergam-se duas possibilidades que podem ser abordadas em futuras pesquisas. A primeira diz respeito à criação de uma disciplina obrigatória, assim como Matemática, que trate dos fundamentos da Computação nos diferentes anos da educação básica. Nessa conjectura, deve-se atentar para diversos aspectos, dentre eles, os possíveis efeitos da introdução de uma nova disciplina no currículo escolar na motivação e na aprendizagem dos estudantes.

A segunda possibilidade tem uma perspectiva interdisciplinar onde o pensamento computacional é trabalhado atrelado às disciplinas já existentes no currículo escolar. Nesse contexto, a promoção da aprendizagem de conceitos computacionais é feita aliada à construção de conhecimento de conteúdos curriculares dos diferentes anos da educação básica.

Além disso, outra questão que deve ser considerada diz respeito ao quê e quando ensinar o pensamento computacional na educação básica. Nesse contexto, é preciso definir diretrizes curriculares para o ensino de tal habilidade nas escolas brasileiras, enfatizando que conceitos devem ser introduzidos e em quais anos escolares. Isto deve levar em consideração que o ensino do pensamento computacional não deve cobrir apenas a manipulação de recursos digitais, mas sim os fundamentos da Computação, enquanto ciência.

### 3.2. Formação de professores

O ensino do pensamento computacional na educação básica brasileira irá requerer a formação de professores especializados para atuarem nas escolas. Nesse contexto, segundo Nunes (2010), os cursos de Licenciatura em Computação têm uma enorme responsabilidade de formar professores para introduzir a Ciência da Computação na educação básica, disseminando assim o pensamento computacional.

Há de se considerar que tais cursos não devem apenas concentrar suas atividades na construção de *softwares* educativos e no uso de tecnologias aplicadas à educação, mas preparar os futuros licenciados em Computação para que atuem no ensino do pensamento computacional. Isto irá requerer que os cursos ofertados pelas instituições de Educação possibilitem a tais profissionais uma sólida formação em Ciência da Computação, Matemática e Educação, como definido nas Diretrizes Curriculares Nacionais para os cursos de graduação em Computação<sup>1</sup>.

Ademais, acredita-se que os outros cursos de licenciatura também poderiam beneficiar-se de uma formação em pensamento computacional. Isto poderia prover aos futuros professores de Matemática, Biologia, e outros, uma formação em Computação que os ajudaria na resolução de problemas de suas respectivas áreas de atuação. Além disso, o desenvolvimento de ações educativas envolvendo o pensamento computacional de modo interdisciplinar também poderia ser favorecido nas escolas brasileiras.

### 3.3 Os processos de ensinar e aprender

Tem-se observado que diferentes abordagens de ensino têm sido empregadas na educação básica vislumbrando-se o ensino do pensamento computacional. Quando se tem como objetivo promover essa habilidade por meio da lógica de programação, por exemplo, é recorrente o uso de ambientes visuais de programação tais como *Scratch* ([www.scratch.mit.edu/](http://www.scratch.mit.edu/)), *App Inventor* ([www.appinventor.mit.edu/](http://www.appinventor.mit.edu/)), *AgentSheets* ([www.agentsheets.com/](http://www.agentsheets.com/)) e *RoboMind* ([www.robomind.net/pt](http://www.robomind.net/pt)). Como produtos dessas atividades tem-se, por exemplo, jogos, simulações e animações, criados a partir dos interesses dos estudantes ou a partir de recomendações dos instrutores dos cursos.

Ainda constata-se que os princípios da Computação podem ser ensinados aos estudantes sem depender do uso de uma tecnologia específica. Nesse contexto, destaca-se o trabalho de Bell *et al.* (2011) que propuseram um conjunto de atividades lúdicas envolvendo fundamentos da Ciência da Computação, publicadas no livro *Computer Science Unplugged*. Tais atividades estimulam o pensamento computacional, sem o uso do computador, e têm despertado o interesse de pesquisadores e professores e empregadas em diversos países, entre eles o Brasil.

Tendo em vista que os processos de ensinar e aprender perpassa a concepção, o desenvolvimento e a avaliação da experiência desenvolvida, acredita-se que a participação ativa dos estudantes nessas atividades pode trazer bons resultados de aprendizagem. Isto irá requerer momentos de reflexão por partes de educadores e educandos, que poderão (re)pensar suas práticas e melhorar suas estratégias de ensino e

---

<sup>1</sup> Parecer CNE/CES nº 136/2012, aprovado em 8 de março de 2012 - Diretrizes Curriculares Nacionais para os cursos de graduação em Computação. Disponível em <[http://portal.mec.gov.br/index.php?option=com\\_content&id=12991:diretrizes-curriculares-cursos-de-graduacao](http://portal.mec.gov.br/index.php?option=com_content&id=12991:diretrizes-curriculares-cursos-de-graduacao)>. Acesso em 02 de agosto de 2015.

aprendizagem. Acredita-se, ainda, que a aprendizagem pode ser significativa quando os interesses dos estudantes são considerados, o que inclui sua participação na definição de objetivos e avaliação da aprendizagem.

Assim, em atividades que busquem promover a aprendizagem do pensamento computacional, diversos fatores devem ser considerados, dentre eles os interesses dos aprendizes o que resulta em um desafio que educadores e pesquisadores devem se preocupar: como ensinar conceitos fundamentais da Computação considerando o público-alvo e suas peculiaridades provendo mecanismos para uma aprendizagem significativa?

### **3.4 Aprendizagem de Computação**

A Educação em Computação é um desafio, especialmente o ensino introdutório de programação (CASPERSEN; KÖLLING, 2009) e pesquisas apontam que são várias as dificuldades enfrentadas pelos estudantes ao iniciarem seus estudos na área. Dentre os problemas apontados podem-se citar métodos tradicionais de ensino que não consideram as particularidades de aprendizado dos estudantes, como também a forma inadequada de estudo tida pelos iniciantes (GOMES; MENDES, 2007).

Por outro lado, pesquisas também evidenciam que há fatores que podem contribuir para o sucesso na aprendizagem de Computação. Na pesquisa de Byrne e Lyons (2001) os resultados sugerem que possuir experiência prévia em programação pode favorecer o aprendizado dos estudantes na disciplina. Outros trabalhos mostram que a habilidade matemática é um importante indicativo de bom desempenho obtido em disciplinas introdutórias nos cursos de Ciência da Computação (WILSON; SHROCK, 2001).

Nesse contexto, é importante conhecer a literatura sobre ensino de Computação, especificamente os fatores que podem contribuir para o sucesso ou insucesso na aprendizagem, e considerar que estudantes da educação básica poderão também enfrentar situações similares. Nesse sentido, lições aprendidas e apresentadas por pesquisadores e educadores poderão orientar experiências de ensino de Computação a serem vivenciadas nos níveis fundamental e médio no Brasil.

## **4. Um caminho para vencer alguns dos desafios apontados**

Tendo em vista os desafios relacionados aos processos de ensino e aprendizagem de Computação, uma abordagem é apresentada para minimizar tais problemas. Trata-se de um modelo para a aprendizagem do pensamento computacional, intitulado penC<sup>2</sup> (FRANÇA, 2015).

### **4.1 O modelo penC**

Sua concepção baseou-se em estudos sobre autorregulação da aprendizagem (ZIMMERMAN, 2002), metacognição (GAMA, 2004; TOBIAS; EVERSON, 2002), ensino de Ciência da Computação (LYE; KOH, 2014; COMPUTING AT SCHOOL WORKING GROUP, 2011; CSTA, 2011; ISTE *et al.*, 2011; NAS, 2010; WING, 2006; SITTHIWORACHART *et al.*, 2004) e aprendizagem no nível médio (LONG;

---

<sup>2</sup> Pronuncia-se 'pense'

ALEVEN, 2013). Como forma de mitigar problemas na aprendizagem de conceitos computacionais, recorreu-se a fundamentos da autorregulação da aprendizagem, escolhendo-se as estratégias de *autoavaliação* e *avaliação por pares* para serem integradas ao modelo.

O penC tem a intenção de criar condições adequadas para que estudantes do ensino médio desenvolvam habilidades e competências requeridas na atualidade pensando sobre si mesmos (consciência metacognitiva) como solucionadores de problemas e refletindo sobre suas experiências contínuas de aprendizagem. O modelo é constituído de quatro fases, a saber: Pré-Reflexão, Resolução, Avaliação por Pares e Pós-Reflexão (Figura 1). Elas são instanciadas enquanto os estudantes resolvem cada novo problema de Lógica de Programação.

A primeira fase é a *pré-reflexão*, que antecede a resolução de um problema e é composta de duas atividades. Com o objetivo de estimular a reflexão dos estudantes sobre seu processo de aprendizagem, a primeira atividade dessa fase apresenta o estado atual do aprendiz com relação aos resultados de aprendizagem que devem ser alcançados em um curso introdutório de programação. A segunda atividade tem como objetivo apoiar a reflexão dos educandos sobre o problema a ser resolvido. Para tanto, eles devem autoavaliar sua compreensão e dificuldade de resolver o problema proposto.



**Figura 1. Fases do modelo penC**

Na segunda fase, chamada de *resolução*, os estudantes resolvem o problema seguindo padrões de codificação. Para auxiliá-los nessa tarefa, há *scaffolds* que podem ajudar os aprendizes a refletir sobre o problema corrente com base em sua experiência anterior de resolução de problemas de programação. No final dessa fase, o estudante envia sua solução antes do término do prazo definido pelo professor que pode monitorar aqueles com dificuldades na resolução do exercício.

A *avaliação por pares* é a terceira fase do modelo penC. Nesta fase, as soluções apresentadas pelos estudantes são avaliadas por seus pares. No início desta fase, as soluções são atribuídas a, pelo menos, outros três estudantes, chamados de avaliadores. Antes de iniciar a avaliação cada um dos avaliadores deve informar se compreendeu os critérios de análise da solução. Esta atividade é importante, pois um entendimento impreciso dos critérios poderá interferir na forma de avaliação e, conseqüentemente, repercutir na aprendizagem dos estudantes que receberão os comentários. O outro

aspecto é que o professor é notificado desses estudantes-avaliadores que não entenderam os critérios e pode guiá-los em sua avaliação.

Para auxiliar os avaliadores na análise das soluções *scaffolds* foram projetados objetivando auxiliar os estudantes no fornecimento de *feedbacks* que evidenciem os pontos fortes da solução, que apontem os erros e sugiram melhorias ajudando, assim, os criadores das soluções a corrigirem seus erros, quando houver, ou a repetirem boas práticas de programação em suas atividades enquanto solucionadores de problemas da área.

A quarta fase do penC é a *pós-reflexão*, que visa envolver os estudantes na reflexão sobre a resolução de problemas de programação. No início desta fase, os educandos podem ver o *feedback* dado à sua solução na fase anterior. Depois, também podem avaliar a qualidade do *feedback* recebido e optar por compartilhar sua solução com os demais estudantes e receber novos comentários, um processo que estimula uma nova discussão sobre a solução compartilhada.

Ainda na última fase, os estudantes podem monitorar seu processo de aprendizagem do pensamento computacional através de diferentes atividades, baseadas na precisão no monitoramento do conhecimento (*Knowledge Monitoring Accuracy – KMA*) definido por Tobias e Everson (2002) e no viés no monitoramento do conhecimento (*Knowledge Monitoring Bias – KMB*) definido por Gama (2004).

## 4.2 Uso do penC em um curso de construção de jogos digitais

Como forma identificar o impacto do modelo penC na autorregulação e na aprendizagem do pensamento computacional, um quasi-experimento foi realizado em um curso de desenvolvimento de jogos digitais que introduziu conceitos de lógica de programação em iniciantes na área.

Nas atividades propostas em tal curso era possível aos estudantes construir jogos de acordo com seus interesses pessoais, tendo, no entanto, que atender a requisitos mínimos, envolvendo conteúdos de lógica de programação. Buscou-se, com isso, manter o engajamento dos educandos nas atividades, ao mesmo tempo em que aprendiam conceitos norteadores do pensamento computacional. Na Figura 2 são exibidos alguns jogos digitais desenvolvidos pelos estudantes com o uso do *Stencyl* ([www.stencyl.com](http://www.stencyl.com)).

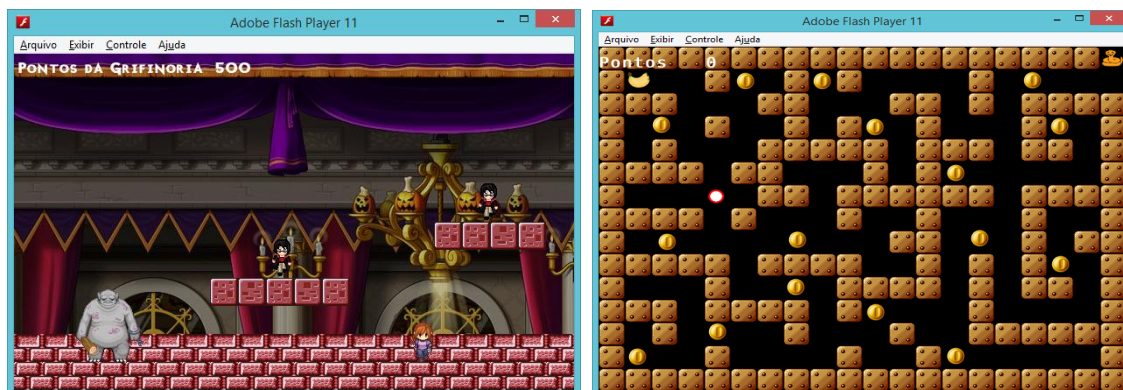


Figura 2. Exemplos de jogos digitais desenvolvidos pelos estudantes

Durante o referido curso, dois grupos foram formados: 1) grupo experimental e 2) grupo controle. No grupo experimental o processo de desenvolvimento de jogos digitais ocorreu considerando-se as fases do penC. Assim, à medida que atividades de construção de *games* eram propostas, envolvendo conceitos de lógica de programação, os estudantes se envolviam em atividades de reflexão antes e após desenvolver o jogo e ainda avaliavam a produção dos colegas. No grupo controle, no entanto, o professor executava a tarefa de avaliação de jogos e não havia atividades de pré e pós-reflexão. Assim, a abordagem adotada por tal grupo foi nomeada como tradicional.

A partir de tal experimento foi possível observar resultados positivos da abordagem proposta, frente à tradicional. Em relação à aprendizagem do pensamento computacional, o modelo penC trouxe contribuições na formação dos estudantes, sendo observadas melhorias em seu desempenho em habilidades de depuração e criação de códigos, consideradas importantes para os programadores. Na variável autorregulação da aprendizagem também se constatou efeito positivo no grupo experimental, sugerindo que o penC pode contribuir para o desenvolvimento de tal competência. Complementarmente foram observados efeitos positivos sobre a jogabilidade dos *games* produzidos pelos estudantes. Também, o modelo proposto foi avaliado satisfatoriamente pelo professor que percebeu ganhos na aprendizagem de seus estudantes com o uso do penC.

## 5. Considerações finais

O pensamento computacional é uma das habilidades a serem desenvolvidas pelos estudantes do século XXI. Contudo, seu ensino ainda não integra o currículo escolar brasileiro, resultando em diversos desafios a serem enfrentados por pesquisadores e comunidade escolar.

Nesse contexto, este trabalho apresentou iniciativas realizadas em diversos Estados, enfatizando desafios e oportunidades para o ensino do pensamento computacional no Brasil. Frente aos desafios discutidos foi apresentada uma proposta que pode ajudar a superar problemas identificados, ajudando estudantes da educação básica em sua formação em conceitos computacionais. O quasi-experimento descrito foi realizado no contexto de um curso de desenvolvimento de jogos digitais. Contudo, isto não limita a utilização do penC em intervenções didáticas que tratem o pensamento computacional de modo interdisciplinar.

Espera-se que as questões aqui expostas corroborem para a educação em Computação no cenário brasileiro, provendo *insights* para realização de futuras pesquisas sobre a disseminação do pensamento computacional nos níveis fundamental e médio.

## 6. Agradecimentos

Rozelma Soares de França agradece à CAPES pela concessão de bolsa de mestrado no período de realização desta pesquisa.

## Referências

ANDRADE, Daiane *et al.* Proposta de Atividades para o Desenvolvimento do Pensamento Computacional no Ensino Fundamental. In: **Anais do XVI Workshop de Informática na Escola**. SBC, 2013.



- BARCELOS, T. S.; SILVEIRA, I. F. Relações entre o pensamento computacional e a matemática através da construção de jogos digitais. In: **Anais do XII Simpósio Brasileiro de Jogos e Entretenimento Digital**, 2013.
- BELL, T.; WITTEN, I, H.; FELLOWS, M. **Computer Science Unplugged: Ensinando Ciência da Computação sem o uso do computador**. Tradução coord. por Luciano Porto Barreto, 2011.
- BYRNE, P.; LYONS, G.. The effect of student attributes on success in programming. In: **Proceedings of the SIGCSE**. ACM, 2001. p. 49-52.
- CARVALHO, M. L. B.; CHAIMOWICZ, L.; MORO, M. M. Pensamento Computacional no Ensino Médio Mineiro. In: **Anais do XXI Workshop sobre Educação em Computação**. 2013.
- CASPERSEN, M. E.; KOLLING, M.. STREAM: A first programming process. **ACM Transactions on Computing Education (TOCE)**, v. 9, n. 1, p. 4, 2009.
- COMPUTING AT SCHOOL WORKING GROUP. **Computing: A curriculum for schools**. Dezembro, 2011.
- CSTA - Computer Science Teacher Association. **CSTA K-12 Computer Science Standards**. CSTA Standards Task Force. ACM - Association for Computing Machinery, 2011.
- CSTA - Computer Science Teacher Association. **The New Educational Imperative: Improving High School Computer Science Education**. Final Report of the CSTA. Curriculum Improvement Task Force. ACM - Association for Computing Machinery, 2005.
- DE SOUZA, Clarisse S. *et al.* Cultural appropriation of computational thinking acquisition research: seeding fields of diversity. In: **Proceedings of the 19th Conference on Innovation & Technology in Computer Science Education**. ACM, 2014. p. 117-122.
- FRANÇA, R. S. ; FERREIRA, V. F. S.; ALMEIDA, L. C. F. ; AMARAL, H. J. C. A disseminação do pensamento computacional na educação básica: lições aprendidas com experiências de licenciandos em computação. In: **Anais do XXII Workshop sobre Educação em Computação**. SBC, 2014.
- FRANÇA, R. S. **Um modelo para a aprendizagem do pensamento computacional aliado à autorregulação**. 2015. Dissertação (Mestrado em Ciência da Computação) – Centro de Informática, Universidade Federal de Pernambuco, Recife. 2015.
- FRANÇA, R. S; SILVA, W. C.; AMARAL, H. J. C. do. Ensino de ciência da computação na educação básica: Experiências, desafios e possibilidades. In: **Anais do XX Workshop sobre Educação em Computação**. SBC, 2012.
- GAMA, C. A. . **Integrating metacognition instruction in interactive learning environments**. 2004. Tese (Doutorado). University of Sussex. 2004.
- GARCIA, R. E.; CORREIA, R. C. M.; SHIMABUKURO, M. H.. Ensino de Lógica de Programação e Estruturas de Dados para Alunos do Ensino Médio. In: **Anais do XVI Workshop sobre Ensino em Computação**. SBC, 2008. p. 246-249.

- GOMES, A.; MENDES, A. J.. Learning to program-difficulties and solutions. In: **International Conference on Engineering Education**. 2007.
- GOMES, T. C. S.; MELO, J. C. B. O Pensamento Computacional no Ensino Médio: Uma Abordagem Blended Learning In: **Anais do XXI Workshop sobre Educação em Computação**. SBC, 2013.
- ISTE - International Society for Technology in Education; CSTA - Computer Science Teachers Association; NSF - National Science Foundation. **Computational thinking: leadership toolkit**. First Edition, 2011.
- LONG, Y.; ALEVEN, V.. Active Learners: Redesigning an Intelligent Tutoring System to Support Self-regulated Learning. In: **Scaling up Learning for Sustained Impact**. Springer Berlin Heidelberg, 2013. p. 490-495.
- 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.
- MACHADO, E. Z. A. *et al.* Uma Experiência em Escolas de Ensino Médio e Fundamental para a Descoberta de Jovens Talentos em Computação. In: **Anais do XVIII Workshop sobre Educação em Computação**. SBC, 2010.
- NAS - National Academy of Sciences. **Report of a Workshop on The Scope and Nature of Computational Thinking**, 2010.
- NUNES, D. J.. Computação ou Informática? **Jornal da Ciência**. 30 de março, 2010.
- NUNES, D. J.. Licenciatura em Computação. **Jornal da Ciência**. 30 de maio, 2008.
- SCAICO, P. D. *et al.* Ensino de programação no ensino médio: uma abordagem orientada ao design com a linguagem Scratch. **Revista Brasileira de Informática na Educação**, v. 21, n. 02, p. 92, 2013.
- SITTHIWORACHART, J.; JOY, M.. Effective peer assessment for learning computer programming. In: **Proceedings of the ACM Technical Symposium on Computer Science Education**. ACM, 2004. p. 122-126.
- TOBIAS, S.; EVERSON, H. T. **Knowing what you know and what you don't: Further research on metacognitive knowledge monitoring**. 2002.
- VIEIRA, A.; PASSOS, O.; BARRETO, R.. Um Relato de Experiência do Uso da Técnica Computação Desplugada. In: **Anais do XXI Workshop sobre Educação em Computação**. SBC, 2013. p. 670-679.
- WILSON, B. C.; SHROCK, S.. Contributing to success in an introductory computer science course: a study of twelve factors. In: **Proceedings of the SIGCSE**. ACM, 2001. p. 184-188.
- WING, Jeannette M. Computational thinking. **Communications of the ACM**, v. 49, n. 3, p. 33-35, 2006.
- ZIMMERMAN, B. J.. Becoming a self - regulated learner: An overview. **Theory into practice**, v. 41, n. 2, p. 64-70, 2002.