

Pensamento Computacional no Ensino de Programação: Uma Revisão Sistemática da Literatura Brasileira

Humberto Augusto Piovesana Zanetti¹, Marcos Augusto Francisco Borges¹,
Ivan Luiz Marques Ricarte¹

¹Faculdade de Tecnologia – Universidade Estadual de Campinas (UNICAMP)
Limeira – SP – Brasil

h016304@dac.unicamp.br, {marcosborges, ricarte}@ft.unicamp.br

Abstract. *This paper aims to present a systematic review of the literature of research related to Computational Thinking, published in Brazilian events between 2012 and 2015. This review included papers published in three major national events, namely the Brazilian Symposium of Informatics in Education (SBIE), Computer Workshop School (WIE) and Workshop on Education in Informatics (WEI). There are also papers published in the Workshop Teaching on Computational Thinking, Algorithms and Programming (WAlgProg), as to keep up with the latest research on this subject. We analyzed 16 papers, in order to present a critical view of this emerging research area and define the lines of action of Brazilian researchers.*

Resumo. Este artigo tem como objetivo apresentar uma revisão sistemática da literatura dos trabalhos relacionados ao Pensamento Computacional, publicados em eventos brasileiros de 2012 até o ano de 2015. Esta revisão foi realizada a partir de artigos publicados nos três importantes eventos nacionais: Simpósio Brasileiro de Informática na Educação (SBIE), Workshop de Informática na Escola (WIE) e Workshop sobre Educação em Informática (WEI). Também foram utilizados trabalhos publicados no evento Workshop de Ensino em Pensamento Computacional, Algoritmos e Programação (WAlgProg), a fim de investigar quais são as pesquisas mais recentes. Foram analisados 16 artigos, com o objetivo de apresentar uma visão crítica desta área de pesquisa emergente e definir quais as linhas de atuação dos pesquisadores no Brasil.

1. Introdução

O ensino de programação de computadores não é mais uma prática exclusiva em cursos da área da Computação e de Engenharias. Em um mercado competitivo como o que temos atualmente, engenheiros, economistas e cientistas, por exemplo, devem saber utilizar ferramentas e métodos para criar modelos computacionais [Blikstein 2008]. Wing (2006) afirma que o Pensamento Computacional (PC) deve ser uma habilidade básica a ser ensinada às crianças, assim como ler, escrever e aritmética. É possível encontrar práticas de ensino de programação dentro das escolas do ensino fundamental e médio, sendo aplicadas como atividade extracurricular. Essas práticas podem desenvolver habilidades para auxiliar os alunos na resolução de problemas, ser um elemento ativo na construção do conhecimento e entender o contexto tecnológico em que vivemos.

PC, segundo Wing (2006), é um método para solucionar problemas, conceber sistemas e compreender o comportamento humano inspirado em conceitos da Ciência da

Computação. Blikstein (2008) afirma que PC é saber usar o computador como instrumento do poder cognitivo e operacional humano, afim de aumentar nossa produtividade, inventividade e criatividade. O trabalho de Mestre et al. (2015) justifica que as habilidades estimuladas pelo PC estão diretamente relacionadas à resolução de problemas, pois envolvem a capacidade de compreender as situações propostas e criar soluções através de modelos matemáticos, científicos ou sociais.

O desenvolvimento do PC em alunos por meio do ensino de programação demonstra resultados positivos. Brennan e Resnick (2012) mostram que uma abordagem baseada em PC pode ser utilizada no aprendizado de programação, principalmente para jovens e usando ferramentas que motivem e estimulem a criatividade. Para Wing (2006), PC é baseado em abstração e decomposição de problemas ou sistemas complexos, habilidades altamente recomendadas para alunos de programação.

Este artigo tem como objetivo principal apresentar uma visão sobre a pesquisa relacionada a PC no Brasil nos últimos anos. Para isso, foram considerados somente estudos direcionados às práticas educacionais que estimulem ou envolvam o PC, com o objetivo de ensinar programação de computadores, ou que utilizem métodos e ferramentas que usem conceitos de programação. Este trabalho é parte de uma pesquisa mais ampla, que investiga a abrangência na adoção do PC no ensino de programação, e visa contribuir para pesquisas futuras, mostrando onde os esforços da comunidade estão focados nessa linha de pesquisa.

O presente trabalho apresenta resultados de uma Revisão Sistemática da Literatura (RSL), assim como o processo de desenvolvimento, mostrando os protocolos da pesquisa até chegar aos resultados finais. A Seção 2 descreve o contexto da pesquisa e a Seção 3 explica os métodos utilizados na revisão sistemática. A Seção 4 apresenta a síntese dos resultados obtidos e discussões sobre os resultados. Por fim, a Seção 5 apresenta as conclusões da pesquisa.

2. Contexto da Pesquisa

As pesquisas relacionadas ao PC vêm crescendo, tanto no Brasil como no exterior. Desde a criação do termo “*Computational Thinking*” (Pensamento Computacional, em inglês) pela pesquisadora Jeannette Wing em 2006, vários pesquisadores cujo foco da pesquisa é o ensino de programação, veem o PC como uma estratégia de grande valia. Trabalhos como os de Cuny, Snyder e Wing (2010), Barr e Stephenson (2011) e Brennan e Resnick (2012) mostram a eficácia da adoção do PC como método de sistematizar o pensamento na formulação de soluções, de maneira incremental e recursiva.

Pensar computacionalmente não é sinônimo de “programar”. No entanto, Blikstein (2008) aponta que umas das etapas fundamentais do PC é saber programar um computador para realizar tarefas cognitivas e de maneira automatiza, para que este conhecimento seja um suporte ao raciocínio humano. De Paula, Valente e Burn (2014) afirmam que o PC é uma maneira específica de se pensar e de analisar uma situação ou um artefato, sendo independente do uso de tecnologia, mas argumentam que as ideias teóricas, que são a base do conhecimento, podem ser aliadas aos aspectos práticos da programação. Nesse contexto, se faz relevante uma revisão sistemática que possa indicar como a pesquisa brasileira avança no uso de PC no ensino de programação.

2.1. Questões da Pesquisa

O objetivo desta RSL é identificar e avaliar os estudos publicados nos principais eventos e revistas brasileiras associadas à Sociedade Brasileira de Computação (SBC), que apresentem alguma contribuição no ensino de programação de computadores e PC. Com esse objetivo em vista, pretende-se responder à questão principal desta pesquisa: “*O que se tem feito nas pesquisas realizadas no Brasil sobre Pensamento Computacional no contexto de ensino de programação?*”.

Com base na questão principal apresentada, outras três questões de pesquisa (QP) mais específicas foram definidas:

- QP1: *Quais são as práticas pedagógicas mais realizadas?*
- QP2: *Em quais níveis de escolaridade as pesquisas estão sendo feitas?*
- QP3: *Quais são os objetivos a serem alcançados com essas práticas?*

A QP1 serve para mapear quais são as práticas (ações) mais comuns realizadas em pesquisas no Brasil, mostrando também quais são os principais instrumentos (tecnologias e ferramentas) utilizados. Na QP2, o objetivo é identificar quais são os níveis de escolaridade (ensino fundamental, médio ou superior) nos quais estão ocorrendo as pesquisas, traçando quais são as tendências em cada nível. Por fim, a QP3 delimita quais são os objetivos definidos nas práticas realizadas nas pesquisas. Objetivo, nesse contexto, representa o propósito ou o que se pretende alcançar na realização da prática.

A resposta à QP3 resulta na identificação de quais são os objetivos que os pesquisadores desejam alcançar com o PC, podendo trazer informações relevantes para definir uma tendência da pesquisa nacional e relacionar com o cenário mundial. Para a pesquisa deste artigo, os objetivos serão categorizados segundo o trabalho de Barr e Stephenson (2011), que define nove habilidades fundamentais no processo de PC e na resolução de problemas. Essas habilidades foram especificadas a partir das premissas definidas pela *International Society for Technology in Education (ISTE)*¹, *Computer Science Teachers Association (CSTA)*² e *National Science Foundation (NSF)*³ [ISTE, 2016]. O PC não se limita exclusivamente a essas habilidades e nem todas essas devem ser abordadas para caracterizar o processo de PC. Os artigos analisados pela revisão sistemática satisfazem um ou mais desses critérios, através dos resultados obtidos na pesquisa descrita. Com isso, será possível sumarizar os objetivos definidos e alcançados pelos trabalhos e a relação direta com as premissas no uso de PC.

3. Métodos aplicados na Revisão Sistemática

A revisão bibliográfica foi realizada seguindo algumas abordagens da metodologia PRISMA (*Preferred Reporting Items for Systematic Reviews and Meta-Analyses*), descrita em Galvão, Pansani e Harrad (2015). A busca por artigos abrangeu as bases de dados dos próprios mecanismos de busca dos anais eletrônicos dos eventos Simpósio Brasileiro de Informática na Educação (SBIE), Workshop de Informática na Escola (WIE), Workshop sobre Educação em Informática (WEI) e I Workshop de Ensino em Pensamento Computacional, Algoritmos e Programação (WAlgProg), no período de 2007

¹ ISTE - <http://www.iste.org/>

² CSTA - <http://www.csta.acm.org/>

³ NSF - <http://www.nsf.gov/>

a 2015, mas os trabalhos sobre o tema surgiram apenas a partir do ano de 2012. Para todas as bases, a chave de pesquisa utilizada foi “pensamento computacional” ou “*computational thinking*”, tanto no título, quanto no resumo (ou *abstract*). Foram incluídos nas buscas artigos publicados na Revista Brasileira de Informática na Educação, mas não foi encontrado nenhum resultado de pesquisa que atendesse os critérios de inclusão para esta RSL.

3.1. Critérios de inclusão e exclusão

O critério de inclusão dos trabalhos selecionados foi definido como a capacidade do mesmo em responder pelo menos umas das questões da pesquisa. Após esse primeiro filtro, foram definidos outros critérios de inclusão, como mostra a Tabela 1. Os critérios de exclusão definiram diretrizes para excluir os trabalhos que não trariam contribuições à pesquisa, e também podem ser vistos na Tabela 1. Não foram selecionados artigos que apenas faziam uma revisão da literatura, sendo considerados aqueles que descrevem ou aplicam alguma prática pedagógica ou descrevem o uso de ferramentas de ensino.

Tabela 1. Critérios de inclusão e exclusão

Critérios de Inclusão	Critérios de Exclusão
I1. Artigos completos (quatro ou mais páginas)	E1. Artigos no formato de revisão sistemática da literatura
I2. Publicados após o ano de 2006	E2. Estudos que não aprofundem o PC na prática de ensino de programação
I3. Artigos que abordem pelo menos uma prática didática e/ou proponha alguma metodologia e/ou alguma ferramenta de ensino de programação.	E3. Trabalhos que não contemplem nenhum dos componentes da QP3
I4. No trabalho deve haver práticas de programação e PC	

Após as buscas nas bases de dados, houve a aplicação dos filtros com os critérios de inclusão e exclusão através de uma leitura rápida (*scanning*), com ênfase nas seções de resultados e/ou conclusões. Com isso, foram selecionados 6 artigos do WEI, 3 do SBIE, 4 do WIE e 3 do WAlgProg, em um total de 16 artigos. Com a seleção finalizada, todos os trabalhos foram lidos na íntegra e feito o fichamento, seguindo as diretrizes PRISMA.

Pelo fichamento, foi possível criar um formulário de extração, o qual descreve e identifica o artigo e a sua contribuição junto à revisão. Esse formulário possui as seguintes informações: identificador, título, autores, fonte do artigo (evento e ano) e atendimento aos critérios específicos de cada questão de pesquisa. A Tabela 2 mostra o formato do formulário de extração.

Tabela 2. Formulário de extração

Identificador	
Título	
Autores	
Fonte do Artigo	{evento/revista, ano}
Atendimento a questão QP1	[C. “Desplugada”, Jogos Digitais, Ling. Prog, Ling. Prog. Visual, Rob. Pedag.]
Atendimento a questão QP2	[Fundamental, Médio/Técnico, Superior]
Atendimento a questão QP3	[H1, H2, H3, H4, H5, H6, H7, H8, H9]

A lista dos estudos e demais informações se encontram disponíveis para consulta em <<https://goo.gl/3OT4kN>>.

3.2. Ameaças à validade da pesquisa

Esta revisão sistemática utilizou um processo de seleção de estudos através de critérios

de inclusão e exclusão com o objetivo de ser coerente e fiel ao tema proposto. Os critérios de inclusão e exclusão foram cuidadosamente projetados e discutidos entre os autores para minimizar o risco de exclusão de estudos relevantes. A busca foi realizada nas principais bases de dados dos eventos brasileiros, mas é possível que estudos relevantes possam não ter sido incluídos.

Outro fator que ameaça a validade da pesquisa pode estar presente nas decisões subjetivas ocorridas durante o processo de seleção e extração de alguns estudos que não apresentavam uma descrição clara, consequentemente dificultando a aplicação objetiva dos critérios e análise, principalmente em relação à questão QP3. Para minimizar essa ameaça, a seleção e extração foi feita de forma iterativa e colaborativa pelos autores, sendo discutidos os possíveis conflitos em interpretações individuais.

4. Resultados obtidos e discussões

Esta seção apresenta uma visão geral dos estudos que foram selecionados pelos processos de extração e filtragem apresentados na Seção 3. Em seguida, serão apresentados e discutidos de forma detalhada os resultados relacionados às questões de pesquisa apresentados na Seção 2.

4.1. Visão geral dos estudos

Os artigos selecionados são do período entre 2012 e 2015. O intuito inicial era pesquisar artigos publicados após o ano de 2006, ano da criação do termo *Computational Thinking*, mas durante a pesquisa foi identificado que o surgimento de trabalhos nas fontes pesquisadas ocorreu apenas a partir do ano de 2012. A Tabela 3 apresenta a lista de estudos (em ordem alfabética por autores), e seu respectivo evento e ano.

Tabela 3. Lista de estudos em ordem alfabética por autores

ID	Autores	Evento	Ano
E01	de Campos, Gleider M. et al.	WIE	2014
E02	de Carvalho, M. L. B., Chaimowicz, L. e Moro, M. M.	WEI	2013
E03	de França, R. S., dos Santos Ferrerira, V. A. F., de Almeida, L. C. F. e do Amaral, H. J. C.	WEI	2014
E04	de França, R. S. e Tedesco, P. C. A. R.	SBIE	2014
E05	de França, R. S. e Tedesco, P. C. A. R.	WEI	2015
E06	Falcão, T. P. e Barbosa, R.	SBIE	2015
E07	de França, R. S. e Tedesco, P. C. A. R.	WAlgProg	2015
E08	Gomes, T. e Melo, J.	WEI	2013
E09	Ramos, F. e Da Silva Teixeira, L.	WIE	2015
E10	Rodriguez, C., Zem-Lopes, A. M., Marques, L., e Isotani, S.	WIE	2015
E11	Santos, G., Silva, W., Cavalheiro, S., Foss, L., Aguiar, M., Pernas, A. M. e Reiser, R.	WIE	2015
E12	Scaico, P., Duarte, A., Alves, G. M., Maia, M., de Oliveira, F. M., Neto, S. V. M., Dantas, V.	WEI	2012
E13	Schoeffel, P. M., Varela, G. M., Durigon, L. R., Albuquerque, G. C. e Niquelati, M. S.	WAlgProg	2015
E14	Silva, A. e Soares, L. S.	WEI	2015
E15	Viel, F., Raabe, A. e Zeferino, C.	WIE	2014
E16	Zanetti, H. A. P. e Oliveira, C. L. V.	WAlgProg	2015

A Figura 1 mostra a distribuição dos números de artigos que abordam PC e ensino de programação por ano e evento. Neste gráfico, não foram incluídos os artigos do WAlgProg, pois o evento teve sua primeira ocorrência no ano de 2015.

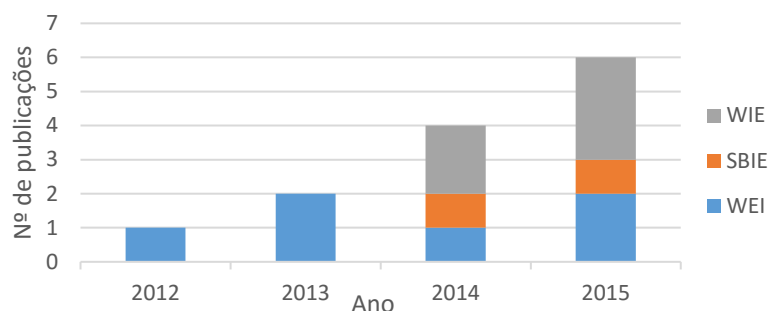


Figura 1. Número de publicação entre 2012 e 2015 nos eventos WIE, SBIE e WEI

Como podemos observar, é crescente o número de publicações que abordam o ensino de programação e PC nos últimos anos, quando considerados todos os eventos. É importante ressaltar também o surgimento de publicações nos eventos SBIE e WIE, a partir de 2014, e o aumento de publicações no WIE no ano seguinte.

4.2. Práticas pedagógicas nos estudos (QP1)

Para esta pesquisa, o contexto da prática pedagógica pode ser interpretado como uma interferência feita pelo professor no desenvolvimento ou aprendizagem de um grupo de alunos. É um procedimento que pode ser adotado no momento em que é apresentado algum problema no aprendizado, com o objetivo de compreendê-lo, explicitá-lo ou corrigi-lo, podendo trazer novos elementos para que os alunos possam pensar e elaborar soluções de maneira diferenciada [Freire e Prado 1996].

4.2.1 Análise e discussões

A categorização dessas práticas que foi utilizada nesta RSL foi feita baseando-se nas ações mais comuns utilizadas em trabalhos recuperados pelas buscas, e nas práticas mais comumente abordados em trabalhos nos eventos pesquisados. Foram definidas cinco categorias, apresentadas na Tabela 4 com uma breve descrição. Essa categorização tem como objetivo prover um mecanismo de mapeamento para futuro pesquisadores que queiram relacionar o uso de PC em ensino de programação e uma prática em específico. Após a discussão dos autores, as categorias definidas foram: Computação “Desplugada” (CD); Jogos Digitais (JD); Linguagem de Programação (LP); Linguagem de Programação Visual (LPV); e Robótica Pedagógica (RP).

Tabela 4. Categorias de práticas pedagógicas

Categoria	Descrição
Computação “Desplugada” (CD)	Métodos que procuram promover o ensino de computação sem o uso de computador, utilizando atividades lúdicas para atingir pessoas de todas as idades [CSUnplugged 2016].
Jogos Digitais (JD)	Utilização de jogos digitais para o ensino de conceitos de computação ou programação de computadores.
Linguagem de Programação (LP)	Utilização de linguagens de programação de alto nível, como paradigma imperativo (<i>e.g.</i> C ou Pascal) ou orientada a objetos (<i>e.g.</i> Java ou Python).
Linguagem de Programação Visual (LPV)	Utilização de linguagens de programação visual, como Scratch ou App Inventor.
Robótica Pedagógica (RP)	Utilização de artefatos robóticos, reais ou virtuais, como ferramenta para de ensino de programação.

Essa análise baseou-se na descrição contida nos textos dos trabalhos pesquisados,

de modo a identificar qual foi a prática escolhida e claramente definida no estudo ou, quando não houve essa descrição clara da prática adotada, quais ferramentas foram adotadas, denotando assim a categoria da prática. Nos casos de proposta de metodologias de ensino, foram verificadas em qual categoria a proposta contemplava. A Tabela 5 mostra os resultados encontrados nessa questão de pesquisa.

Tabela 5. Relação entre as práticas e os estudos

Prática	Estudo (ID)	Qtd.
Computação “Desplugada”	E01, E02, E03, E11, E13, E14	6
Jogos Digitais	E05, E06, E07, E09, E10, E12	6
Uso de Linguagem de Programação	E12, E15	2
Uso de Linguagem de Programação Visual	E02, E03, E08, E09, E10, E14, E16	7
Robótica Pedagógica	E03, E12, E13, E16	4
Não declarada	E04	1

4.3. Níveis de escolaridade (QP2)

Esta questão busca mostrar o cenário das pesquisas feitas no Brasil nos diferentes níveis de escolaridade. Os níveis estão delimitados por ensino fundamental, ensino médio/técnico e ensino superior e os estudos foram classificados de acordo com o nível de escolaridade para os quais as pesquisas se destinaram.

4.3.1 Análise e discussões

Após o levantamento, foi possível perceber que a maioria dos estudos foram voltados ao ensino médio/técnico, totalizando 10 estudos. Os estudos de nível fundamental, somam 6 estudos e, por fim, apenas 1 estudo no ensino superior. É importante destacar que apenas 1 estudo [E03] destinava a pesquisa para dois níveis distintos de escolaridade (fundamental e médio/técnico).

Pelo levantamento de dados, é possível perceber uma tendência de ensino de programação aplicando PC no nível médio/técnico em 10 estudos [E02, E03, E04, E05, E07, E08, E09, E10, E15 e E16]. Alunos nesse nível de escolaridade possuem uma capacidade de pensamento e abstração que possibilita uma melhor assimilação de conceitos de programação e maturidade para desenvolver habilidades fundamentais para o PC, como abstração, decomposição de problemas e análise de dados. Além disso, há uma grande gama de instrumentos de ensino de programação voltados aos alunos do ensino médio, como jogos digitais e plataformas para robótica.

O segundo nível mais abordado nas pesquisas foi o fundamental, com 6 estudos [E01, E03, E06, E11, E13, E14] predominantemente com práticas usando LPV, RP e CD. Houve apenas um estudo aplicado ao nível superior [E12], com práticas de LP, RP e JD.

4.4. Objetivos alcançados nos estudos (QP3)

O objetivo desta questão de pesquisa é verificar quais as habilidades são mais estimuladas no ensino de programação com PC. Nesse contexto, foram utilizadas as habilidades definidas pela ISTE (2016), para categorizar quais habilidades são presentes e abordadas por pesquisadores brasileiros. A Tabela 6 mostra as habilidades e uma breve descrição das mesmas.

Tabela 6. Lista de habilidades

Habilidade	Descrição	Habilidade	Descrição
(H1) Coleção de dados	Processo de recolher as informações adequadas.	(H6) Algoritmos e Procedimentos	Série de passos ordenados e encadeados para resolver um problema ou atingir algum fim.
(H2) Análise de Dados	Encontrar o sentido dos dados, encontrar padrões e tirar conclusões.	(H7) Automação	Usar computadores ou máquinas fazer tarefas repetitivas ou tediosas.
(H3) Representação de dados	Representação e organização de dados em gráficos adequados, cartas, palavras ou imagens.	(H8) Simulação	Representação ou modelo de um processo. Simulação envolve também as experiências em andamento utilizando modelos.
(H4) Decomposição do Problema	Quebrar tarefas em partes menores e gerenciáveis.	(H9) Paralelização	Organizar recursos para, simultaneamente, realizar tarefas para alcançar um objetivo comum.
(H5) Abstração	Reduzir a complexidade para definir ideia principal		

Baseado nos textos presentes no estudo, os autores desta pesquisa identificaram que alguns estudos abordam integralmente ou parcialmente algumas habilidades em suas propostas e, algumas vezes, foi identificada a ausência de algumas habilidades. Para uma demonstração de aplicação dessas habilidades nos estudos, foi definida uma classificação de três modos: *i)* aborda claramente (C) – quando a habilidade está claramente descrita nos objetivos das práticas ou nos resultados e/ou discussões do estudo; *ii)* apresenta indícios (I) – quando se identifica através dos resultados e discussões do estudo que a habilidade foi abordada durante a prática e; *iii)* não identificado (N) – quando a habilidade não foi identificada pelos autores.

4.4.1 Análise e discussões

A Tabela 7 apresenta uma distribuição matricial que possibilita a visualização clara de quais habilidades (colunas) são abordadas nos estudos levantados (linhas), identificados com a classificação (C, I ou N) referente à abordagem da habilidade no cruzamento entre as linhas e colunas. Como é possível observar, cada estudo aborda ao menos duas habilidades e, na maioria, de maneira clara. Há algumas exceções [E03, E04 e E13], mas majoritariamente entre os estudos há uma presença significativa dessas habilidades nas práticas apresentadas, sendo colocadas como objetivos a serem alcançados junto aos alunos. Como já supracitado, um estudo [E07] não contempla uma prática e sim um modelo de aplicação, mas define claramente quais habilidades são abordadas.

Tabela 7. Relação entre as habilidades e os estudos

Estudo	H1	H2	H3	H4	H5	H6	H7	H8	H9	Estudo	H1	H2	H3	H4	H5	H6	H7	H8	H9
E01	I	I	C	N	C	C	N	C	N	E09	C	C	C	C	C	C	C	C	N
E02	C	I	N	C	C	C	C	I	N	E10	C	C	C	C	C	C	C	C	N
E03	N	N	C	N	C	C	N	C	N	E11	N	N	C	N	C	C	C	C	N
E04	N	C	N	C	N	N	N	N	N	E12	I	I	I	C	C	C	C	C	N
E05	N	N	N	C	I	C	N	C	N	E13	N	N	N	C	C	C	N	N	N
E06	I	I	I	C	C	C	N	C	N	E14	C	C	C	C	C	C	N	N	N
E07	C	C	C	C	C	C	N	C	N	E15	C	C	C	N	C	C	N	C	N
E08	C	C	C	C	C	C	N	C	N	E16	C	C	C	C	C	C	N	C	N

Legenda: C – Aborda claramente, I – Apresenta indícios, N – Não identificado

As habilidades H5 e H6 aparecem na maioria dos trabalhos (93,75%), seguidas

por H8 (81,35%), H2, H3 e H4 (75%), H1 (68,75%), H7 (31,25) e, por fim, a habilidade H9, que não aparece em nenhum dos estudos resgatados nesta RSL. A Tabela 4 mostra que há uma predominância de habilidades fundamentais para o ensino de programação, como a abstração (H5) e desenvolvimento de algoritmos (H6), assim como habilidades de coleta, interpretação e representação de dados (H1, H2 e H3, respectivamente).

O uso de modelos e simulação também está presente em grande parte dos estudos, devido às práticas adotadas na maioria dos trabalhos, em especial as que utilizam JD e RP. Essas práticas fornecem ferramental que facilita a criação de modelos intermediários à solução final, enriquecendo o ambiente de aprendizado. A habilidade de decomposição de problemas (H4) também é bastante significativa e aparente nos estudos, que se apresenta como uma premissa essencial no ensino de programação, principalmente nas etapas iniciais, onde os alunos precisam planejar sua solução do problema proposto.

A baixa incidência da habilidade de automação (H7) é devida à abordagem que os pesquisadores adotam em seus estudos. O processo de automação é muito relevante na programação, mas dificilmente o termo “automatizar” é utilizado nesse contexto. Nessa RSL foram identificadas descrições que indicavam que a modularidade de alguns procedimentos de laços de repetição, como práticas usando LP, sugeriam a automação de processos repetitivos. Dentre os estudos levantados não foi possível identificar de maneira clara a intenção de qualquer pesquisador estimular a habilidade de paralelização (H9). O conceito de paralelização é explorado em atividades com alunos mais experientes e, provavelmente, esse pode ser o motivo da ausência nos estudos.

5. Conclusões

Este estudo teve como objetivo levantar e analisar pesquisas existentes com relação a aplicação do PC no ensino de programação, delimitando o cenário no Brasil. Para cumprir este objetivo foi realizada uma revisão sistemática da literatura cobrindo os principais eventos associados à SBC.

Como apresentado na análise sobre a QP1, vemos que o uso de LPV, seguido por JD e CD, são as práticas mais abordadas nos estudos, sendo que alguns estudos utilizavam essas práticas em conjunto. Em práticas utilizando LPV, a ferramenta mais comum foi o *Scratch*, projeto criado pelo *Massachusetts Institute of Technology* (MIT), muito difundido no meio acadêmico e que apresenta resultados positivos. O uso de JD aparece como instrumento de motivação aos alunos, pelo caráter lúdico, desafiador e motivador. A utilização da CD está presente principalmente nos momentos de introdução a noções básicas de computação e programação, sendo um recurso lúdico e de baixo custo.

De acordo com os resultados obtidos é possível observar uma tendência de pesquisas no ensino médio/técnico e no ensino fundamental, como apontam as análises referentes a QP2. O ensino de programação nesses níveis pode ser apoiado por diversas ferramentas, que são de interesse dos alunos nessas faixas etárias correspondentes, como, por exemplo, jogos digitais, programação visual e plataformas robóticas.

Os resultados obtidos na QP3 mostram quais são as habilidades mais abordadas em pesquisas que se apoiam em PC no ensino de programação. É possível notar que habilidades indispensáveis ao programar, e que caracterizam o PC, são comuns entre os trabalhos. Habilidades como coleta, análise e representação de dados, decomposição de problemas, abstração e desenvolvimento de algoritmos encontram-se na maioria dos

estudos.

Como trabalho futuro desta revisão sistemática, espera-se pesquisar e desenvolver novas soluções para o ensino de programação alinhado ao PC, a fim de colaborar com essa área de pesquisa. Após a análise da QP3, é possível notar a demanda por novos métodos que explorem habilidades necessárias para programar, como abstração, decomposição de problemas, entre outros, e, portanto, a continuidade deste trabalho será direcionada para este objetivo.

Referências

- Barr, V. e Stephenson, C. (2011). Bringing computational thinking to K-12: what is Involved and what is the role of the computer science education community?. *Acm Inroads*, 2(1), 48-54.
- Blikstein, P. (2008). “O Pensamento Computacional e a Reinvenção do Computador na Educação”. Disponível em: < <http://bit.ly/1lXlbNn>>. Acesso em: 10 de abr. 2016.
- Brennan, K. e Resnick, M. (2012). New frameworks for studying and assessing the development of computational thinking. In *Proceedings of the 2012 annual meeting of the American Educational Research Association*, Vancouver, Canada.
- CSUnplugged (2016). Computer Science without a Computer. Disponível em: < <http://csunplugged.org/>>. Acesso em: 25 abr. 2016.
- Cuny, J., Snyder, L. e Wing, J. M. (2010). Demystifying computational thinking for non-computer scientists. Manuscrito não publicado. Disponível em: <http://www.cs.cmu.edu/~CompThink/resources/TheLinkWing.pdf>.
- De Paula, B. H., Valente, J. A. e Burn, A. (2014). O uso de jogos digitais para o desenvolvimento do currículo para a Educação Computacional na Inglaterra. *Currículo sem Fronteiras*, v. 14, n. 3, p. 46-71.
- Freire, F. M. e Prado, M. E. B. B. (1996). Professores construcionistas: a formação em serviço. In *Actas do III Congresso Ibero-Americano de Informática Educativa*. Disponível em: <http://lsm.dei.uc.pt/ribie/docfiles/txt200352145836PROFESSORES%20CONSTRUCIONISTAS.pdf>. Acesso em: 08 jan 2014.
- Galvão, T. F., Pansani, T. S. A. e Harrad, D. (2015). Principais itens para relatar Revisões sistemáticas e Meta-análises: A recomendação PRISMA. *Epidemiologia e Serviços de Saúde*, v. 24, p. 335-342.
- ISTE (2016). Computational thinking teacher resources. Disponível em: https://csta.acm.org/Curriculum/sub/CurrFiles/472.11CTTeacherResources_2ed-SP-vF.pdf. Acesso em: 28 jan. 2016.
- Mestre, P., Andrade, W., Guerrero, D., Sampaio, L., da Silva Rodrigues, R. e Costa, E. (2015) Pensamento Computacional: Um estudo empírico sobre as questões de matemática do PISA. In: *Anais dos Workshops do Congresso Brasileiro de Informática na Educação*. p. 1281.
- Wing, J. M. (2006). “Computational Thinking”. *Communications of the ACM*. March, Vol. 49, No. 13.