

# PENSAMENTO COMPUTACIONAL

## Revisão Bibliográfica

Desenvolvido no âmbito do Projeto UFRGS/MEC

TED 676559/SAIFI – Avaliação de Tecnologias Educacionais

com a participação direta de:

Rosa Maria Vicari  
Álvaro Moreira  
Paulo Blauth Menezes

com a colaboração de (em ordem alfabética):

Crediné Silva de Menezes  
Daltro Nunes  
Maria Aparecida C. Livi

2018

## SUMÁRIO

<b>1. Introdução.....</b>	<b>9</b>
<b>2. Objetivo e Metodologia da Pesquisa.....</b>	<b>19</b>
2.1 Objetivo e Limites do Trabalho .....	19
2.2 Seleção da Metodologia .....	19
2.3 Localização e Seleção dos Documentos-alvo do Estudo .....	20
2.4 Coleta de Dados .....	21
2.5 Análise e Interpretação de Dados .....	22
2.6 Avaliação de Materiais Educacionais .....	24
<b>3. Pensamento Computacional .....</b>	<b>25</b>
3.1 Bases do Pensamento Computacional .....	30
3.1.1 Decomposição.....	31
3.1.2 Reconhecimento de Padrões.....	31
3.1.3 Abstração.....	34
3.1.4 Algoritmos .....	35
3.2 O pensamento PC e as ideias de Jean Piaget .....	37
<b>4. Abordagem Pensamento Computacional Unplugged .....</b>	<b>39</b>
4.1 Jogos para PC <i>Unplugged</i> .....	43
4.1.1 Jogo <i>Haathi Mera Saathi (Meu Amigo Elefante)</i> .....	43
4.1.2 Jogo <i>Robot Rally</i> .....	44
4.1.3 Jogo <i>Bits &amp; Bytes</i> .....	45
4.1.4 Jogo <i>Littlecodr</i> .....	46
4.1.5 Jogo <i>Giggle Chip</i> .....	46
4.1.6 Jogo <i>Chocolate Fix</i> .....	47
4.1.7 Jogos <i>Circuit Maze</i> e <i>Laser Maze</i> .....	48
4.1.8 Jogo <i>Code Master</i> .....	48
4.1.9 Jogo <i>Code Monkey Island</i> .....	49
4.1.10 Jogo <i>Coding Is Good</i> .....	50
4.2 Livros que trazem jogos e atividades vinculadas ao PC <i>unplugged</i> .....	51
4.2.1 Livro <i>Computer Science Unplugged... off-line activities and games for all ages</i> .....	51
4.2.2 Livro <i>Teaching Thinking Across the Curriculum</i> .....	52
4.2.3 Livro <i>Lift-the-Flap Computers and Coding</i> .....	54
4.2.4 Livro <i>CodyRoby</i> .....	55
4.3 Considerações sobre os Materiais Utilizados no PC <i>Unplugged</i> .....	56
<b>5. Abordagem Pensamento Computacional Plugged .....</b>	<b>58</b>
5.1 Plataformas Voltadas para PC <i>Plugged</i> .....	58
5.1.1 Ambiente de Programação <i>Scratch</i> .....	59
5.1.2 Ambiente de Programação <i>App Inventor</i> .....	59
5.1.3 Ambiente de Programação <i>Context Free</i> .....	60
5.1.4 Ambiente de Programação <i>NodeBox</i> .....	61
5.1.5 Ambiente de Programação <i>Processing</i> .....	61
5.2 Jogos <i>Plugged</i> .....	63
5.2.1 Jogos <i>Plugged Não-Determinísticos</i> .....	64
5.2.2 Jogo <i>AlphaGo</i> .....	64
5.2.3 Jogo MS. <i>Pac-Man</i> .....	65

5.2.4	Jogo Coding Is Good.....	66
5.3	Livros para o PC Plugged.....	67
5.3.1	Livro Guia Mangá de Bancos de Dados .....	67
5.3.2	Livro Head First Learn to Code.....	68
5.3.3	Livro Coding as a Playground: Programming and Computational Thinking in the Early Childhood Classroom.....	69
5.3.4	Livro Hello Ruby Adventures in Coding .....	70
<b>6.</b>	<b>Avaliação de Materiais Educacionais Voltados ao Desenvolvimento do Pensamento Computacional nas Escolas Brasileiras .....</b>	<b>71</b>
<b>7.</b>	<b>Crítica às Abordagens e à Pesquisa do Pensamento Computacional .....</b>	<b>75</b>
<b>8.</b>	<b>Avaliação do uso do Pensamento Computacional.....</b>	<b>76</b>
<b>9.</b>	<b>Integração na Educação Básica .....</b>	<b>82</b>
9.1	Introdução da BNCC .....	82
9.2	Etapa do Ensino Fundamental .....	83
9.2.1	Linguagens .....	84
9.2.2	Arte.....	87
9.2.3	Língua Inglesa .....	88
9.2.4	Matemática.....	89
9.2.5	Ciências da Natureza .....	93
9.2.6	Ciências Humanas .....	94
9.3	Etapa do Ensino Médio .....	95
9.3.1	As Tecnologias Digitais e a Computação .....	96
9.3.2	Línguas e suas Tecnologias .....	98
9.3.3	Matemática e suas Tecnologias .....	101
9.3.4	Ciências da Natureza e suas Tecnologias.....	102
9.3.5	Ciências Humanas e Sociais Aplicadas.....	102
9.4	A BNCC e o Ensino de Computação Através do PC .....	103
9.5	Análise Apresentada pela SBC para Itens Específicos da BNCC .....	104
9.6	Integrar o PC ao Ensino Básico.....	109
<b>10.</b>	<b>Formação de Professores .....</b>	<b>113</b>
10.1	Foco na Formação para o PC Unplugged .....	114
10.2	Proposta para a Formação de Professores .....	115
<b>11.</b>	<b>Panorama Global da Adoção de Políticas para o Ensino de Computação ...</b>	<b>123</b>
11.1	Países .....	123
11.1.1	Alemanha.....	124
11.1.2	Argentina.....	125
11.1.3	Austrália.....	126
11.1.4	Coreia do Sul.....	127
11.1.5	Escócia .....	128
11.1.6	Estados Unidos da América.....	128
11.1.7	Estônia .....	130
11.1.8	França .....	130
11.1.9	Finlândia.....	131
11.1.10	Israel.....	132
11.1.11	Grécia .....	133

11.1.12 Portugal.....	134
11.1.13 Reino Unido.....	135
11.1.14 Resumo Comparativo Entre as Políticas.....	135
11.2 Iniciativa Privada e Organizações Não-Governamentais .....	138
11.2.1 European Schoolnet .....	138
11.2.2 European Commission (Europa) .....	138
11.2.3 Code.Org (EUA/Global) .....	140
11.2.4 Instituto Horizon.....	141
11.2.5 Programae.org.br (Brasil) .....	142
11.2.6 Supergeeks.com.br (Brasil) .....	142
<b>12. Análise Crítica.....</b>	<b>144</b>
<b>13. Considerações Finais .....</b>	<b>146</b>
<b>14. Referências Bibliográficas .....</b>	<b>149</b>
<b>15. Anexos.....</b>	<b>167</b>

## ÍNDICE DE FIGURAS

Figura 1.1: Desempenho dos alunos brasileiros em Ciências, Leitura e Matemática.	11
Figura 1.2: Vinculação da Computação com os conceitos de PC, Cultura Digital e Mundo Digital, visão da SBC	15
Figura 2.1: Aspectos envolvidos no PC	21
Figura 3.1: Bases do Pensamento Computacional segundo BBC Learning	31
Figura 3.2: Decomposição de problemas em subproblemas	32
Figura 3.3: Similaridade entre as figuras (Reconhecimento de Padrões)	33
Figura 3.4: Sequência de padrões e subpadrões	33
Figura 3.5: Exemplo de exercício de abstração	35
Figura 4.1: Jogo “Mapa do Pirata”	40
Figura 4.2: Jogos para ensino de linguagem	40
Figura 4.3: Autômatos Finitos da Turma da Mônica	41
Figura 4.4: Vestindo Personagens	42
Figura 4.5: Baralho para ensino de Números Binários	42
Figura 4.6: Tabuleiro e cartas do jogo Haathi Mera Saathi	43
Figura 4.7: Tabuleiro de Programação do Jogo Haathi Mera Saathi	44
Figura 4.8: Jogo Robot Rally (edição 2015)	45
Figura 4.9: Jogo “Bits & Bytes”	45
Figura 4.10: Execução de um programa no jogo “Littlecodr”	46
Figura 4.11 Jogo Gigggle Chips: frente (esquerda) e verso (direita)	47
Figura 4.12: Jogo Chocolate Fix	47
Figura 4.13: Jogo Circuit Maze e Laser Maze	48
Figura 4.14: Jogo Code Master (tabuleiros e peças)	49
Figura 4.15: Visão do Jogo Code Monkey Island	49
Figura 4.16: Jogo Code Monkey Island - Cartelas	50
Figura 4.17: Tabuleiros do jogo Coding Is Good	50
Figura 4.18: Tabuleiro do jogo CodingFarmer	51
Figura 4.19: Capa do livro Ensinando Ciência da Computação sem o uso do Computador	52
Figura 4.20: Livro Teaching Thinking Across the Curriculum	53
Figura 4.21: Fichas do Personagem e Pistas	53
Figura 4.22: Exemplo de condicional utilizando pistas na história de Sherlock Holmes	54
Figura 4.23: Exemplo de atividade do livro Lift-the-Flap Computers and Coding	55
Figura 4.24: Peças do livro CodyRoby	56
Figura 5.1: Jogo das letras	64
Figura 5.2: Jogo Go	65
Figura 5.3: Ms. Pac-Man	66
Figura 5.4: Jogo Coding Is Good, versão plugged	66
Figura 5.5: Exemplo do “Guia Mangá de Banco de Dados”	67
Figura 5.6: Livro “Head First Learn to Code”	68
Figura 5.7: Capa do livro Coding as a Playground	69
Figura 5.8: Capa do livro Hello Ruby	70
Figura 6.1: Percentual de critérios atendido pelo conjunto total de materiais desplugados avaliados de acordo com etapa de ensino	72
Figura 6.2: Percentual de critérios atendido pelos materiais desplugados avaliados disponíveis em português, de acordo com etapa de ensino	73
Figura 6.3: Percentual de critérios do Pensamento Computacional atendido pelos materiais plugados avaliados de acordo com etapa de ensino	73

<i>Figura 6.4: Percentual de critérios do Pensamento Computacional atendido pelos materiais plugados avaliados de acordo com etapa de ensino</i>	<i>74</i>
<i>Figura 10.1: Localização dos cursos de licenciatura no Brasil</i>	<i>116</i>
<i>Figura 10.2: Estrutura para abordar o PC na formação de professores</i>	<i>120</i>

## ÍNDICE DE TABELAS

<i>Tabela 1.1: Recursos nas Escolas de ensino Básico do Brasil</i>	12
<i>Tabela 2.1 Documentos selecionados para a Análise de Conteúdo</i>	22
<i>Tabela 6.1: Visão geral dos materiais avaliados</i>	71
<i>Tabela 9.1: Proposta de Nunes 2011</i>	103
<i>Tabela 11.1: Integração da Computação no currículo</i>	136
<i>Tabela 11.2: Quadro Comparativo das políticas adotadas nos países estudados</i>	137

## ÍNDICE DE ANEXOS

<i>Anexo I - Resultados do Teste de Pisa com Utilização de Tecnologia</i>	167
<i>Anexo II - Políticas do MEC mais Relacionadas com o Foco deste Trabalho</i>	169
<i>Anexo III – Exemplos de Conteúdos</i>	170
<i>Anexo IV – Avaliação de Materiais Educacionais Voltados ao Desenvolvimento do Pensamento Computacional nas Escolas Brasileiras</i>	192



# 1. Introdução

A evolução da tecnologia trouxe a possibilidade de resolver problemas complexos. Mas a tecnologia é também cada vez mais indispensável para a resolução de problemas simples do cotidiano. A cada dia mais, as pessoas estão interagindo, em seus lares ou empresas, com computadores, dispositivos móveis e robôs. A robótica embarcada está dominando várias atividades antes só exercidas por humanos. As estatísticas dos Estados Unidos da América (EUA) são de que um robô substitui 6,2 humanos, em média, e na União Europeia (UE), 5,6 humanos, em média (CITI Bank, <https://www.citigroup.com/citi/> - acesso em: 4/12/2018). Enquanto a demanda segue crescendo, por pessoas que saibam utilizar a tecnologia, ou que tenham no mínimo o domínio de alguma área de tecnologia necessária para o seu trabalho, saber criar e projetar tecnologia tornou-se uma condição de domínio ou dominação, que cada vez mais distingue os países. Em particular, pode-se citar a tecnologia digital.

A tecnologia digital codifica, processa e transmite informação. No caso deste documento, o foco está em tecnologias educacionais, expressão que, usualmente, refere-se à aplicação da fluência digital aos conteúdos escolares, de qualquer área, mas pode também englobar o uso de outros recursos tecnológicos (por exemplo, audiovisuais), para auxiliar na Educação (SBC 2018). É importante salientar que o MEC tem adotado um conceito mais amplo para definir o termo tecnologia educacional. Na visão do MEC, esse conceito engloba todo e qualquer aparato tecnológico, digital, analógico ou mecânico, que possa ser utilizado com fins educacionais.

No século passado, as cidades ofereciam incentivos para as empresas nelas se instalarem. Nos dias de hoje, muitas cidades de países desenvolvidos já oferecem incentivos para atrair jovens criativos. Hoje é mais importante o lugar onde se mora do que o local onde se trabalha. O importante é que o local escolhido para a residência ofereça a infraestrutura necessária para o trabalho (mesmo a distância) e para a criatividade.

O Pensamento Computacional PC<sup>1</sup> está diretamente relacionado e seu desenvolvimento é concomitante com o da Ciência da Computação<sup>2</sup>, mas sua proposta, como metodologia, pode ser utilizada em várias áreas do conhecimento,

---

<sup>1</sup> Neste trabalho, a abreviatura PC significa Pensamento Computacional e não *Personal Computer*.

<sup>2</sup> Ciência da Computação: O estudo da Computação é milenar e esteve presente, de maneira independente e em diferentes formas, em quase todas as civilizações ao longo da História. Alguns de seus principais resultados datam do início do século XX e, portanto, antecedem ao primeiro computador como o conhecemos hoje. Já o termo Informática refere-se a uma instância tecnológica atual da Computação, baseada em computadores. Assim, a Computação antecede, convive e possivelmente sucederá a Informática. A Computação pode ajudar a entender/projetar sistemas complexos e assim atingir patamares de conhecimento mais elevados. Trata de como resolver problemas (construção de modelos abstratos, da definição dos passos para a solução, que são os algoritmos), e de como usar as máquinas para resolver esses problemas, através dos programas, que são a linguagem que a máquina entende.

ou seja, o PC é considerado transversal às demais ciências. O *ACM Model Curriculum for K-12<sup>3</sup> Computer Science* (ACM, 2003) defende que é aconselhável o desenvolvimento de habilidades computacionais na educação básica<sup>4</sup>, no sentido da Ciência da Computação poder promover múltiplos caminhos profissionais, desenvolver a capacidade de resolver problemas, apoiar e relacionar-se com outras ciências e motivar os estudantes. Como ressalta Bezerra & Silveira (2011), tal currículo de referência sugere a formação em Ciência da Computação de estudantes da educação básica de acordo com uma abordagem que contempla conceitos, habilidades e competências<sup>5</sup>. Ou seja, o PC é uma metodologia<sup>6</sup>. Uma metodologia aplica-se em diferentes áreas, mas também, para que seja utilizada de forma consciente, precisa ser conhecida. Por esse motivo existe, neste texto, um item sobre a formação dos professores para o uso do PC, de forma análoga às disciplinas de “Metodologia da Pesquisa”. Nessa mesma linha, o relatório *Bugs in the System* da CSTA (*Computer Science Teacher Association*, 2013), afirma que os cursos de Licenciatura em Computação são fracos principalmente em aspectos fundamentais para o PC, tais como ensino de Metodologia Científica e formação multidisciplinar. Ainda, segundo o mesmo relatório, é vital que os programas de formação de professores abordem a falta de formação adequada em Ciência da Computação.

Nesse sentido, o PC precisa estar no mesmo nível da leitura, da escrita e da aritmética. Esforços recentes para capacitar professores para incorporar o PC têm se concentrado no desenvolvimento profissional do professor em serviço, mas há

---

<sup>3</sup> K-12 compreende a soma da educação fundamental e secundária na Índia, Estados Unidos, Canadá, Equador, Coreia do Sul, Turquia, Filipinas, Egito, Austrália, Afeganistão e Irã, antes da faculdade. A expressão abrevia do jardim de infância (K), para crianças de 4 a 6 anos, até o décimo segundo grau (12), para jovens de 17 a 19 anos, o primeiro e o último grau de educação gratuita nesses países, respectivamente.

<sup>4</sup> De acordo com a BNCC, no Brasil a educação básica inclui a Infantil, a Fundamental e a Média. Estas nomenclaturas variam de país para país e procurou-se respeitar isso neste texto. A BNCC organiza-se de modo a respeitar as Diretrizes Curriculares. Desse modo não contempla a ideia de tecnologias digitais ou equivalentes para o período 0 a 3 anos (creche). Já para o período da pré-escola, 4 a 5 anos, contempla-se o uso de tecnologias digitais quando do interesse da criança, pelo menor tempo possível e sempre ligado à busca de informação para o desenvolvimento de projetos.

<sup>5</sup> Na Ásia, os currículos de Singapura e de Xangai são centrados em competências. Recentemente o Japão, a Estônia e a Nova Zelândia, por exemplo, também centraram seus currículos em competências. Todos considerados bons sistemas educacionais no mundo. O contraponto é os Estados Unidos, que não segue oficialmente essa linha, embora alguns estados, usando da liberdade para escolherem seguir ou não as diretrizes gerais, o façam. Os EUA, como um todo, estão em 36º lugar no PISA (*Programme for International Student Assessment*), apesar de serem a 1ª economia do mundo. Ainda, Charbel El-Hani (El-Hani 2007) lembra que o termo ‘competências’ pode significar coisas distintas: para o suíço Philippe Perrenoud (Perrenoud 2018) o foco são as competências para a vida, enquanto para a OECD (*The Organisation for Economic Co-operation and Development*), o foco são as competências para o trabalho.

<sup>6</sup> É o conjunto de técnicas e processos (métodos) empregados para a pesquisa e a formulação de uma produção científica.

uma compreensão limitada de como envolver os professores em serviço de outras áreas, no conteúdo em Ciência da Computação e pensamento computacional.

As principais questões envolvidas se relacionam com:

*Como os educadores de futuros professores do ensino fundamental e médio, ou os cursos de licenciaturas, desenvolvem mecanismos para expor os futuros professores, em formação, a construtos do PC e compreensão desses construtos no contexto de suas áreas temáticas (Matemática, Línguas, Ciências, etc.)? e*

*como desenvolver a base de conhecimento de professores em serviço e professores em formação, para que eles possam fornecer experiências de PC relevantes, envolventes e significativas para seus alunos?*

Para Darling-Hammond e Bransford (2005), os formadores de professores precisam primeiro desenvolver os conhecimentos e as habilidades dos professores em formação, sobre como pensar computacionalmente e, em seguida, como ensinar seus alunos a pensar computacionalmente.

Para parte das escolas brasileiras, que nem possuem luz elétrica (5,5%), totalizando 1.536.217 (um milhão quinhentos e trinta e seis mil duzentos e dezessete) matrículas, que dizer de Internet e computadores, a adoção do PC pode ser a única solução. Mesmo assim, introduzir essa metodologia vai ter um custo, no que se refere à formação de professores e à aquisição de materiais educacionais.

Conforme recente publicação da Organização para a Cooperação e Desenvolvimento Económico (OECD) (2017), o comparativo do Brasil em relação à média mundial do desempenho dos estudantes em Matemática, Leitura e Ciências está muito abaixo dessa média e, principalmente, não se pode dizer que esteja melhorando (ver Figura 1.1). Em leitura, por exemplo, o índice de alunos com nível insuficiente, em 2016, correspondia a 54,73%.

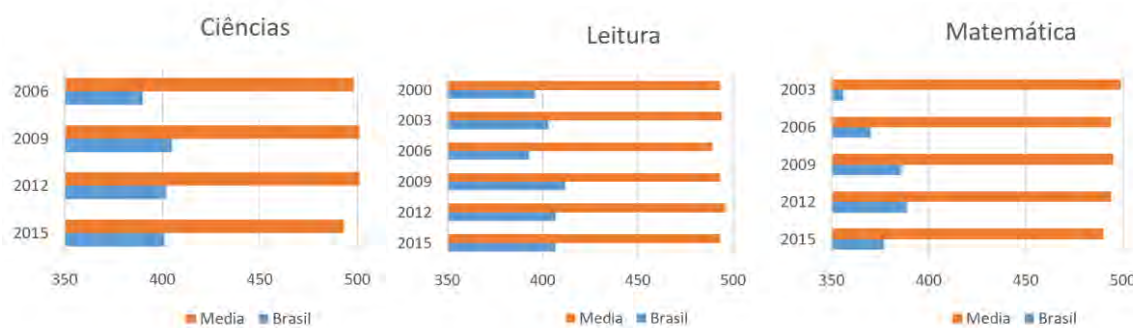


Figura 1.1: Desempenho dos alunos brasileiros em Ciências, Leitura e Matemática.

Fonte: OECD - Organização para a Cooperação e Desenvolvimento Económico - 2017  
<http://jconline.ne10.uol.com.br/canal/mundo/brasil/noticia/2018/02/28/brasil-so-deve-dominar-leitura-em-260-anos-aponta-banco-mundial-329565.php>

Com base nesses dados, o Banco Mundial, em seu relatório *World Development Report* (WBG, 2018) dedicado totalmente à educação, previu que os

estudantes do Brasil levarão 260 anos para atingir o desempenho em Leitura, dos países desenvolvidos e 75 anos, para a Matemática. Mas, quando se trata da leitura digital, o Brasil alcança índices acima de zero. Ou seja, entre 1 e 1,3. (*The Future of Work Regional Perspectives*, 2015).

Ainda, segundo o MEC/INEP – Ministério da Educação e Cultura / Instituto Nacional de Estudos e Pesquisa, há importantes recursos ausentes em escolas do Brasil (ver Tabela 1.1).

*Tabela 1.1: Recursos nas Escolas de ensino Básico do Brasil*

<i>Recurso Ausente</i>	<i>%</i>	<i>Relativo ao Total de Matrículas</i>
Internet	44,2%	12.345.595
Laboratório de Informática	48,8%	13.630.430
Energia elétrica	5,5%	1.536.217

Fonte: MEC/INEP (2017)

Pode-se constatar que o Brasil tem realidades diferentes e espera-se com este estudo poder contribuir para que a SEB/MEC possa tomar decisões a respeito do ensino do PC e de suas potencialidades e limites para o contexto brasileiro. Por exemplo, o Brasil não apresenta, no momento, e provavelmente nem em um futuro próximo, uma demanda reprimida de programadores. Esse fato pode ser considerado na definição de uma política para a introdução do ensino de Ciência da Computação, principalmente pelo MEC, que pode estabelecer parâmetros para uma abordagem mais desvinculada do mercado das grandes empresas de tecnologia, estabelecidas em países desenvolvidos. Tal fato, que em princípio pode parecer uma desvantagem, pode ser utilizado em favor de uma abordagem mais adequada para o uso do PC ou de outras estratégias como, por exemplo, ensino de Ciência da Computação e desenvolvimento de competências para a resolução de problemas complexos, que venham a ser propostas para o ensino de Ciência da Computação, no Ensino Básico.

Alguns países têm implantado um currículo mínimo de Ciência da Computação em suas escolas, como poderá ser constatado neste texto. No entanto, a forma e a metodologia empregadas por cada país são diferenciadas. Ou seja, o que as diferentes propostas de ensino de Ciência da Computação nos diferentes níveis de ensino de cada país têm em comum é a constatação de que esse conhecimento é necessário para a formação de seus cidadãos. Alguns países focam no ensino da Ciência da Computação, outros focam no mercado de trabalho e na empregabilidade para seus cidadãos. Outros ainda defendem uma forma diferenciada de ensino, dependendo dos interesses dos alunos.

No Brasil, o desenvolvimento da Ciência da Computação é essencial do ponto de vista intelectual, econômico, tecnológico e social para preparar as próximas gerações de profissionais. Este texto busca apresentar formas para inserir

conceitos do PC nos currículos escolares, com foco no ensino fundamental, com ênfase nas séries iniciais.

O PC pode auxiliar na resolução de problemas das mais diversas áreas, através de conceitos como abstração, decomposição, entre outros. Além disso, pode incentivar a busca pelo aperfeiçoamento das tecnologias ligadas à informação e comunicação. No entanto, o relatório *Introducing Computational Thinking to Young Learners: Practicing Computational Perspectives Through Embodiment in Mathematics Education* (Sung, W.; Ahn, J-H.; Black, J. 2017) sobre os aspectos pedagógicos do PC argumenta que esse conteúdo poderia colocar os professores em papéis novos e não familiares, na sala de aula, onde os alunos colaboram para resolver problemas. Desenvolver as competências dos professores, em formação, para incorporar o PC em suas futuras salas de aula requer que eles sejam capacitados a pensar computacionalmente, bem como saibam ensinar seus alunos a pensar em computação, especialmente no contexto de áreas específicas. Nesse sentido, os autores consultados (Barr, V.; Stepheson, C. Bringin 2011), (Yadav, A. et al, 2014) (Yadav, A.; Korb, J.T 2012) e (Darling-Hammond, L.; Bransford, 2005) defendem que, os programas de capacitação de professores em serviço ou as Licenciaturas são um lugar natural para apresentar aos professores o PC e o como incorporá-lo em suas disciplinas. O recente movimento em torno do PC é uma oportunidade para, novamente, redefinir e redesenhar os cursos de tecnologia educacional, tornando-os mais relevantes. Na grande maioria dos países estudados, que adotam ou sugerem a introdução do PC nos ensinos fundamental e médio, a capacitação acontece através de disciplinas de metodologia científica e de disciplinas de “Introdução à Tecnologia Educacional”, como parte de uma formação dos professores para as Tecnologias da Informação e Comunicação (TIC).

A Tecnologia da Informação e Comunicação (TIC) engloba todas as áreas que estudam a automação do processamento e da comunicação de informação, incluindo todas as ferramentas necessárias para tal. De modo geral, é possível dizer que a inclusão de conceitos de Pensamento Computacional na Escola Brasileira pode vir a ser uma ação concreta na busca do desenvolvimento de TICs no Brasil. Assim TIC compreende tanto a infraestrutura física (componentes que permitem codificar, armazenar, processar e transmitir a informação), quanto o *software* (aplicações e sistemas).

O PC apresenta distintas abordagens e as diferentes propostas para implementação variam de acordo com as necessidades de cada país. Isso pode ser entendido pelo fato do conceito não estar estabelecido. É possível encontrar várias definições e, portanto, diferentes níveis de integração com a Ciência da Computação. O PC não envolve apenas conceitos e resultados formais, também agrega práticas de projetar sistemas, entender o comportamento humano e o pensamento crítico (Wing, 2010).

Quando iniciar o ensino de Ciência da Computação para as crianças ainda é um tema em debate, principalmente entre os educadores. Mas, os fatos atropelam essa questão, pois muitos bebês, atualmente, utilizam dispositivos eletrônicos

antes mesmo de falar. Nem sempre esses dispositivos são brinquedos, muitas vezes tratam-se de telefones celulares e de *tablets*.

Dentro desse contexto, a BNCC (Base Nacional Comum Curricular) do Brasil optou pela não utilização de tecnologia até os 3 (três) anos de idade e, após, durante o chamado ensino infantil, só utilizá-la, por pouco tempo, e se as crianças o quiserem, para encontrarem conteúdos que podem ser utilizados no desenvolvimento de projetos. Diferentes países possuem diferentes posturas a esse respeito, como pode ser visto, neste texto, no item destinado ao estudo de políticas de outros países (por exemplo, Estônia e Argentina, dentre outros). Quanto ao emprego do PC nessa faixa etária, cabe lembrar que o PC, tanto na sua forma *unplugged* (ou desplugado – o que não utiliza máquina) quanto *plugged* (ou plugado – o que utiliza máquinas, incluindo robôs), pode ser utilizado de maneira lúdica, o que estaria de acordo com a BNCC. Ainda, cabe mencionar a proposta de Israel. O país não dissocia PC da Ciência da Computação e aborda, explicitamente, conceitos como algoritmos evolutivos, linguagens não-determinísticas, dentre outros conceitos que não costumam estar presentes nas propostas dos demais países.

Ainda é importante lembrar que não foi encontrado, no âmbito desta revisão bibliográfica e dos limites da metodologia estabelecida para tal, conteúdo que aborde a Máquina de Turing (MT). A MT é um dispositivo teórico apresentado, em 1936, pelo matemático Inglês Alan Turing (1912-1954), que realiza operações simples, mas que, combinadas, permitem resolver (conforme a Hipótese de Church<sup>7</sup>) qualquer problema computável e que, além de universalmente conhecida, é aceita como uma definição formal de algoritmo. O problema do que é computável é um dos conceitos tratados no PC. Ainda, existem vários simuladores da Máquina de Turing propostos na Internet, tal como o apresentado pela Google, em comemoração ao 100º aniversário de Alan Turing (<https://goo.gl/rIFbYi> - acesso em 4/12/2018). Essas simulações comumente utilizam o alfabeto binário (0 e 1), mas esse alfabeto pode ser trocado por figuras coloridas, para ser mais adaptado a crianças. Nesse sentido foi encontrado apenas um artigo (Bombasar *et al*, 2015), onde é proposto o desenvolvimento de uma ferramenta para esse fim.

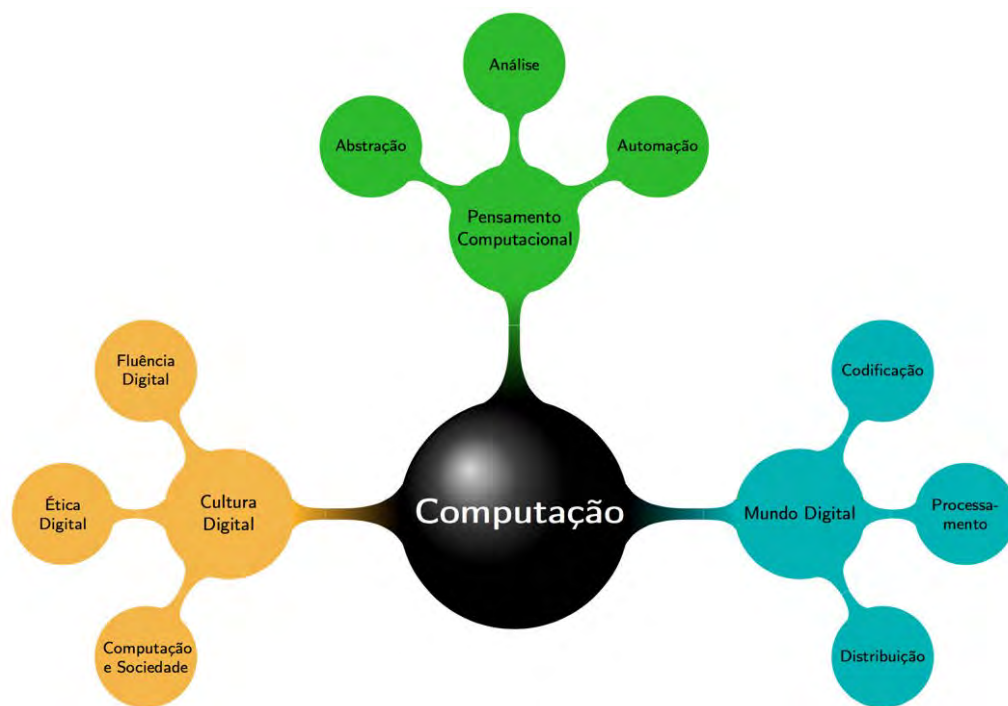
A SBC – Sociedade Brasileira de Computação, em seus documentos de referência para o tema do PC, adota uma abordagem mais ampla. Essa abordagem é apresentada neste documento, para que se possa estabelecer os limites de abrangência de ambas as abordagens. Segundo a SBC (SBC, 2018a), ao PC estão

---

<sup>7</sup> Em Teoria da Computação, a Tese de Church-Turing, ou Tese de Church, assim nomeada em referência a Alogo Church e Alan Turing, é uma hipótese que, em resumo, afirma que a capacidade de computação representada pela Máquina de Turing é o limite máximo que pode ser atingido por qualquer dispositivo de computação. Em outras palavras, a Tese de Church afirma que qualquer outra forma de expressar algoritmos terá, no máximo, a mesma capacidade computacional da Máquina de Turing. Como a noção de algoritmo ou função computável é intuitiva, a Hipótese de Church não é demonstrável. Entretanto, como é assumida como verdadeira (ou seja, é assumida como hipótese) para toda a Ciência da Computação, também é conhecida como Hipótese de Church ou Hipótese de Turing-Church.



vinculados os conceitos de abstração, análise e automação. Esse por sua vez é diretamente vinculado à Computação. Além do PC, a SBC também vincula à Computação o conceito de Cultura Digital, que é formado pelos conceitos de Fluência Digital, Ética Digital e Computação e Sociedade. No presente documento, o conceito de Fluência Digital vem associado a plataformas digitais e, portanto, está diretamente vinculado ao PC *plugged*, que utiliza essas tecnologias digitais educacionais, ou seja, a Fluência Digital é apenas abordada de outra forma. Ainda, na definição da SBC, o conceito de Mundo digital, também está ligado, diretamente, ao conceito de Computação. Mundo digital compreende os conceitos de codificação, processamento e distribuição de informação. Mais uma vez, os conceitos de codificação e processamento, no presente documento, estão diretamente ligados ao PC, principalmente ao PC *plugged*. A Figura 1.2, apresenta a visão da SBC para ilustrar esses conceitos.



*Figura 1.2: Vinculação da Computação com os conceitos de PC, Cultura Digital e Mundo Digital, visão da SBC*

Fonte: SBC 2018a

Sendo assim, para a SBC, o Pensamento Computacional refere-se à capacidade de compreender, definir, modelar, comparar, solucionar, automatizar e analisar problemas (e soluções) de forma metódica e sistemática, através da construção de algoritmos. Apesar de ser um termo recente, vem sendo considerado como um dos pilares fundamentais do intelecto humano, junto com a leitura, a escrita e a aritmética, pois, como essas, serve para descrever, explicar e modelar o universo e seus processos complexos. O PC envolve abstrações e técnicas

necessárias para a descrição e análise de informações (dados) e processos, bem como para a automação de soluções. O conceito de algoritmo está presente em todas as áreas e está intrinsecamente ligado à resolução de problemas, pois um algoritmo é uma descrição de um processo (que resolve um determinado problema).

A compreensão do mundo digital é importante para que o estudante possa apropriar-se dos processos que nele ocorrem, podendo compreender e criticar tendências, sendo ativo nesse cenário. Para uma compreensão estruturada do mundo digital, e não apenas efêmera e permeada de tecnologias, identificam-se 3 pilares principais, chamados de codificação, processamento e distribuição. A codificação diz respeito à representação, no mundo digital, dos mais diferentes tipos de informação que possam ser de interesse. A capacidade de processamento dos dados codificados no mundo digital confere extrema agilidade para desempenhar vários processos, assim como habilita vários outros a acontecerem. De forma indissociável, nesse contexto está a capacidade de distribuição de informação no mundo digital. Essa capacidade é fator fundamental para tamanho impacto do mundo digital. Aqui deve-se prestar atenção que, além de uma facilidade de aceleração do processo de transmissão da informação, testemunha-se dia a dia os impactos de uma mudança singular de paradigma: todos os indivíduos são geradores de informação para o consumo de todos os demais. As fontes tradicionais de informação, outrora acreditadas até certo ponto, dão lugar a um ambiente fragmentado, com incontáveis fontes, muitas vezes desconhecidas. A compreensão do potencial e riscos dessa nova lógica passa pela compreensão do funcionamento da Internet. Ainda, cabe a compreensão de novos paradigmas permitidos pelo mundo digital, onde a Computação está imersa de forma transparente no nosso dia a dia.

Para conseguir estabelecer comunicação e expressão através do Mundo Digital, é necessário um letramento em tecnologias digitais, que no documento da SBC denomina-se de Cultura Digital. Também faz parte da Cultura Digital uma análise dos novos padrões de comportamento e novos questionamentos morais e éticos na sociedade, que surgiram em decorrência do Mundo Digital. A Cultura Digital compreende as relações interdisciplinares da Computação com outras áreas do conhecimento, buscando promover a fluência no uso do conhecimento computacional, para expressão de soluções e manifestações culturais, de forma contextualizada e crítica.

Por fim, a SBC apresenta o conceito de Computação. A Computação é uma área consolidada e independente. A Computação investiga processos de informação, desenvolvendo linguagens e técnicas para descrever processos existentes e também e métodos de resolução e análise de problemas, gerando novos processos.

O documento da SBC coloca também que, tanto para resolver problemas em todas as áreas, quanto para ter uma compreensão do mundo em que vive, todo cidadão do século XXI deve dominar os fundamentos da Computação. O pilar fundamental da solução de problemas é a abstração. Dado um problema a ser



resolvido, é necessário que se construa um modelo abstrato da realidade, incluindo apenas aspectos estritamente relevantes ao problema. Esse modelo ajuda a entender o problema e suas condições de contorno, permitindo a clareza necessária para solucioná-lo. Porém, para que se consiga construir um modelo abstrato que possa ser compreendido e analisado, esse precisa estar descrito em uma linguagem precisa. A Matemática provê uma linguagem formal e universal, que pode ser usada para construir os mais diferentes tipos de modelos, bem como várias técnicas para analisar modelos com precisão. A Computação, como outras ciências, usa a Matemática para a construção de modelos computacionais, modelos de processos. Esses modelos são chamados de algoritmos, e podem estar descritos em vários níveis de abstração diferentes. A Computação provê técnicas e abstrações para auxiliar no processo de construção e análise de soluções, bem como linguagens para descrever algoritmos. Portanto, a Computação provê habilidades distintas das outras áreas de conhecimento. Nesse ponto o documento da SBC estabelece a conexão entre Computação e PC:

*“A habilidade de sistematizar a atividade de resolução de problemas, representar e analisar as soluções através de algoritmos é chamada Pensamento Computacional, e esta exige domínio de objetos abstratos que são necessários para descrever tanto a informação quanto os processos que a manipulam”.*

Dessa forma, é possível estabelecer comparações entre a abrangência dos conceitos tratados no presente estudo e aqueles nos documentos da SBC. O presente estudo trata do PC e, dentro desse tema, são apontadas tecnologias educacionais, tendo em consideração o conceito mais amplo, utilizado pelo MEC. O conceito de fluência digital (ramo Cultura Digital, segundo a visão da SBC), em particular, no presente estudo, está vinculado mais diretamente com o PC *plugged*, no que se refere às plataformas digitais e, portanto, também é aqui abordado. No presente documento não são abordados os conceitos de ética digital e nem de Computação e Sociedade. Isso não significa que os autores deste texto não os considerem importantes. Significa apenas uma opção de corte, para limitar o escopo da revisão bibliográfica realizada. No que se refere ao Mundo Digital, o conceito de distribuição também não foi abordado, por ser ele praticamente inexistente no referencial teórico adotado, para a revisão sistemática do presente estudo.

O Capítulo 1 apresentou a introdução a este estudo. O Capítulo 2 apresenta a metodologia da pesquisa. O Capítulo 3 apresenta os fundamentos do Pensamento Computacional. O Capítulo 4 apresenta a abordagem Pensamento Computacional *unplugged*. O Capítulo 5 apresenta abordagem Pensamento Computacional *plugged*. O Capítulo 6 apresenta uma avaliação de materiais educacionais voltados ao desenvolvimento do Pensamento Computacional nas escolas Brasileiras. O Capítulo 7 explora crítica às abordagens e à pesquisa do Pensamento Computacional. O Capítulo 8 discute avaliação do uso do Pensamento Computacional. O Capítulo 9 apresenta a Integração com Educação Básica no contexto da BNCC. O Capítulo 10 a Formação de Professores. O Capítulo 11 traz um

panorama geral mundial das políticas para a adoção do ensino da Ciência da Computação. O Capítulo 12 apresenta uma visão crítica da adoção do Pensamento Computacional. O Capítulo 13 traz as considerações finais. Após seguem as referências bibliográficas e os anexos.

## 2. Objetivo e Metodologia da Pesquisa

### 2.1 Objetivo e Limites do Trabalho

O objetivo deste trabalho é apresentar o PC da forma como vem sendo abordado e utilizado, internacionalmente e no Brasil, bem como uma significativa amostragem das tecnologias empregadas para a sua introdução, nos diversos níveis da educação básica. Considerando a realidade brasileira, uma atenção especial é dada ao PC *unplugged*.

Este trabalho apresenta os seguintes limites: trata-se de uma revisão bibliográfica e, portanto, não se trata de um trabalho original. Ou seja, o propósito não é apresentar uma nova proposta e nem representar a opinião pessoal dos autores sobre o tema, mas realizar um levantamento bibliográfico efetuado nos limites da metodologia adotada e das palavras-chave utilizadas. O texto apresenta como autores nacionais e internacionais vêm abordando e aplicando o tema, com foco particular nos últimos 3 (três) anos. No caso de identificada alguma tendência na revisão bibliográfica, esta pode ser assumida para dar prosseguimento ao tema. Nesse caso, tal fato é destacado.

Na realização de estudos de revisão bibliográfica é comum a utilização de uma revisão sistemática de literatura na Web e nas bases de artigos científicos reconhecidas. Neste caso, optou-se por buscar artigos predominantemente dos últimos três anos, a não ser quando se tratavam de artigos que introduziram ideias e conceitos.

### 2.2 Seleção da Metodologia

Para a revisão bibliográfica foram utilizadas as cinco primeiras etapas metodológicas das sete recomendadas em *Cochrane Handbook for Systematic Reviews of Interventions* (Higgins, J. PT.; Green, S., 2011):

1. Localização e seleção dos documentos-alvo do estudo;
2. Coleta de dados;
3. Análise e apresentação dos dados;
4. Interpretação dos dados;
5. Avaliação crítica dos resultados alvo do estudo.

As duas últimas etapas, dentre as sete da metodologia propostas em *Cochrane Handbook for Systematic Reviews of Interventions*, não foram utilizadas, pois tratam da necessidade contínua da atualização da revisão bibliográfica durante trabalhos de prazo mais longo.

## 2.3 Localização e Seleção dos Documentos-alvo do Estudo

Para a busca dos artigos foram utilizadas além da Web, duas bases internacionais de artigos científicos: a Scopus e a *Web of Science* (WOS) e uma base internacional de teses e dissertações, a *Networked Digital Library of Thesis and Dissertations* (NDLTD), que inclui o Instituto Brasileiro de Informação em Ciência e Tecnologia (IBICT), do Brasil.

Como se trata de um estudo bem específico, as palavras-chave utilizadas para a localização dos textos foram:

- Pensamento Computacional *unplugged*<sup>8</sup>;
- Pensamento Computacional *plugged*<sup>9</sup>;
- Pensamento Computacional Concorrente<sup>10</sup>.

Essas palavras-chave foram as selecionadas, pois as duas primeiras (Pensamento Computacional *unplugged* e Pensamento Computacional *plugged*) representam as duas formas como essa metodologia vem sendo utilizada nas escolas e definem a elaboração e apresentação dos materiais de ensino utilizados para tal. A terceira (Pensamento Computacional Concorrente) foi selecionada por representar uma tendência atual para a abordagem do tema. Ou seja, todos os artigos selecionados e utilizados nas referências bibliográficas desses estudos poderão ser encontrados nesses repositórios. As pesquisas foram realizadas nas línguas inglesa e portuguesa. Dentre o total de artigos e textos de teses e dissertações encontrados, selecionou-se os com maior aderência ao tema e, principalmente, os que apresentam foco no ensino básico (infantil, fundamental e médio). Devido ao estabelecido na BNCC, o ensino infantil não terá ênfase especial neste texto.

A Figura 2.1 resume alguns dos aspectos que podem ser encontrados com frequência em diferentes propostas de uso do Pensamento Computacional. Em particular, observa-se que, o PC como um todo, não é desvinculado do uso de equipamentos e do ensino da programação. Essa característica encontra-se presente na maioria dos textos estudados e também nas políticas dos países que adotam essa metodologia para a introdução do PC nos currículos do ensino básico. Também, observa-se que o emprego principal está no ensino de programação procedimental, ou seja, o PC concorrente e não determinístico não é ainda explorado. Pode-se constatar também que o PC possui abrangência

---

<sup>8</sup> O Pensamento Computacional *Unplugged* é o que não utiliza máquinas.

<sup>9</sup> O Pensamento Computacional *Plugged* é o que utiliza máquinas (incluindo robôs).

<sup>10</sup> O Pensamento Computacional Concorrente frequentemente refere-se à "concorrência falsa", onde o "paralelismo" de fato não necessariamente existe no *hardware* e é simulado usando uma intercalação de processos, dando uma "aparência" de concorrência. Esse procedimento é adotado na maioria dos Sistemas Operacionais.

multidisciplinar, embora, o foco principal das implementações em curso esteja na Matemática (90% dos artigos encontrados) e nas Ciências (86% dos artigos encontrados).



Figura 2.1: Aspectos envolvidos no PC

Fonte: Google - Pensamento Computacional

## 2.4 Coleta de Dados

Utilizando as palavras chave, nas bases Scopus, WOS e na Internet, foram encontrados um total de 1.040 (mil e quarenta) artigos acadêmicos. A grande maioria sobre o PC *unplugged*. O menor número de artigos trata do PC concorrente, 5 (cinco), e nenhum artigo foi encontrado sobre o PC não procedimental (apenas na legislação de Israel é possibilitado às escolas optarem por essa modalidade). Na base de teses e dissertações foram encontradas apenas 104 (cento e quatro) teses e dissertações. Aqui também a maioria foca o PC *unplugged*. Desses artigos foram selecionados 109 (cento e nove) para estudo mais detalhado e três teses de doutorado. Todos os relatórios técnicos e documentos sobre políticas de governos foram encontrados na Internet. Grande parte dos artigos selecionados nas bases Scopus e WOS foram também encontrados na Internet ou no repositório *ResearchGate* e *Google for Education*. Os acessos foram feitos a partir desses endereços, pelo fato de serem livres, ou seja, sem custo. (Os acessos foram realizados em março de 2018). Dos artigos encontrados, foram selecionados os que servem de base para este trabalho e se encontram citados nas referências bibliográficas.

## 2.5 Análise e Interpretação de Dados

Como se trata de um estudo específico, duas etapas da metodologia foram unificadas, a saber, as etapas de análise e de interpretação dos dados obtidos na fase de coleta.

A seguir, são apresentados, na Tabela 2.1, os números de artigos, relatórios e sítios selecionados, que foram pesquisados e utilizados nas várias partes deste trabalho:

*Tabela 2.1 Documentos selecionados para a Análise de Conteúdo*

	<i>Artigos</i>	<i>Teses</i>	<i>Relatórios técnicos</i>
Embasar a introdução	2		8
Apresentar o PC, sua origem e suas definições	17		
Embasar o estudo	8	3	
Abordar o tema da integração do PC com o ensino fundamental	11		
Formar o estudo apresentado sobre PC <i>unplugged</i>	44		3
Breve apresentação do PC <i>plugged</i>	35		3
PC concorrente/paralelo	5		
Compor o texto sobre a formação de professores	30	1	
<i>TOTAL</i>	<i>152</i>	<i>4</i>	<i>14</i>

O PC *plugged* é a forma mais utilizada pela maioria dos países, principalmente para o ensino fundamental (séries iniciais do 1º ao 5º ano). Mas embora a metodologia PC seja uma forma diferente de se abordar a formação em Ciência da Computação que é fornecida para os estudantes do ensino básico, pelo fato dela utilizar máquinas<sup>11</sup> (computadores, tablets, celulares e robôs), a metodologia é pouco explorada, 12 (doze) artigos selecionados tratam do tema. Pela pesquisa realizada pode ser constatado que esse fato se deve ao foco do uso do PC no ensino. Basicamente o foco está no ensino de programação, até mesmo na maioria

<sup>11</sup> Todo o tipo de *hardware*. Escolher as máquinas adequadas é uma etapa importante para que elas ajudem na solução de problemas.

dos casos que utilizam a robótica, e não no ensino de Ciência da Computação. O aspecto Ciência da Computação do PC é ainda pouco explorado. A justificativa encontrada é que o “mercado” demanda programadores. Esse fato justifica também a pouca pesquisa sobre o PC concorrente, ou seja, nos 3 (três) artigos encontrados a possibilidade é apenas mencionada. Na apresentação das políticas e programas de diversos países foram utilizados 57 (cinquenta e sete) textos, entre artigos e relatórios técnicos. Da mesma forma, foi feito acesso a 18 (dezoito) sítios de instituições governamentais e empresas da área de Ciência da Computação. Finalmente, 7 (sete) textos foram utilizados para embasar as considerações finais.

Ainda, a revisão bibliográfica demonstra que grande parte da literatura trata dos mesmos aspectos, sem apresentar uma evolução significativa para os pontos principais dessa metodologia, que estão exatamente no pensamento computacional para além dos algoritmos e da programação. Essa precariedade de estudos nesses outros aspectos justifica a falta de material pedagógico para apoiar os professores do ensino básico, bem como expõe a lacuna na capacitação desses profissionais. Embora 31 (trinta e um) textos tratem da formação de professores, será possível constatar no Capítulo 10 - Formação de Professores, dedicado a esse assunto, que faltam métodos, métricas e conteúdos para tal. De forma geral, os artigos científicos que tratam sobre o tema da formação de professores são mais antigos, do início dos anos 2000 e, por este motivo, ficam fora do intervalo de três anos adotado neste texto. Mesmo assim, recuperamos textos básicos e os discutimos no contexto da introdução e do Capítulo 10 - Formação de Professores. Quanto ao material fornecido para o uso em sala de aula do PC *unplugged*, a grande maioria está focada em jogos lineares e que utilizam o pensamento repetitivo. Quando 5 (cinco) textos tocam no assunto PC concorrente, e na busca de um espaço de soluções, remetem-se, invariavelmente, ao jogo de xadrez.

Cabe salientar ainda, que em três documentos os termos Ciência da Computação e Informática<sup>12</sup> são utilizados de forma indiferenciada, embora os referidos textos abordem o PC. Esse fato demonstra o pouco cuidado na elaboração das políticas educacionais, pois Computação pode ser ensinada sem máquinas, mas Informática não.

A etapa de análise crítica dos dados obtidos encontra-se distribuída no texto, ao final de cada capítulo.

Após a colocação dos aspectos metodológicos utilizados no texto, segue a apresentação do estudo em detalhes.

---

<sup>12</sup> Informática é um termo usado para descrever o conjunto de atividades ligadas ao armazenamento, à transmissão e ao processamento de informações em meios digitais, estando incluídas nesse grupo: a Teoria da Informação, o Cálculo, a Análise Numérica e a modelagem dos problemas.

## 2.6 Avaliação de Materiais Educacionais

Para atingir de forma mais efetiva o objetivo do presente trabalho, especialmente no que se refere a apresentar uma significativa amostragem das tecnologias empregadas para a introdução do Pensamento Computacional, foi realizado um trabalho de avaliação de um conjunto representativo de materiais didáticos sobre PC, plugadas (*plugged*) e desplugadas (*unplugged*), em português ou em outros idiomas.

Contudo, tendo-se em vista a dificuldade de acesso adequado a muitos desses itens, ampliou-se o método de busca para identificar outros materiais que também pudessem ser utilizados para este mesmo fim. Foram utilizados três caminhos de busca complementares:

- busca por itens relacionados aos materiais inicialmente alavancados neste trabalho;
- busca por materiais desenvolvidos e disponibilizados por grupos de pesquisa do diretório do CNPq;
- busca exploratória por materiais comerciais.

A partir da combinação de metodologias, os números de tecnologias pré-selecionadas para avaliação foram:

- Desplugadas: 37
- Plugadas: 34
- Total: 71

Entre elas, em função de sua disponibilidade foram efetivamente selecionadas:

- Desplugadas: 7
- Plugadas: 10
- Total: 17

A disponibilidade do material foi considerada como acesso via Internet sem custo para aquisição de sua versão completa ou versão de avaliação.

No item 15 - Anexos, encontra-se o relatório detalhado, dividido em quatro partes: na (1) *Contextualização da avaliação* explicitamos os critérios e o processo de seleção das tecnologias PC que foram avaliadas; na (2) *Descrição do processo avaliativo* explicamos como foi constituída a equipe avaliadora (com os critérios de seleção da mesma) e como foram estabelecidos os critérios e instrumentos de avaliação; no (3) *Detalhamento das Avaliações* apresentamos as planilhas avaliativas, os pareceres descritivos e os gráficos representativos das avaliações. Por fim, nas (4) *Considerações finais* objetivamos explicitar, potencialidades e limitações das tecnologias PC disponíveis no mercado brasileiro.



### 3. Pensamento Computacional

O Pensamento Computacional é uma metodologia que se adquire aprendendo conceitos da Ciência da Computação e, portanto, não se caracteriza como uma disciplina por si. Entretanto, como metodologia, pode e deve ser utilizado, de maneira interdisciplinar, em qualquer outra disciplina.

Acima de tudo, o PC não pode ser confundido com Informática, que necessariamente envolve máquinas e a execução de programas. Ou seja, o estudo e a utilização do PC não necessariamente envolvem máquinas. Podemos utilizá-lo *unplugged* ou *plugged*. Mas ele exige um grande grau de abstração, o que torna sua aplicação muitas vezes inadequada, se ocorrer sem a devida capacitação dos professores. Além disso, a confusão com a Informática acontece, sobretudo, porque no PC os algoritmos desempenham um papel importante e seu uso, na maioria das aplicações, visa ensinar programação, principalmente no ensino médio. A forma de abordagem mais utilizada é a sequencial<sup>13</sup> e procedimental. O PC concorrente e/ou não determinístico não é ainda utilizado, embora existam na literatura alguns poucos textos a esse respeito. Por exemplo, no âmbito do PC, a abordagem da Inteligência Artificial (IA), com algoritmos dinâmicos (não estáticos), não é praticamente abordada, nem mesmo via teoria dos grafos. Neste estudo foi encontrada apenas uma plataforma<sup>14</sup> *plugged*, que utiliza algoritmos genéticos na sua concepção (como poderá ser visto no Item 5.1 - Plataformas Voltadas para PC *Plugged*).

O PC *unplugged* é mais difundido com crianças, nos anos iniciais de sua formação (muitos países já ensinam conceitos de Ciência da Computação a partir do ensino infantil, diferente do proposto na BNCC brasileira). Costuma utilizar jogos, em sua maioria colaborativos, para abordar as noções de Computação. Ou seja, a metodologia traz a proposta do ensino colaborativo. Atualmente, o ensino colaborativo é destaque e esse fato se reflete no *software* educacional<sup>15</sup>.

O PC *plugged* é a forma mais utilizada internacionalmente. As ideias são as mesmas, mas os alunos utilizam máquinas no processo de aprendizagem.

A definição do que significa Pensamento Computacional não está estabelecida. A expressão PC foi utilizada por Jeannette Wing, em 2006, de forma explícita. No entanto, as ideias do PC já estavam presentes no artigo *Twenty things to do with a computer*, de Seymour Papert e Cynthia Solomon, escrito em 1972 (Papert; Solomon, 1972). Em 1980, Papert, em seu livro intitulado *Mindstorms*:

---

<sup>13</sup> A sequencialidade é uma restrição ao pensamento natural ou "livre", tipicamente concorrente e não-determinista.

<sup>14</sup> Neste texto o termo plataforma significa genericamente um sistema de *software* e eventuais dispositivos físicos, que disponibiliza uma série de recursos, síncronos e assíncronos, que dão suporte ao processo de aprendizagem, permitindo seu planejamento de acordo com as necessidades do usuário. Não necessariamente um *Learning Management System*.

<sup>15</sup> Uma visão interessante do PC pode ser encontrada no sítio <http://lite.acad.univali.br/pt/pensamento-computacional/> - acesso 8/12/2018.

*Children, Computers, And Powerful Ideas* (Papert, 1980), utilizou a expressão Pensamento Computacional, porém os seus princípios não foram explorados na altura. Essa obra de Papert aborda os computadores e o papel da tecnologia no ensino de crianças. Ainda, Ryan (1987) também já falava na ideia do ensino de conceitos da Ciência da Computação de forma lúdica, e lançou o livro *Teaching Thinking Across the Curriculum* dedicado ao tema. Esse livro é muito utilizado ainda nos dias de hoje.

Em 2006, a expressão PC voltou a ser utilizada e, dessa vez, causou repercussão, devido ao artigo resumido, de três páginas, publicado por Jeannette M. Wing, uma das pesquisadoras mais influentes da área, sobre o que é o PC e como esse pode ser útil para qualquer pessoa, independentemente da área (Wing, 2006). Entre 2006 e 2014, Wing redefiniu essa expressão várias vezes.

Sua mais recente definição é de 2014, quando a autora faz uma pequena alteração na definição anterior afirmando que PC:

*“são os processos de pensamento envolvidos na formulação de um problema e que expressam sua solução ou soluções eficazmente, de tal forma que uma máquina ou uma pessoa possa realizar”;*

ainda complementa como sendo uma *“automação da abstração”* e *“o ato de pensar como um cientista da Computação”*.

Já Liukas (2015), coautora do currículo de Ciência da Computação da Finlândia, resume a definição como:

*“pensar nos problemas de forma que um computador consiga solucioná-los”.*

Mas, apesar da definição, Liukas entende que o PC é executado por pessoas e não por computadores.

Nesse sentido, sabe-se que nem todos os problemas podem ser solucionados por máquinas. Existe uma categoria chamada de NP-Completo<sup>16</sup>, ou o "problema de decisão", na lógica simbólica<sup>17</sup>, que consiste em achar um algoritmo genérico, para determinar se um dado enunciado da lógica de primeira ordem pode ser provado

---

<sup>16</sup> Na teoria da complexidade computacional, a classe de complexidade é o subconjunto dos problemas NP, de tal modo que todo problema em NP se pode reduzir, com uma redução de tempo polinomial, a um dos problemas NP-completo. Se fosse possível encontrar uma maneira de resolver *qualquer* problema NP-completo rapidamente (em tempo polinomial), então poderiam ser utilizados algoritmos para resolver *todos* os problemas NP rapidamente. Como exemplo de um problema NP-completo, pode-se utilizar a soma dos subconjuntos, ou seja, dado um conjunto S de inteiros, determinar se há algum conjunto não vazio de S cujos elementos somem zero. É fácil verificar se uma resposta é correta, mas não se conhece uma solução significativamente mais rápida para resolver esse problema do que testar todos os subconjuntos possíveis, até encontrar um que cumpra a condição. Na prática, o conhecimento de NP-completo pode evitar que se desperdice tempo, tentando encontrar um algoritmo de tempo polinomial, para resolver um problema, quando esse algoritmo não existir.

<sup>17</sup> A Lógica simbólica, ou lógica matemática, substitui a palavra por ideia. Ideias são entes artificiais, de razão, e por isso, a lógica simbólica, em vez de se fixar em termos e palavras, faz uso de símbolos, que substituem qualquer proposição, conferindo-lhes valores verdade (verdadeiro ou falso).

com a tecnologia das máquinas atuais<sup>18</sup>. Ou seja, se não existe uma Máquina de Turing<sup>19</sup> capaz de resolver o problema, ele não é computável. Esses problemas, embora façam parte do PC, não são abordados com cuidado na literatura atual, nem mesmo por Liukas.

Assim, cabe nesta introdução ao PC, apontar os limites físicos dos sistemas computadores em geral. Qualquer sistema computador, independentemente de porte, configuração ou rede a qual está conectado (ou até mesmo qualquer modelo teórico imaginável), possui fortes limitações do que pode computar (ou dos problemas que consegue solucionar). De fato, em Teoria da Ciência da Computação estuda-se que o cardinal do universo dos problemas não-computáveis (ou não-solucionáveis por sistemas computadores) é infinito. Portanto, é importante entender que existe uma infinidade de problemas reais para os quais não existe solução usando um sistema computador.

Ainda, a arquitetura das máquinas atuais, sejam elas sequenciais ou paralelas, é composta por três processadores: o lógico, o aritmético e o gráfico (muitos o consideram diferente do matemático). Sabe-se que a tomada de decisão nos humanos envolve afetividade. Esse processador está faltando ainda no *hardware*, o que limita o poder de tomada de decisão das máquinas a processos matemáticos ou/e lógicos limitados pelo conceito de arquitetura que cada máquina possui.

Buscando outros autores, que tratam da definição do PC, pode-se citar Bundy (2007) e Nunes (2011) que definem o PC como:

*habilidades comumente utilizadas na criação de programas computacionais, como uma metodologia para resolver problemas específicos nas mais diversas áreas.*

Esses autores são os primeiros a tratar o PC como uma metodologia. Essa postura é a adotada neste texto. É importante salientar que esses autores definem o PC como habilidades utilizadas para programação e como uma metodologia para resolver problemas.

---

<sup>18</sup> O modelo de Turing data de 1936 (Turing 1936), muito antes do primeiro computador como se conhece hoje. Esse fato levanta outro aspecto discutido neste texto, que é a independência de conceitos entre Computação e Informática.

<sup>19</sup> A máquina de Turing é importante porque permite definir a noção de computabilidade e de algoritmo: define as possibilidades, mas também permite entender os limites da Computação. A máquina de Turing é um modelo teórico, independente de máquina concreta (computador, por exemplo). Turing também mostrou que existe uma *máquina universal*, que é capaz de simular o comportamento de qualquer outra máquina de computar.

Cabe ainda salientar que, quando o PC é abordado de forma conceitual, os limites do universo discreto<sup>20</sup> <sup>21</sup> das máquinas não existem, o que libera o pensamento lógico de artifícios como, por exemplo, controle, sequência e corte, presentes em todas as linguagens de programação, mesmo nas declarativas, pois são limites ligados à atual arquitetura das máquinas.

Discutidos os limites atuais da Ciência da Computação e da Informática, passa-se a apresentar as habilidades com potencial de serem desenvolvidas em um trabalho com PC, que podem contribuir para uma melhor formação dos estudantes. Dentre elas pode-se destacar:

- Entender quais os aspectos de um problema que podem ser resolvidos usando Ciência da Computação;
- Avaliar e escolher as ferramentas e técnicas computacionais adequadas para cada problema;
- Entender as limitações das ferramentas e técnicas computacionais;
- Reconhecer oportunidades de usar a Ciência da Computação de novas maneiras;
- Aplicar estratégias computacionais, como divisão e conquista, em qualquer área;
- Reformular problemas, para permitir a aplicação de técnicas computacionais na sua solução;
- Fazer novas descobertas, através da análise de grandes volumes de dados;
- Fazer novas perguntas, que não eram sequer cogitadas pelas dificuldades de escalabilidade que suas respostas necessitam;
- Explicar problemas e soluções em termos computacionais.

Com base nessas habilidades, mostram-se adequadas tanto a definição apresentada no *Google for Education* (2015):

*“uma abordagem usada para a solução de problemas utilizando o que se sabe sobre Computação”,*

quanto a da Sociedade Real do Reino Unido (Furber, 2012):

*“o processo de reconhecer aspectos da computação em um mundo que nos cerca e, aplicar ferramentas e técnicas da Ciência da Computação para entender e argumentar sobre sistemas e processos naturais e artificiais”.*

Em 2011, a *International Society for Technology in Education* (ISTE), em conjunto com a *Computer Science Teachers Association* (CSTA), divulga a “definição

---

<sup>20</sup> Discreto refere-se ao universo finito ou limitado ao conjunto dos números naturais (ou estruturas isomorfas a esse), refletindo uma ideia de que seus elementos podem ser contáveis ou enumerados (segundo algum critério) de tal forma que não existe um elemento entre quaisquer dois elementos consecutivos.

<sup>21</sup> Contínuo, em oposição a discreto, reflete uma ideia de “não-contável” ou “não-enumerável”. Exemplos comuns de contínuo são o conjunto dos números reais e o Cálculo Diferencial e Integral.

operacional” do PC que passou por um processo de avaliação e aprovação de quase 700 professores de Ciência da Computação (CSTA/ISTE, 2011), a saber:

*“O Pensamento Computacional é um processo de resolução de problemas que inclui (mas não está limitado a) as seguintes características”:*

- Formulação de problemas de forma que nos permita usar um computador e outras ferramentas para nos ajudar a resolvê-los;
- Organização e análise lógica de dados;
- Representação de dados através de abstrações, como modelos <sup>22</sup> e simulações;
- Automatização de soluções através do pensamento algorítmico (uma série de etapas ordenadas);
- Identificação, análise e implementação de possíveis soluções com o objetivo de alcançar a combinação mais eficiente e efetiva de etapas e recursos;
- Generalização e transferência de um determinado processo de resolução de problemas, para uma grande variedade de problemas.

Essas características são apoiadas e reforçadas por uma série de qualidades ou atitudes que são consideradas as dimensões do PC. Essas qualidades ou atitudes incluem:

- Confiança em lidar com a complexidade;
- Persistência ao trabalhar com problemas difíceis;
- Tolerância para ambiguidades;
- A capacidade de lidar com os problemas em aberto;
- A capacidade de se comunicar e trabalhar com outros para alcançar um objetivo ou solução em comum.

O conceito de PC está em constante evolução e sua definição, bem como os seus limites, igualmente evoluem. Assim, como era de se esperar, não foi encontrada uma definição precisa ou formal do termo, pois o PC não envolve apenas conceitos e resultados formais de Ciência da Computação, também agrega práticas de projetar sistemas, entender o comportamento humano e o pensamento crítico (Wing, 2010) (ou seja, esse conjunto de definições representam o estado atual do PC). Esse fato não limita o surgimento de novas propostas suportadas por outros conjuntos de referências.

Grover e Pea (2013) resumiram em 9 (nove) os elementos que o PC tende a atender, para apoiar a aprendizagem dos alunos de forma interdisciplinar, bem como avaliar o seu desenvolvimento. No que se refere à avaliação, infelizmente, ainda não foi desenvolvido um protocolo que contemple esses aspectos. São eles:

- Abstração e reconhecimento de padrões (incluindo modelos e simulações);
- Processamento sistemático da informação;

---

<sup>22</sup> Modelos semânticos, linguagens e técnicas de análise do problema são métodos formais da Computação.

- Sistema de símbolos e representações;
- Noções de controle de fluxo em algoritmos;
- Decomposição de problemas estruturados (modularização);
- Pensamento iterativo, recursivo e concorrente;
- Lógica condicional;
- Eficiência e restrições de desempenho;
- Depuração e detecção de erro sistemático.

O PC propõe-se ser um pensamento filosófico, um paradigma, e na sua componente Computação pode ser entendido dessa forma. Mas ele também tem a componente Engenharia, que traz a busca pela solução de problemas, que é, atualmente, a parte dominante do PC.

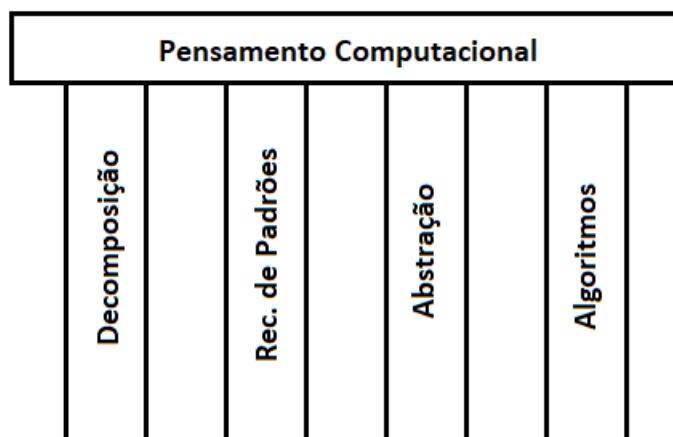
### 3.1 Bases do Pensamento Computacional

Pesquisas lideradas pela instituição Code.Org (2016), Liukas (2015), BBC Learning (2015), Grover e Pea (2013) e o guia Computer at School (Csizmadia et al, 2015) geraram os “Quatro Pilares do PC”, ou bases do PC, que são:

- Decomposição;
- Reconhecimento de Padrões;
- Abstração;
- Algoritmos.

O Pensamento Computacional envolve identificar um problema (que pode ser complexo) e quebrá-lo em pedaços menores de mais fácil análise, compreensão e solução (*decomposição*). Cada um desses problemas menores pode ser analisado individualmente em profundidade, identificando problemas parecidos que já foram solucionados anteriormente (*reconhecimento de padrões*), focando apenas nos detalhes que são importantes, enquanto informações irrelevantes são ignoradas (*abstração*). Passos ou regras simples podem ser criados para resolver cada um dos subproblemas encontrados (*algoritmos ou passos*). Os passos ou regras podem ser utilizados para criar um código ou programa, que pode ser compreendido por sistemas computacionais e, consequentemente, utilizado na resolução de problemas complexos, independentemente da carreira profissional que o estudante deseje seguir. Esses pilares surgiram com a proposta do PC e não foi encontrada uma justificativa para a escolha desses e não de outros.

Assim, o PC utiliza essas quatro dimensões ou pilares, para atingir o objetivo principal: a resolução de problemas. Os pilares que formam a base do PC podem ser resumidos na Figura 3.1. Esses Pilares são interdependentes durante o processo de formulação de soluções computacionalmente viáveis. Esses pilares são detalhados na sequência.



*Figura 3.1: Bases do Pensamento Computacional segundo BBC Learning*

Fonte: adaptado de BBC Learning, (2015).

### 3.1.1 Decomposição

Para [Liukas \(2015\)](#), esse é processo no qual os problemas são decompostos em partes menores. Ela utiliza exemplos simples, como a decomposição de refeições, receitas culinárias e as fases que compõem um jogo. Trata-se de quebrar um problema ou sistema complexo em partes menores, que são mais fáceis de entender e resolver. Ainda, a autora afirma que programadores utilizam frequentemente essa técnica para dividir um algoritmo em pedaços menores, para facilitar sua compreensão e manutenção.

Segundo [Brackmann \(2017\)](#), quando a decomposição é aplicada a elementos físicos, como, por exemplo, consertar uma bicicleta através da decomposição de suas partes, a manutenção torna-se mais fácil.

Para [Csizmadia \(2015\)](#), essa técnica da decomposição possibilita resolver problemas complexos de forma mais simples, facilita a compreensão de novas situações e possibilita projetar sistemas de grande porte.

### 3.1.2 Reconhecimento de Padrões

Padrões são similaridades ou características que problemas compartilham e que podem ser exploradas para que os mesmos sejam solucionados de forma mais eficiente. [Liukas \(2015\)](#) define o Reconhecimento de Padrões como encontrar similaridades e padrões com o intuito de resolver problemas complexos de forma mais eficiente. Para isso, procura-se por elementos que sejam iguais ou muito similares em cada problema. Ao realizar-se a decomposição de um problema complexo, seguidamente são encontrados padrões entre os subproblemas gerados



pela decomposição (ver Figura 3.2). No exemplo é possível isolar as partes, mas é possível também pensar que a correia é um elemento de ligação entre a roda traseira e o pedal, para providenciar a tração, e que o guidom conecta o quadro à roda dianteira, para viabilizar a direção. Isso torna o exemplo mais próximo de um sistema complexo do mundo real.

O reconhecimento de padrões também pode ser chamado de automação. Automatizar é mecanizar as abstrações, e as relações entre níveis de abstrações. A automação é possível se são associados significados precisos às abstrações. Ela permite que algum tipo de máquina auxilie na solução dos problemas.



*Figura 3.2: Decomposição de problemas em subproblemas*

Fonte: (Brackmann 2017)

O Reconhecimento de Padrões é uma forma de resolver problemas rapidamente, fazendo uso de soluções previamente definidas em outros problemas e com base em experiências anteriores. Esses padrões podem trazer a definição dos dados, processos e estratégias que serão utilizados para resolver o problema.

Os professores, ao tratarem dessa etapa, podem realizar perguntas como:

- Este problema é similar a um outro que já tenhas resolvido? ou
- Em que pontos o atual problema é diferente de outro que já tenhas resolvido?

Ainda segundo Brackmann (2017), outro exemplo prático de como os professores podem conduzir esse processo é através da identificação de similaridades entre figuras (Figura 3.3). Note-se que as figuras das pessoas possuem características comuns, como, estão sorrindo, possuem cabelo encaracolado, ou diferenças, como usar chapéu. No Pensamento Computacional, essas características são chamadas de Padrões. No momento em que se consegue



um padrão, pode-se descrever outros, simplesmente seguindo o padrão e alterando as características, como, por exemplo: olhos pretos, cabelos presos, etc.

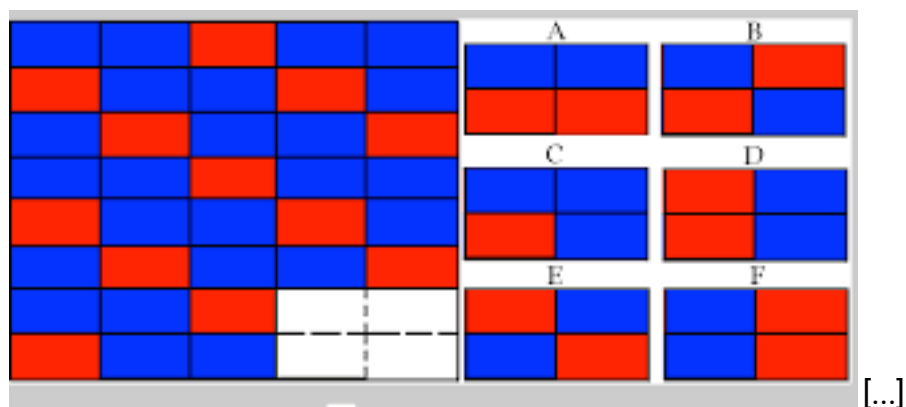


*Figura 3.3: Similaridade entre as figuras (Reconhecimento de Padrões)*

Fonte: Computational Think (Google)

Através do reconhecimento de padrões, é possível simplificar a solução de problemas e replicar essa solução em cada um dos subproblemas, caso haja semelhança. Quanto mais padrões consegue-se encontrar, mais rapidamente a macro solução é encontrada.

Outra forma de exemplificar o uso do reconhecimento de padrões pode ser vista na Figura 3.4. A sequência de padrões e cores pode ser decomposta em subpadrões (A, B, C, D, E e F).



*Figura 3.4: Sequência de padrões e subpadrões*

Fonte: Computational Think (Google)

### 3.1.3 Abstração

Esta etapa talvez seja a mais difícil de ser aplicada. Ela envolve a filtragem dos dados e sua classificação, desconsiderando elementos que não são essenciais e focando apenas nos que são relevantes. Através dessa técnica, consegue-se criar uma representação abstrata do que se quer resolver. O difícil é escolher o que será considerado irrelevante, para que o problema se torne mais fácil de ser compreendido, sem perder nenhuma informação relevante (CAS, 2014). Na programação, essa etapa significa eliminar variáveis cujo conteúdo possa ser representado por outras mais significativas.

De acordo com Wing (2006), a abstração é o conceito mais importante do PC, pois o processo de abstrair é utilizado em diversos momentos, tais como:

1. Na escrita do algoritmo e suas iterações;
2. Na seleção dos dados importantes;
3. Na escrita de uma pergunta;
4. Na natureza de um indivíduo em relação a um robô;
5. Na compreensão e organização de módulos em um sistema.

Do ponto de vista da Computação, na etapa da abstração, muitas vezes são necessárias diferentes formas de composição, permitindo trabalhar ideias potencialmente complexas a partir de ideias mais simples (Menezes et al, 1998):

- Composicionalidade vertical: composição de diferentes níveis de abstração, permitindo fluir do mais abstrato ao mais concreto;
- Composicionalidade horizontal: composição de partes componentes, usando combinadores como composição paralela/concorrente.

Em situações educacionais, uma forma de exemplificar diferentes formas de abstração seria via gramáticas de linguagens, as quais são essencialmente Gramáticas de Chomsky, as mesmas usadas em Computação (inclusive para definir algoritmo como alternativa à Máquina de Turing):

- composicionalidade vertical: a sucessiva aplicação de regras gramaticais partindo de uma ideia abstrata, gerando concretamente textos (sintáticos) de uma linguagem;
- composicionalidade horizontal: diferentes regras podem ser escolhidas em um dado momento (combinador não-determinístico), permitindo gerar concretamente diferentes textos (sintáticos) que satisfazem a mesma gramática.

Gramáticas de Chomsky <sup>23</sup> podem ser facilmente extrapoladas para gramáticas não-textuais, permitindo tratar imagens, animações, grafos, flexibilizando de forma ampla seu uso em abordagens educacionais.

---

<sup>23</sup> Fundamentalmente, uma gramática é composta por regras de produção, ou regras de reescrita, através das quais, partindo de um símbolo inicial e aplicando sucessivamente as regras, é possível obter todos os textos da linguagem. Hierarquia de Chomsky é a classificação de gramáticas formais

Uma forma de exemplificar abstrações é utilizar situações das histórias para abstrair informações que possam ser generalizadas (reconhecimento de padrões) tais como:

- Após ler algumas histórias infantis resumir o comportamento dos vilões.
- Quais os números sempre presentes na Figura 3.5?

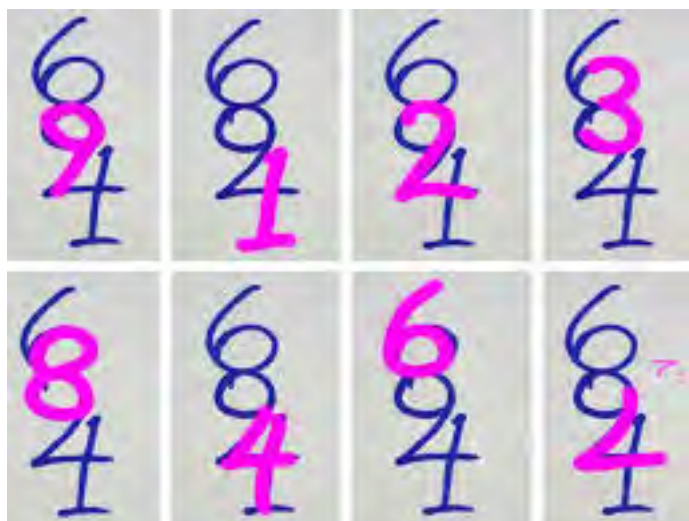


Figura 3.5: Exemplo de exercício de abstração

Fonte: Computational Think (Google)

### 3.1.4 Algoritmos

Algoritmos é o componente do PC que traz a programação para o domínio do PC, ou seja, repetir sequências de código. Conceitos e notações da matemática discreta (a Computação é discreta) são úteis para o estudo ou a expressão de objetos ou problemas em algoritmos e linguagens de programação.

Sendo assim, na Informática, a definição clássica diz que um algoritmo é uma abstração de um processo que recebe uma entrada que executa uma sequência finita de passos, e produz uma saída que satisfaça um objetivo específico. É necessário que cada passo seja executado em um tempo finito. A abstração proporciona um poder para escalonamento e tratamento da complexidade (Wing, 2010). Já para Csizmadia (2017), um algoritmo é um plano, uma estratégia ou um conjunto de instruções claras necessárias para a solução de um problema.

Projetar algoritmos eficientes, por natureza, envolve também a criação de tipos de dados abstratos. Um algoritmo pode ser transcrito para código de máquina

---

descrita, em 1959, pelo linguista Noam Chomsky. Essa classificação possui 4 níveis ou tipos hierárquicos (numerados de zero a três), expressando diferentes níveis de computacionalidade, sendo que o tipo zero possui o mesmo poder computacional que a Máquina de Turing.

através de uma linguagem de programação. Ou seja, ser entendido por uma máquina,

É possível elaborar perguntas que possam ajudar a compreensão de como gerar e quais as limitações dos algoritmos, tais como:

- É possível solucionar este problema utilizando algoritmos? ou
- Qual a precisão que se necessita para solucionar este problema?

Ao serem utilizados, seguirão sempre os mesmos passos pré-definidos, ou seja, podem ser repetidos quantas vezes forem necessários, para a solução de um mesmo problema. Por exemplo, calcular o contracheque de um certo número de funcionários de uma mesma empresa.

Os algoritmos, como são tratados pelo PC, são estáticos (possuem uma topologia estática – repetem sempre os mesmos procedimentos). A IA já mudou essa ideia há muito tempo com seus algoritmos evolutivos (topologia dinâmica), que se auto modificam e aprendem de diferentes maneiras. Por exemplo, o jogo Go, em sua versão *plugged*, da Google, primeiro aprendeu estratégias de jogo, com o conhecimento de jogadores humanos, mas, em um segundo momento, evoluiu e se aperfeiçoou jogando com outra máquina. Esse jogo pode não ser adequado para as séries iniciais do ensino fundamental, mas certamente o é para as séries finais.

Ainda, humanos pensam e visualizam espaços com múltiplas soluções e de forma não determinista. Computadores já conseguem simular, em parte, essa forma de pensamento, com o uso da programação paralela, de linguagens declarativas, típicas da IA e, mais recentemente, com técnicas avançadas da IA, como *deeplearning*. Mas, infelizmente, tanto na literatura como na prática, as implementações do PC, atuais, não utilizam essas possibilidades, nem para ensinar programação. Muitos autores entendem que ensinar paralelismo, busca em espaço de soluções e programação não determinística é complicado antes do ensino universitário. Mas, em termos de conceitos, isso é possível inclusive para crianças. O que falta são modelos e formas simples de como ensinar. Por exemplo, as crianças aprendem gramáticas. Essas podem ser usadas para estudar concorrência e não determinismo e são tão poderosas quanto a Máquina de Turing em termos de solucionabilidade de problemas. O jogo de xadrez é outro exemplo onde é possível se fazer opções por diferentes movimentos de peças (não-determinismo) que determinam diferentes estratégias de jogo. Entretanto, essas ideias quase não são discutidas na literatura. Não são encontrados jogos que desenvolvam esses tipos de raciocínios no âmbito do PC *unplugged*, também não são encontradas *tools* (plataformas de programação) que os incentivem, no âmbito do PC *plugged*. Na pesquisa bibliográfica, foram encontrados 3 (três) artigos sobre o tema, todos falando nesses princípios de forma teórica ou ideal.

Um exemplo de algoritmo clássico foi apresentado por Daniel Hillis (2014), em seu livro *The Pattern on the Stone* ("O Padrão Gravado na Pedra").

*"Quando eu era estudante de graduação [...] um dos meus colegas de quarto tinha várias dúzias de pares de meias, cada par de uma cor ou modelo ligeiramente diferentes. Como ele costumava adiar a lavagem até que nenhuma meia*

*estivesse limpa, toda vez que ele as lavava enfrentava uma tarefa nada desprezível de combina-las novamente em seus devidos pares. Eis como meu colega fazia isso:*

- Primeiro, ele puxava uma meia qualquer da pilha de meias lavadas,
- depois, tirava outra aleatoriamente, para ver se combinavam.
- Se não combinavam, ele coloca a segunda meia de volta e puxava outra, aleatoriamente.
- Ele continuava esse processo até que encontrasse um par que combinasse.
- Depois ele prosseguia, até que todos os pares estivessem formados.[...]

Um dia, ao trazer a cesta de roupa lavada de volta, ele anunciou "resolvi usar um algoritmo melhor para combinar minhas meias", O que ele quis dizer era que ia utilizar um procedimento de natureza fundamentalmente diferente.

- Ele tirou a primeira meia e colocou sobre a mesa.
- Tirou mais uma e comparou com a primeira meia;
- como não combinavam, colocou ao lado da outra. [...] Utilizando esse método ele conseguia combinar as meias em uma pequena fração do tempo do que era necessário no método anterior."

## 3.2 O pensamento PC e as ideias de Jean Piaget

Um trabalho de Jean Piaget especialmente interessante no contexto do PC é o livro *Morphisms and Categories* (Piaget 1992). Trata-se de uma publicação póstuma, inacabada, onde os manuscritos originais são do início dos anos 80 e a primeira publicação apenas ocorreu em 1990, na França. Dois pontos se destacam nesse trabalho: apresenta indícios sobre a partir de que idade o PC deve começar a ser desenvolvido e orienta como Teoria das Categorias (importante fundamento da Ciência da Computação) pode ser contemplada no PC.

Teoria das Categorias é um formalismo que pode ser visto como uma unificação de estruturas matemáticas, como uma estrutura matemática em si ou como formalização de estruturas matemáticas. Foi criada por S. Eilenberg e S. Mac Lane em 1945 e desde então tem influenciado muitas áreas de pesquisa. Em particular, a Ciência da Computação e a Teoria das Categorias têm se influenciado mutuamente. Um dos principais fatores é a expressividade da Teoria das Categorias, uma vez que o desenvolvimento de soluções para problemas está limitado pela própria capacidade do ser humano em expressar os problemas propostos. O uso de Teoria das Categorias tende a dar uma visão "geral" do problema, sem se preocupar com detalhes irrelevantes, com operações poderosas o suficiente para tratar o problema em alto nível.

Em seu trabalho, Piaget descreve os resultados que obteve em experiências com crianças, mostrando que a ideia de composição, típica da Teoria das Categorias, é geneticamente intuitiva para o homem. De fato, o homem

naturalmente tende a pensar em propriedades simples de esquemas, e então, sucessivamente, em operações similares em esquemas de esquemas, e assim por diante, formando uma cadeia de composições de propriedades. Com base nos resultados de Piaget, que não só a noção de composição é intuitiva, mas que o pensamento “geral” e “unificado”, característico de Teoria das Categorias, é também geneticamente intuitivo ao homem, pode-se reconhecê-lo no raciocínio infantil. Porém, por algum motivo, esse tipo de raciocínio é abafado durante a infância, e vai-se perdendo, tornando-se até antinatural com o decorrer do processo de aprendizagem.

Considerando que as entidades básicas da Teoria das Categorias são “morfismos” e “composição de morfismos”, tais podem ser instanciados, no contexto da Computação, como “ações” e “composição de ações”, gerando algoritmos (entre outros tantos exemplos). Assim, pode-se inferir que o homem é geneticamente computacional e, conseqüentemente, o PC é latente e tende a ser abafado no decorrer da escolaridade.

Trata-se também de uma evidência de que o PC deve ser tratado o quanto antes no desenvolvimento escolar.

## 4. Abordagem Pensamento Computacional *Unplugged*

Muitos aspectos da Computação podem ser ensinados sem o uso de computadores. A abordagem *unplugged* introduz conceitos de *hardware* e *software* que podem ser utilizados para introduzir conceitos de Computação a pessoas não técnicas e, principalmente para crianças, nos anos iniciais de sua formação. A adoção dessa metodologia é uma opção ou uma necessidade devido à escassez de recursos em países pobres. As aulas que adotam o PC podem ser dinâmicas planejadas, com atividades *unplugged*, que podem ocorrer através da aprendizagem cinestésica<sup>24</sup> (e.g. movimentar-se, usar cartões, recortar, dobrar, colar, desenhar, pintar, resolver enigmas e jogos) e com os estudantes trabalhando de forma colaborativa para aprender conceitos da Computação.

Normalmente, o PC é introduzido via aprendizagem colaborativa, por envolver atividades lúdicas, como jogos educacionais. Esse princípio já está presente no construtivismo. Trabalhar com objetos tangíveis do mundo real é central no construtivismo de Papert (Papert e Harel, 1991), conforme já colocado.

No presente texto apresentam-se opções de jogos já utilizados por autores que estudam o PC, mas também são apresentadas novas possibilidades vislumbradas pelos autores.

O uso de exemplos físicos e materiais escolares são comuns para simular o comportamento de máquinas. Quando se trata de salas de aula da educação básica, as ideias do PC podem ser encontradas a partir de 1997, quando Bell et al (1997) lançaram o protótipo do livro, em formato digital, *Computer Science Unplugged... Off-line activities and games for all ages*, para professores de todos os níveis escolares. A ideia foi muito bem recebida pelos professores e a academia, devido à qualidade do material publicado, e a *Association for Computing Machinery* (ACM) recomendou que as atividades contidas no livro passassem a integrar o currículo proposto pela *Computer Science Teachers Association* (CSTA) dos EUA (ACM - *Association for Computing Machinery*, 2003).

Pode-se exemplificar essas atividades através das ideias de Bell (2014), onde, entre outras propostas, é apresentado um jogo, cujo objetivo é completar um mapa de piratas. A ilustração da capa desse jogo é apresentada na Figura 4.1. Trata-se de um problema que pode ser caracterizado como um autômato finito, assim como os dois exemplos seguintes, e a atividade consiste em percorrer o espaço do jogo, na tentativa de encontrar um caminho para a ilha do tesouro. Outros exemplos também podem ser caracterizados como autômatos finitos e podem ter características mais educacionais que lúdicas, como os jogos de encontrar letras que compõem palavras presentes no enigma ou encontrar as palavras que compõem um jogo de

---

<sup>24</sup> Percepção dos movimentos musculares, peso e posição dos membros, por meio de estímulos próprios (Michaelis Moderno, 2017)



palavras cruzadas. Esses últimos podem ser utilizados em aulas de linguagem, pois servem para o ensino de várias línguas (ver Figura 4.2).



Figura 4.1: Jogo “Mapa do Pirata”

Fonte: Bell (2014)



Figura 4.2: Jogos para ensino de linguagem

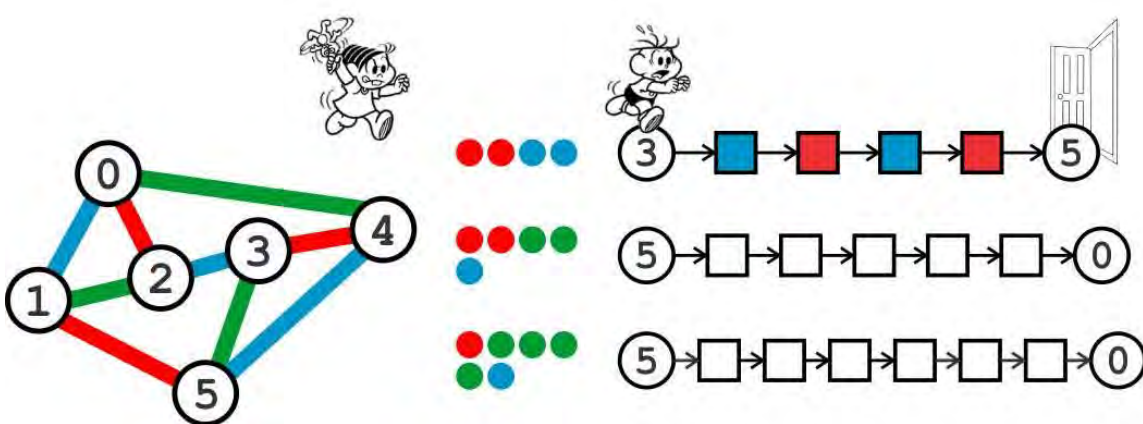
Fonte: Google (jogos pedagógicos)

Esse tipo de jogo existe também nas versões *plugged*, utilizando computadores ou robôs do tipo Lego.

Um exemplo particularmente interessante para o caso brasileiro é o proposto no trabalho de Brackmann (Brackmann 2017), onde são utilizados personagens das histórias em quadrinhos da “Turma da Mônica”. A proposta de Brackmann foi testada em escolas do Brasil e da Espanha, na forma de testes piloto, onde o autor realizou parte dos estudos de seu doutorado, em Informática na Educação. Na Figura 3.8 pode-se ver um exemplo que envolve, também, o conceito de autômatos finitos.



**Atividade 10: Autômatos da Mônica (Apêndice L, p. 215)**



**DESCRIÇÃO**

Material necessário:

- Duas folhas compostas por quatro mapas e dezesseis desafios (atividades)
- Um filete de papel com quadrados coloridos, um círculo com o rosto dos personagens e uma porta. Objetivo: exercitar os pilares de abstração, decomposição, reconhecimento de padrão e algoritmos através da resolução de autômatos finitos determinísticos representados por um diagrama similar ao de transição de estados. Instruções:

Entregar uma folha para cada estudante;

Pedir para que recortem os quadrados e usem para colar no local indicado conforme a solução (rota) encontrada;

Para iniciar um desafio, deve-se posicionar a peça do Cebolinha no nodo (número) do mapa indicado no lado direito. Em seguida, utilizando todas as cores que constam do lado esquerdo da rota, encontrar o caminho que utilize todas as cores indicadas.

Nenhum quadrado deve ficar vazio.

Figura 4.3: Autômatos Finitos da Turma da Mônica

Fonte: Brackmann 2017

Outro exemplo é vestir um personagem com roupas para diferentes ocasiões (Liukas, 2015), realizando uma referência às condições que são impostas dentro de um código (*if/else*), conforme Figura 4.4.

Esse tipo de jogo é bem comum no Brasil e encontra-se, principalmente, com personagens femininas, que podem ser vestidas para diferentes situações (praia, festa, etc.).

O autor possui uma série de 4 (quatro) livros para crianças chamados de: (1) *Computational Thinking*, publicado em 2015, (2) *Computer*, publicado em 2016, (3) *Internet*, publicado em 2017 e (4) *Artificial Intelligence*, previsto para 2018. Além do

material *unplugged* o autor desenvolve também ambientes educacionais com situações de problemas envolvendo o PC.



Figura 4.4: Vestindo Personagens

Fonte: sítio do livro *Computational Thinking* Liukas (2015)

A Figura 4.5 apresenta uma atividade que pode ser utilizada para o ensino de números binários. O jogo é composto por cartas, que têm, na parte dianteira, indicados 1, 2, 4, 8... 16 pontos, cada uma, e são pretas na parte traseira. As cartas são colocadas lado a lado, iniciando da esquerda para a direita, sucessivamente até a última carta, à direita, com apenas um ponto. Cada carta que esteja com sua face para cima, representa o número 1, binário, e as cartas que estão viradas para baixo representam o 0 (zero).

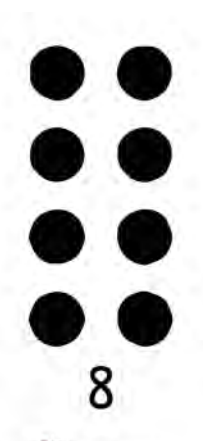


Figura 4.5: Baralho para ensino de Números Binários

Fonte: <https://csunplugged.org/en/resources/binary-cards/> - acesso em 4/12/2018

Esse exemplo ilustra como materiais para o PC *unplugged* podem ser impressos a partir de conteúdos digitais. Essa é a tendência atual para esse tipo de conteúdo.

## 4.1 Jogos para PC *Unplugged*

A seguir serão apresentados, alguns jogos voltadas para o trabalho com as ideias propostas pelo PC *unplugged*.

### 4.1.1 Jogo *Haathi Mera Saathi* (Meu Amigo Elefante)

O trabalho de [Unnikrishnan et al \(2016\)](#) é um relato do uso do PC *unplugged* em áreas rurais e desprivilegiadas da Índia. Devido ao fato do elefante ser um animal bastante comum e respeitado na região, os pesquisadores propuseram uma atividade para crianças, envolvendo esse personagem. O jogo denominado *Haathi Mera Saathi* (Meu Amigo Elefante) é um jogo de tabuleiro, que envolve uma miniatura de elefante e fichas com desenhos de bananas e árvores. As crianças usam as cartas com instruções para que o elefante visite todos os quadrados onde existe uma banana. A Figura 4.6 apresenta o tabuleiro com algumas cartas utilizadas no jogo.

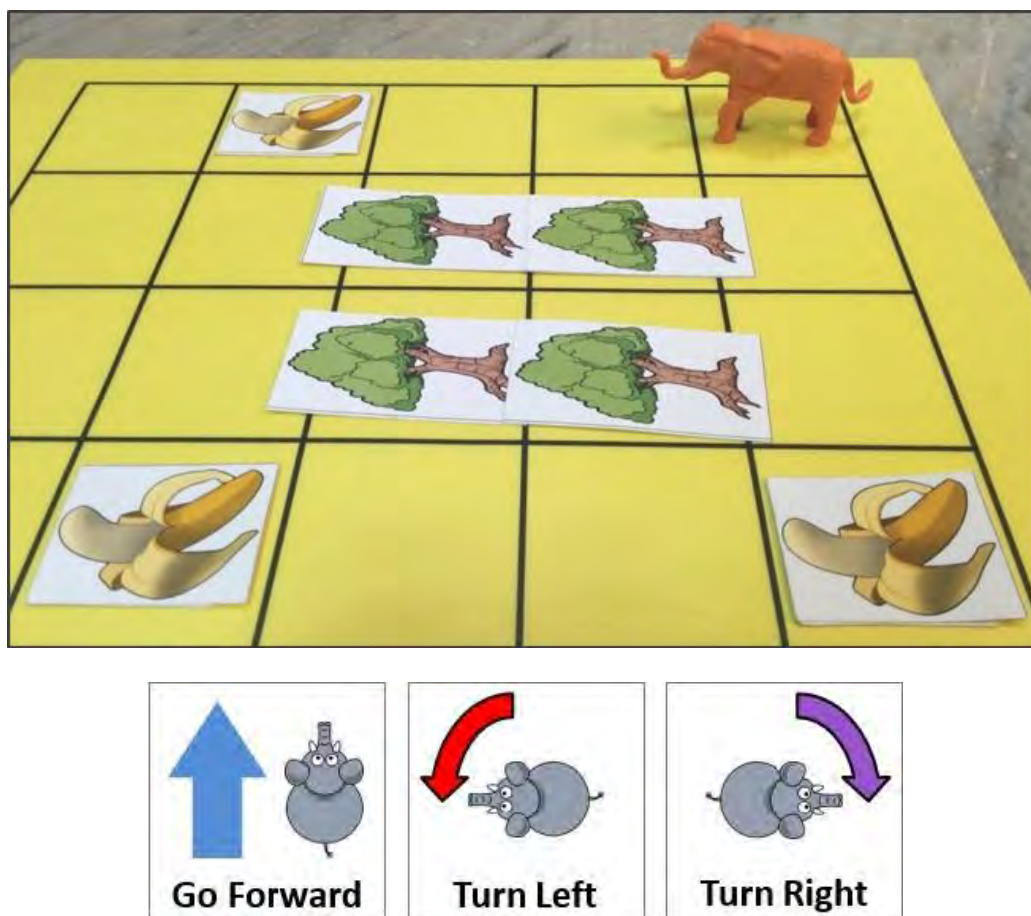


Figura 4.6: Tabuleiro e cartas do jogo *Haathi Mera Saathi*

Fonte: *My Elephant Friend* [Unnikrishnan et al \(2016\)](#)

As cartas com as instruções são posicionadas em outro tabuleiro (Tabuleiro de Programação), com a sequência de passos para atingir o objetivo. Na Figura 4.7 é exemplificada a resolução do desafio proposto na Figura 4.6 de duas formas: com e sem a utilização de iterações.



Figura 4.7: Tabuleiro de Programação do Jogo Haathi Mera Saathi

Fonte: My Elephant Friend Unnikrishnan et al (2016)

### 4.1.2 Jogo Robot Rally

Outros jogos similares, que não necessitam do uso de computadores, para exercitar o PC *unplugged*, existem no mercado desde 1994, como, por exemplo, o *Robot Rally*. O jogo é composto por um tabuleiro e cartas de instruções para os robôs que são os personagens principais. Com as cartas é possível mover os personagens no tabuleiro de tal forma que eles consigam visitar todas as casas (pontos de verificação) indicadas. As regras do jogo incluem cartas de atualização do robô, conceitos de variáveis, consumo de energia, nível de dano, perda de memória (*reboot*) e a criação de estratégias para completar os objetivos do jogo e ainda causar danos nos robôs dos demais competidores. Tendo em vista que o jogo possui uma alta complexidade, ele é recomendado para crianças com doze anos ou mais e suporta de 2 a 6 jogadores em sua última edição de 2015. Devido ao seu constante sucesso em vendas, diversas edições já foram lançadas (ver Figura 4.8).





Figura 4.8: Jogo Robot Rally (edição 2015)

Fonte: Garfield e Hasbro Gaming (2015)

### 4.1.3 Jogo Bits & Bytes

O jogo *Bits & Bytes* é comercializado pela Games4Learning (2014). Trata-se de um jogo de cartas que aborda os temas de lógica, resolução de problemas e pensamento crítico. Para iniciar, as cartas são colocadas de forma organizada e viradas para baixo em cima de uma mesa, formando um tabuleiro, onde os jogadores deverão explorar o espaço, desvirando as cartas, controlando um dos personagens do jogo, com outras cartas de instruções. O objetivo principal é chegar ao centro do tabuleiro, onde se localiza o planeta dos personagens, chamado de RAM. Recomendado para crianças a partir de quatro anos (ver Figura 4.9).



Figura 4.9: Jogo “Bits & Bytes”

Fonte: Games4Learning (2014)

#### 4.1.4 Jogo *Littlecodr*

O jogo *Littlecodr* foi desenvolvido por Greenhill e Slee (2015). É composto por 80 (oitenta) cartas com instruções e 20 (vinte) contendo sugestões de aplicações ou desafios. O jogo pode ser utilizado em diversas situações genéricas, ou seja, não há a necessidade de um tabuleiro. Esse jogo é indicado para crianças de 4 a 8 anos e trabalha com conceitos de codificação, lógica, planejamento, pensamento sequencial, prototipação, solução de problemas, além de ensinar a contar, ler, seguir instruções e distinguir esquerda e direita. Um exemplo de aplicação, para desviar de uma mesa e um sofá, encontra-se na Figura 4.10.

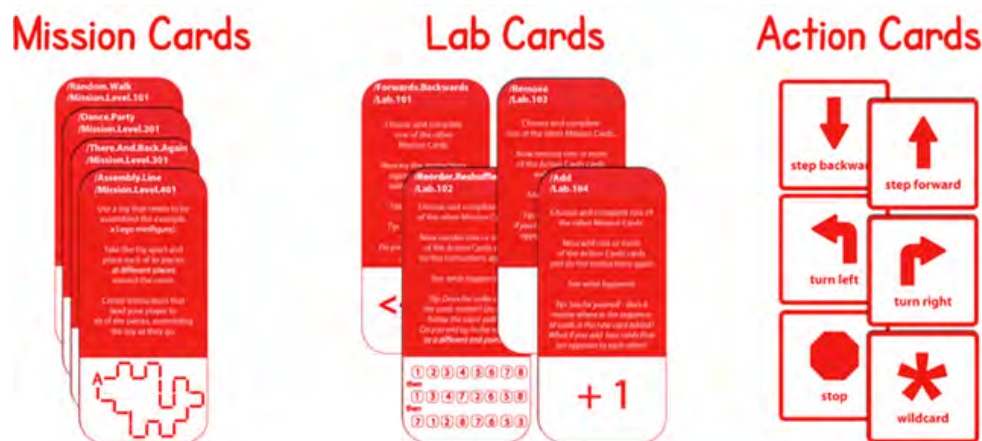


Figura 4.10: Execução de um programa no jogo “Littlecodr”

Fonte: Greenhill e Slee (2015)

É possível observar que as regras dos jogos estão voltadas para movimentos em um espaço pré-definido e com objetivos muito similares.

#### 4.1.5 Jogo *Giggle Chip*

O jogo *Giggle Chip* é colaborativo, pois envolve cartas. Foi desenvolvido por Ellis e Charlotte (2017) e está voltado para a alfabetização e para a introdução de números binários. Ao invés de usar palavras com animais (A para Abelha, B para Boi, C para Cavalo, etc.), usam-se termos voltados para a área da tecnologia e Computação, tais como *App*<sup>25</sup>, *Blog*, *Coding*, *Digital*, etc. Na parte traseira da carta há também outros dois jogos, sendo o primeiro de conversão de números binários e o segundo de perguntas e respostas. O jogo é recomendado para crianças de cinco a nove anos e está disponível somente em língua inglesa. As partes frontal e traseira das cartas encontram-se na Figura 4.11, frente e verso, respectivamente.

<sup>25</sup> *Apps* é a abreviação da palavra *applications*, ou aplicativos. No contexto dos *smartphones*, *app* é um programa que se pode instalar em um celular.

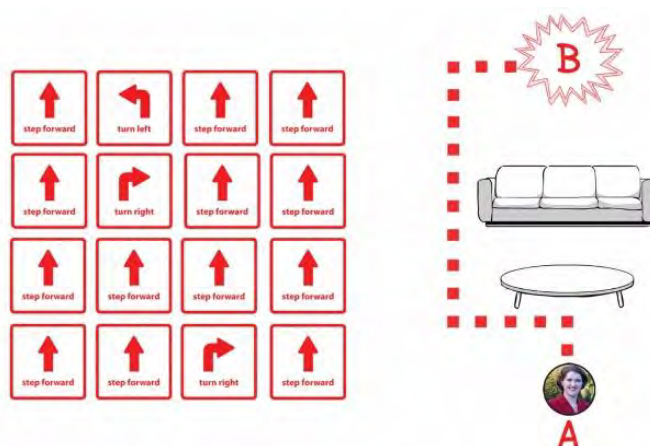


Figura 4.11 Jogo Gigggle Chips: frente (esquerda) e verso (direita)

Fonte: Ellis e Charlotte (2017)

#### 4.1.6 Jogo Chocolate Fix

Da mesma forma, o *Chocolate Fix* é um jogo desenvolvido por Engelberg (Engelberg; Thinkfun, 2012), composto por um tabuleiro com seis reentrâncias para encaixe de peças que lembram chocolates e um bloco com 40 (quarenta) desafios. Cada peça possui uma cor e um formato que não se repetem. Essas devem ser posicionadas conforme pistas que constam no desafio, onde o jogador precisa realizar o reconhecimento de padrões e criar estratégias para que consiga posicionar todos os doces de uma forma que atenda a todos os padrões, os quais fazem parte do desafio. O jogo pode ser jogado individualmente ou com mais pessoas. É indicado para um público a partir de 8 anos de idade. Um exemplo de desafio é apresentado na Figura 4.12.



Figura 4.12: Jogo Chocolate Fix

Fonte: Engelberg; ThinkFun (2012)

#### 4.1.7 Jogos *Circuit Maze* e *Laser Maze*

O jogo *Circuit Maze* (Yakos; Thinkfun, 2015) é composto por um tabuleiro com 25 (vinte e cinco) reentrâncias, para encaixe de peças que podem, ou não, conduzir eletricidade. Cada peça possui uma característica, como: condutor em linha reta, em curva, pontes, transversal, interruptor ou lâmpada. Com essas peças é possível simular circuitos (lógica booleana), criar estratégias para a resolução de 60 (sessenta) problemas propostos, além de exercitar atividades relacionadas às disciplinas de Ciências e Física. O jogo *Laser Maze* (Hooper; Thinkfun, 2012) segue a mesma linha de raciocínio, porém utiliza um raio laser e espelhos para replicar a luz, a fim de atingir o alvo pré-definido no desafio. Ambos os jogos são recomendados para crianças a partir de 8 anos e podem ser jogados de forma individual ou coletiva (ver Figura 4.13).



Figura 4.13: Jogo *Circuit Maze* e *Laser Maze*

Fonte: Yakos; Thinkfun, 2015 e Hooper; Thinkfun, 2012

#### 4.1.8 Jogo *Code Master*

*Code Master*, o jogo criado por Engelberg e Thinkfun (Engelberg; Thinkfun, 2015), é composto por dois tabuleiros: o primeiro possui dez mapas que lembram autômatos, que possuem ligações de diferentes cores entre os nodos, e o segundo tabuleiro serve para o jogador registrar o trajeto que optou por seguir, para chegar de um ponto a outro (ver Figura 4.14). Após o jogador escolher um dos 60 níveis de dificuldade, ele deve posicionar o personagem e o portal de saída nos nodos indicados, abrir o segundo tabuleiro na página referente ao desafio e utilizar a quantidade de fichas coloridas definidas no exercício. O objetivo do jogo é chegar até o portal, utilizando todas as fichas coloridas. O grau de dificuldade aumenta no momento em que há a necessidade de coletar cristais em certos nodos do mapa. Esse jogo foi projetado para ensinar diversas características computacionais, sendo algumas delas: abstração, resolução de problema, repetição (*looping*), reconhecimento de padrões, decomposição, condicionais e, sem esquecer, o algoritmo.





Figura 4.14: Jogo Code Master (tabuleiros e peças)

Fonte: Engelberg; Thinkfun, 2015

#### 4.1.9 Jogo Code Monkey Island

O jogo *Code Monkey Island*, criado por Sidhu (2014), é composto por um tabuleiro, 80 (oitenta) cartas e 12 (doze) miniaturas de macacos. O jogo possui regras muito parecidas com o tradicional “Ludo”, porém sem o uso de dados. Os dados são substituídos por instruções e condições escritas nas cartas. De acordo com o autor, o jogo exercita declarações condicionais, repetições, operadores e lógica booleana, execução de tarefas e estrutura de dados. Seu uso é aconselhado a partir de 8 anos. Uma imagem do jogo e das cartelas encontram-se nas Figuras 3.20 e 3.21, respectivamente.



Figura 4.15: Visão do Jogo Code Monkey Island

Fonte: <https://technical.ly/brooklyn/2014/05/08/board-game-teaches-kids-coding-concepts-monkeys-crowdfunding/> - acesso em 4/12/2018

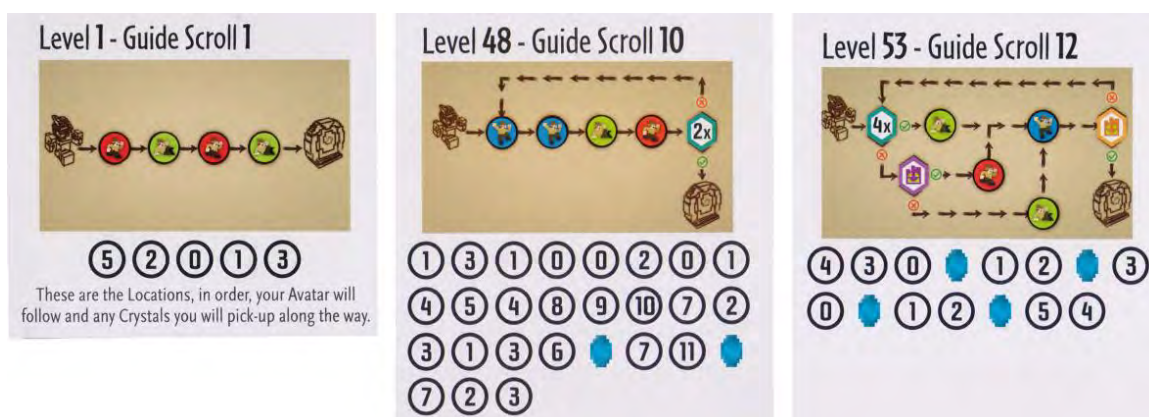


Figura 4.16: Jogo Code Monkey Island - Cartelas

Fonte: Sidhu (2014)

#### 4.1.10 Jogo Coding Is Good

*CodingIsGood* é um jogo de cartas para dois jogadores, desenvolvido por MathAndCoding (2017). Na parte frontal da carta, é apresentado ao jogador um algoritmo em Python e, na parte traseira, a resposta que deve ser visualizada pelo oponente. O jogador deve, então, "executar" o código mentalmente e dizer em voz alta a resposta. Caso a resposta esteja correta, a carta é colocada em uma pilha, senão é descartada. O jogador com a maior pilha é o vencedor da partida. Existem três níveis de dificuldade: básico, intermediário e avançado. Recomendado para crianças a partir de 10 anos. O Jogo possui versões em tabuleiro e computacional, como se pode ver na Figura 4.17, sendo que nessa última pode-se ver que também existe versão *plugged* do jogo.



Figura 4.17: Tabuleiros do jogo Coding Is Good

Fonte: MathAndCoding (2017)

O jogo *CodingFarmer* foi desenvolvido por MathAndCoding (2015), onde até quatro jogadores colocam uma peça no formato de um trator no início do caminho e, através de comandos em inglês e na linguagem de programação Java, que constam nas cartas, devem guiá-lo até o celeiro. Algumas casas (posições) no

trajeto podem influenciar o código a ser executado. Recomendado para jogadores a partir de sete anos. O tabuleiro pode ser visto na Figura 4.18.



Figura 4.18: Tabuleiro do jogo CodingFarmer

Fonte: MathAndCoding (2015)

## 4.2 Livros que trazem jogos e atividades vinculadas ao PC *unplugged*

Além dos jogos isolados, existem alguns livros que trabalham com PC *unplugged*. Cada um deles possui uma estratégia diferente para abordar os conceitos que envolvem o PC. A seguir são apresentados alguns livros encontrados na literatura internacional, que podem ou não possuir versões para o português. Com referência aos livros dedicados para o PC *unplugged*, ao contrário do que acontece com o PC *plugged*, não foram encontradas publicações recentes (2017-2018).

O foco está na produção de sítios, como o da [classic.csunplugged.org](http://classic.csunplugged.org) – acesso em 9/12/2018, que apresentam um conjunto de materiais, organizados por idade e que tratam dos temas: O que é Ciência da Computação e Como posso ensinar Ciência da Computação *unplugged*. Nos sítios existe um conjunto de materiais ilustrados (tabuleiros, cartas, fichas, etc.), que precisam ser impressos pelos professores ou escolas, para cada aluno ou grupo de alunos.

### 4.2.1 Livro *Computer Science Unplugged... off-line activities and games for all ages*

*Computer Science Unplugged...off-line activities and games for all ages* trata-se de um livro, de 2011, traduzido para o português com o título de “Ensinando Ciência da Computação sem o Uso do Computador” (Figura 4.19). Os autores são Tim Bell, Ian H. Witten e Mike Fellows e esse livro foi adaptado para a sala de aula

por Robyn Adams e Jane McKenzie e traduzido para o português em tradução coordenada por Luciano Porto Barreto. Trata-se de um manual que apresenta sugestões de atividades para o uso do PC em sala de aula, na sua forma *unplugged*. Ou seja, sua proposta é diferente dos demais aqui apresentados, pois não está focado em um jogo em particular. São feitas sugestões de inclusão do PC em várias disciplinas em diferentes níveis do ensino. O livro está disponível de forma gratuita no [sítio: https://classic.csunplugged.org/wp-content/uploads/2014/12/CSUnpluggedTeachers-portuguese-brazil-feb-2011.pdf](https://classic.csunplugged.org/wp-content/uploads/2014/12/CSUnpluggedTeachers-portuguese-brazil-feb-2011.pdf) - acesso em 4/12/2018.



Figura 4.19: Capa do livro *Ensinando Ciência da Computação sem o uso do Computador*

Fonte: Tim Bell, Ian H. Witten e Mike Fellows (2011)

#### 4.2.2 Livro *Teaching Thinking Across the Curriculum*

O livro *Teaching Thinking Across the Curriculum*, publicado por Ryan (1987), apresentado na Figura 4.20, é um dos primeiros textos que aborda o tema sob o ponto de vista do PC, isso mesmo antes do termo ser utilizado. Nele é contada uma das histórias de Sherlock Holmes que, através de sua experiência como detetive, busca solucionar um crime. Tendo como pano de fundo a história de Sherlock Holmes, o livro aborda temas como: variáveis, constantes, números randômicos, condicionais, repetições, banco de dados, indexação, entre outros. Na Figura 4.21, é apresentada a ficha utilizada para que o leitor do livro possa apontar as características do seu personagem, tais como: nome, habilidades, equipamentos e dinheiro em posse, apontamentos que se façam necessários e uma relação de



pistas que são encontradas durante a aventura. A experiência é similar a jogos de RPG<sup>26</sup>, porém faz uso de instruções e condições bem claras, como por exemplo: ir até uma página específica de acordo com a ação que o leitor deseja realizar. Ou então o leitor pode seguir até uma seção do livro, caso tenha uma pista necessária (ver Figura 4.22). O livro foi publicado no Brasil em 1994 (Ryan, 1994).

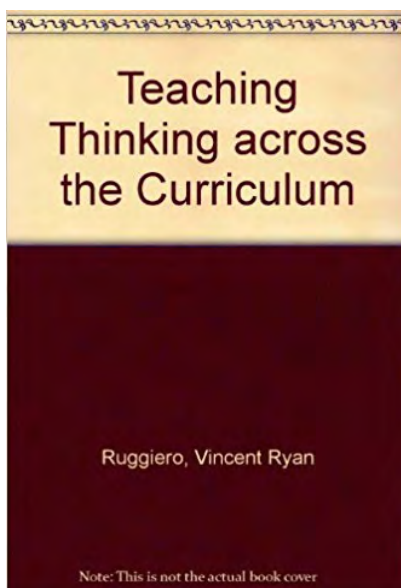


Figura 4.20: Livro Teaching Thinking Across the Curriculum

Fonte: sítio onde o livro foi encontrado á venda (2018)



Figura 4.21: Fichas do Personagem e Pistas

Fonte: Ryan (1994)

<sup>26</sup> A sigla *RPG*, oriunda da expressão em inglês *Role Playing Game*, define um estilo de jogo em que as pessoas interpretam seus personagens, criando narrativas, histórias e um enredo, guiadas por uma delas, que geralmente leva o nome de mestre do jogo.

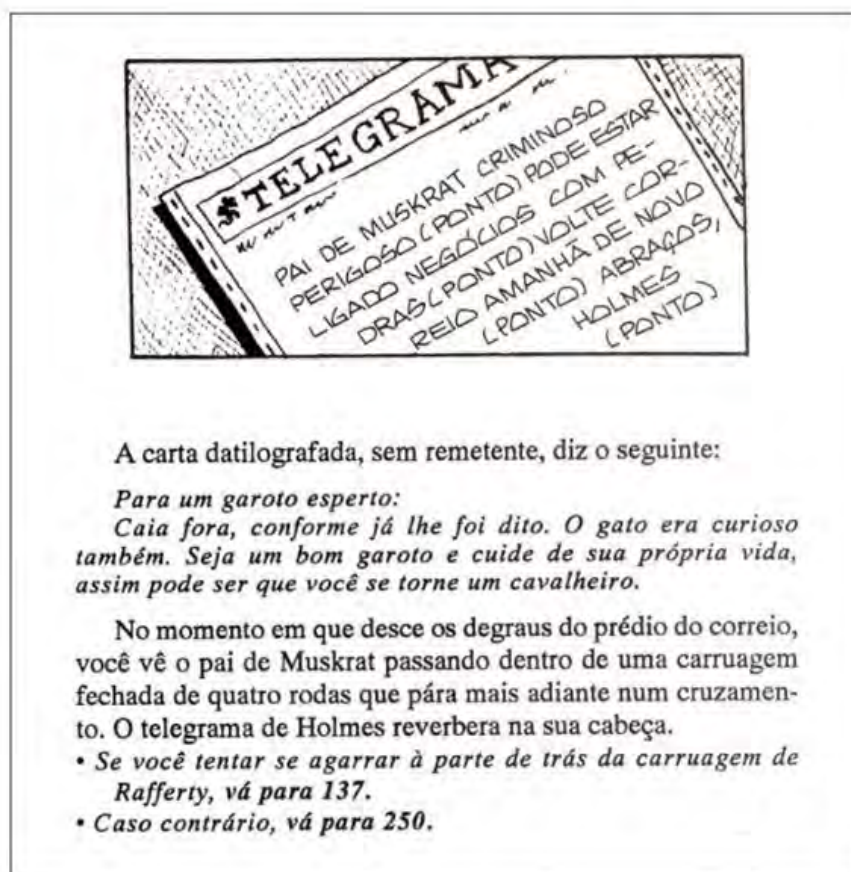


Figura 4.22: Exemplo de condicional utilizando pistas na história de Sherlock Holmes

Fonte: Ryan (1994)

Existe um sítio com sugestões de como ensinar Computação, utilizando-se o livro <http://www.nln.org/calendar/event-details/2018/06/04/default-calendar/how-to-teach-thinking-across-the-curriculum> - acesso em 4/12/2018. As figuras em português foram retiradas da versão traduzida do livro.

### 4.2.3 Livro *Lift-the-Flap Computers and Coding*

O livro *Lift-the-Flap Computers and Coding*, publicado por Dickins e Nielsen (2015) foi originalmente lançado na Inglaterra, para ensinar crianças a partir de 7 (sete) anos, conceitos da Computação. Trata-se de um livro pequeno (16 páginas). Através do uso de abas ensina o que é um computador e o que há dentro dele, o significado de codificar, como executar instruções, o que são as linguagens de programação, como criar um código eficiente, como a Internet funciona e um breve histórico da Computação. Não há até o momento uma tradução do livro para o português. Um exemplo de ilustração do livro é mostrado na Figura 4.23.

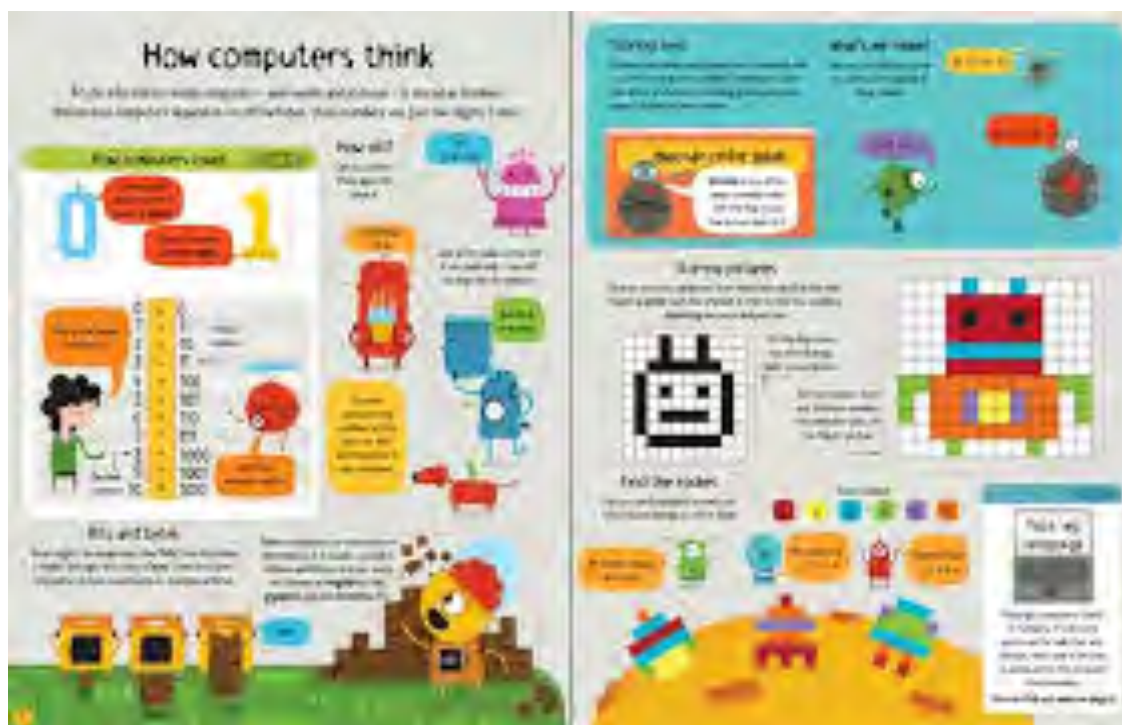


Figura 4.23: Exemplo de atividade do livro *Lift-the-Flap Computers and Coding*

Fonte: Dickins e Nielsen (2015)

#### 4.2.4 Livro *CodyRoby*

Apesar do nome, o livro *CodyRoby*, escrito por Bogliolo (2016), destina-se ao PC *unplugged* e explica as regras de um jogo de tabuleiro cujo objetivo é controlar um robô com a utilização de três instruções básicas (andar para a frente, girar para a direita e girar para a esquerda), para atingir um objetivo. O que o diferencia dos demais jogos é a possibilidade de recortar os tabuleiros, cartas e demais peças necessárias da última seção do livro, que está em um sítio<sup>27</sup> (é uma tendência que os livros para o PC *unplugged* utilizem, cada vez mais, o formato digital). O livro possui um sítio em italiano, com MOOC (*Massive Open Courses*) para ensino de como utilizá-lo, escolas de capacitação, exemplos de aplicação, fórum de debates, etc. O objetivo é o ensino de programação de forma *unplugged* (no acesso em 1/4 2018, o link do livro não se encontrava disponível). Outro ponto a destacar é que embora existam autores italianos importantes sobre o uso do PC, não foram encontrados documentos que tratem como o tema está sendo abordado na Itália, de forma geral, pois naquele país os professores possuem total liberdade pedagógica (ver Figura 3.29).

<sup>27</sup> <http://codemooc.org/codyroby/>

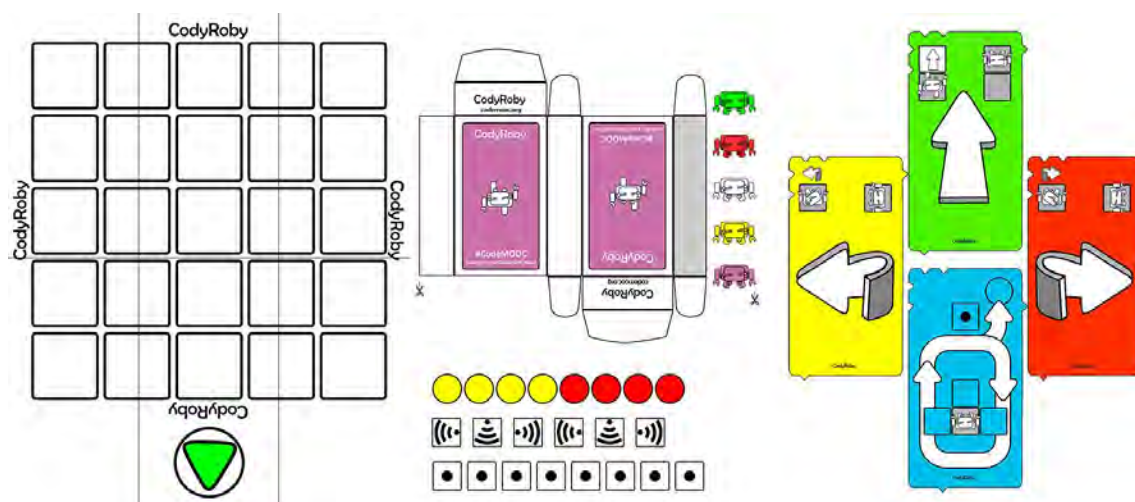


Figura 4.24: Peças do livro CodyRoby

Fonte: Bogliolo (2016)

### 4.3 Considerações sobre os Materiais Utilizados no PC *Unplugged*

Como apresentado, existem diversas opções de aplicação dos fundamentos da Computação, em um formato *unplugged*, porém não se pode deixar de citar também alguns jogos tradicionais que possuem objetivos de exercitar as aptidões de resolução de problemas e estratégia: Xadrez, Go, Gamão, entre outros. Nota-se que os jogos com um foco em fundamentos da Computação tiveram grande proliferação nos últimos três anos, comprovando a alta demanda desse tipo de aptidão.

Outro contexto em que o PC não-determinístico pode ser explorado, tanto em jogos como em plataformas, é o da linguagem, ou seja, através da utilização de linguagens livres do contexto. A IA implementa essas linguagens, através da linguagem de programação declarativa Prolog (Programação em lógica), desde a década de 80. Essas linguagens são gramáticas de Chomsky (ou derivações diretas). Como sua versão *plugged* é feita em máquinas determinísticas, o algoritmo possui uma descrição finita, descrito em regras do tipo Se-Então, que são aplicadas de forma determinística, mas que mesmo assim, permitem análise sintática, semântica e pragmática, e também construir diversos textos. Linguagens de programação declarativas são utilizadas também para a Música, onde realizam as mesmas funções realizadas em um texto. Esse exemplo poderia também possuir uma versão *unplugged*, através de um jogo que fizesse uso de gramática em sentido amplo. As gramáticas podem também ter aplicações na área da Ciência, como as gramáticas de grafos, que podem descrever questões de Biologia (ex: reprodução de células), Química (reações químicas), ou na Música (evolução/construção de



uma música). No entanto, não foram encontrados livros ou materiais que tratem explicitamente desse tema, na forma *unplugged*.

Em revisões bibliográficas, que tratam da aplicação do PC realizadas por Kalelioglu *et al* (2016) e Lye e Koh (2014) professores afirmam estar usando essa metodologia, de alguma forma, em suas aulas. Principalmente, para promover o desenvolvimento das aptidões de programação em computadores. O mesmo não ocorre quando se trata da abordagem *unplugged*. A maioria das pesquisas que utilizam atividades *unplugged* visa apenas promover o interesse dos estudantes pela Computação. Provavelmente isso acontece porque poucos países assumem o uso da abordagem do PC *unplugged* e os que o fazem a utilizam apenas nas séries iniciais do ensino (por exemplo, a Argentina e a Índia, em áreas remotas). E, também por esses países não possuírem tradição em pesquisa científica.

## 5. Abordagem Pensamento Computacional *Plugged*

A seguir serão apresentadas, de forma resumida, algumas plataformas computacionais voltadas para o trabalho com as ideias propostas pelo PC conectado ou *plugged*, embora esse não seja o foco principal deste trabalho e, portanto, as plataformas não serão descritas ou apresentadas com maiores detalhes. De forma geral, as plataformas para o PC *plugged* surgiram na década de 2000.

Dentre a série de plataformas e/ou linguagens voltadas para o PC *plugged*, optou-se por trazer algumas delas que possuem características especiais, como vinculação com Logo, ou especificidades para trabalhar projetos de artes visuais, ou características de IA e introduzem a ideia de algoritmos dinâmicos (isso não significa que algoritmos dinâmicos existam apenas no âmbito da IA, a transformação de grafos a partir de gramáticas de grafos é, também, outro exemplo), ou ainda, são linguagens muito próximas de outras, utilizadas em larga escala, como é o caso da linguagem Java. Ainda, foram selecionadas ferramentas que operam em mais de um sistema operacional e que permitem exportar as aplicações desenvolvidas para dispositivos móveis, ou mesmo se comunicam com robôs físicos. Essas plataformas também foram citadas como as mais utilizadas em um estudo realizado por Mannila *et al* (2014), que apresenta os resultados de um questionário *on-line*, para professores de diversos países da UE. Um dos objetivos do questionário foi identificar as tecnologias utilizadas pelos professores quando trabalhavam com os conceitos do PC em sala de aula. Com base nas respostas fornecidas por 961 (novecentos e sessenta e um) professores, em grande parte do ensino fundamental, o estudo apresenta uma análise estatística sobre os tipos de ferramentas utilizadas (recursos da Web, linguagens, robótica, simulação, etc.). As mais utilizadas são apresentadas a seguir.

Igualmente serão apresentados alguns jogos e livros utilizados para o PC *plugged*.

### 5.1 Plataformas Voltadas para PC *Plugged*

A seguir serão apresentados ambientes utilizados para o ensino de Programação. O ensino de programação é a abordagem mais utilizada no PC *plugged*. A utilização de plataformas digitais está diretamente relacionada com a fluência digital. Segundo documento da SBC (SBC 2018a), essa fluência pode ser definida como a habilidade de encontrar, avaliar, produzir e comunicar informação, usando plataformas digitais (englobando tanto diferentes dispositivos de hardware quanto de software). Refere-se, por exemplo, ao uso de computadores e aplicativos, como software para formatar textos, produzir apresentações, buscar informações e insumos na Internet, etc.

### 5.1.1 Ambiente de Programação Scratch

O Scratch (<http://scratch.mit.edu> – acesso em 4/12/2018), é um ambiente de programação multimídia em rede, provavelmente o mais utilizado no momento, desenvolvido com a finalidade de aumentar a fluência digital de jovens e crianças. O ambiente está disponível para vários sistemas operacionais, como Linux, Microsoft Windows e Apple Mac OS X. O Scratch opera de forma integrada com uma comunidade *on-line*, com projetos compartilhados pelos próprios usuários, tutoriais multimídia, guias de primeiros passos para os aprendizes e uma base de informações de apoio aos professores que desejarem utilizar o aplicativo. Tanto o sítio quanto o conteúdo de apoio didático estão disponíveis em diversos idiomas, inclusive o português. O Scratch é uma ferramenta essencialmente pedagógica, relacionada conceitualmente com a linguagem Logo e outras iniciativas *construcionistas*. Foi planejada de forma a explorar formalmente os conceitos e práticas do construtivismo. (influências do construtivismo no PC também são encontradas em jogos *unplugged*, como o Jogo Mapa do Pirata).

A produção de imagens, no ambiente Scratch, pode ser animada ou estática. O ambiente também pode ser utilizado em projetos de computação com os *kits* de robótica Lego Mindstorms NXT<sup>28</sup>, placas controladoras Arduino<sup>29</sup> e PicoCricket<sup>30</sup>.

### 5.1.2 Ambiente de Programação App Inventor

Criado pelo Google e atualmente mantido pelo MIT *Center for Mobile Learning*, a *App Inventor*<sup>31</sup> é uma linguagem de programação visual dedicada ao projeto de aplicativos para celulares Android (Gray et al, 2012). A *App Inventor* é executada a partir de um navegador da Web. A única dependência externa é o *Java Runtime Environment* – JRE, precisando apenas ser instalado no Microsoft Windows. O computador precisa estar conectado à Internet para utilizar a *App Inventor*, apesar dos aplicativos desenvolvidos poderem ser utilizados no modo *off-line*. O projeto pode ser exportado para celular.

Assim como no ambiente Scratch, os elementos primitivos da linguagem são blocos organizados conforme sua função – declaração de variáveis, tratamento de texto, listas, funções matemáticas, operadores lógicos, estruturas de controle e definições de cores – que são combinados por meio de encaixes particulares. A comunicação com os *kits* de robótica Lego Mindstorms NXT ou controladoras Arduino dá-se através do *Bluetooth*.

Como tutoriais, a ferramenta possui um conteúdo organizado em três áreas: 1) *Teach*, direcionado a educadores; 2) *Explore*, orientado a aprendizes; e 3) *Invent*, que carrega o ambiente de programação. Não foram encontrados conteúdos

---

<sup>28</sup> <http://www.mindstorms.com> - acesso em 6/12/2018.

<sup>29</sup> <http://www.arduino.cc> - acesso em 6/12/2018.

<sup>30</sup> <http://www.picocricket.com/> - acesso em 6/12/2018.

<sup>31</sup> <http://appinventor.mit.edu/> - acesso em 6/12/2018.

oficiais em português para a *App Inventor*, no entanto existem *blogs* brasileiros e vídeos que oferecem introduções à linguagem.

### 5.1.3 Ambiente de Programação *Context Free*

Criado por Chris Coyne, Mark Lentczner e John Horigan (Chrstensen *at al*, 2009), *Context Free*<sup>32</sup> é um ambiente para a edição, rasterização e exploração de conjuntos de regras não determinísticas, para a produção de imagens usando a linguagem CFDG (*Context Free Design Grammar*). Essa linguagem é similar a uma gramática formal livre de contexto, apresentando regras de produção, símbolos não terminais e símbolos terminais para a produção de sentenças, com a adição de operadores de transformação ou regras de ajuste. Cabe lembrar que o poder computacional das gramáticas livre de contexto é restrito e, como o próprio nome diz, não conseguem trabalhar questões "sensíveis ao contexto". Em suma, não tem o mesmo poder computacional da Máquina de Turing.

O ambiente possui o código-fonte aberto e está disponível para Linux, Microsoft Windows XP ou superior e Apple Mac OS X. A interface do ambiente é dividida de forma que a digitação do código acontece à esquerda da saída gráfica do ambiente. Na linguagem CFDG, há diversos grupos de elementos primitivos<sup>33</sup>: variáveis, funções (matemáticas, de animação, especiais e de configurações das definições da imagem), curvas, estruturas de controle (laços, condicionais), operadores de transformação e clonagem, além dos símbolos terminais *quadrado* (*square*), *círculo* (*circle*) e da diretiva *startshape*, que define o símbolo raiz do desenho. Os meios de combinação da linguagem articulam os elementos primitivos em grupos de instruções rotuladas que definem os símbolos não terminais, permitindo a construção de ideias mais complexas pela abstração de ideias mais simples.

A prática da programação no *Context Free* começa pela criação do símbolo raiz (*startshape*), que será expandido por meio das regras de produção, até encontrar um símbolo terminal. Colocando de outra forma, as imagens são geradas por substituição: símbolos não terminais são definidos nos termos de outros símbolos e a renderização procede pela substituição em árvore dos símbolos mais abstratos em direção aos mais concretos. O processo de desenho é recursivo, podendo gerar um número infinito de terminais, de forma que o *Context Free* acaba a execução das regras de produção sempre que as primitivas tornam-se muito pequenas para serem vistas.

A linguagem não oferece suporte a outros tipos de mídia além de imagens, nem apresenta interfaces para projetos de computação robótica. O foco da ferramenta é a geração de imagens de síntese, inclusive em alta resolução. Por outro lado, é das poucas opções para se trabalhar explicitamente a questão do não determinismo.

---

<sup>32</sup> <http://tinyurl.com/livroappinventor> - acesso em 6/12/2018.

<sup>33</sup> <http://contextfree.org> - acesso em 6/12/2018.

Os conteúdos do sítio do projeto, disponíveis exclusivamente em inglês, consistem em uma introdução à linguagem, uma galeria de programas CF DG, enviados pelos usuários e um fórum de discussão. Os autores da linguagem publicaram o livro *Community of Variation* (LENTCZNER, 2008) que apresenta trabalhos desenvolvidos por artistas nos primeiros anos do projeto.

#### 5.1.4 Ambiente de Programação *NodeBox*

O ambiente *NodeBox*<sup>34</sup> foi construído a partir do *DrawBot*<sup>35</sup>, mantendo a linguagem Python como base dos *scripts*. Além das funções de desenho, o *NodeBox* possui bibliotecas organizadas nas categorias: bancos de dados, processamento e mineração de textos, grafos, funções linguísticas, busca e recuperação de conteúdos na Web e redes semânticas, *pixels* (processamento e captura de imagens, áudio e vídeo), *curvas* (manipulação e edição de vetores), *sistemas* (simulação do comportamento de organismos vivos e sistemas naturais), *design* (construção de *grids* e manipulação avançada de cores), *tipografia* (simulação de caligrafia e manipulação de glifos de fontes digitais) e *tangíveis* (uso de interfaces e protocolos de comunicação diversos). Esse *software* é ótimo para projetos da disciplina de artes. Ainda, lança mão de ferramentas da Inteligência Artificial, como algoritmos genéticos, sistemas emergentes, processamento de linguagem natural, entre outros (De Smedt; Daelemans, 2011).

O ambiente *NodeBox* está disponível apenas para Apple Mac OS X e seu código fonte é aberto. Possui suporte para projetos que envolvam computação física, utilizando a biblioteca do protocolo *Open Sound Control* – OSC<sup>36</sup>.

Quando comparado ao *DrawBot*, o *NodeBox* dispõe de uma estrutura mais completa de apoio aos aprendizes: o sítio do projeto oferece conteúdo teórico, voltado para aos fundamentos da arte generativa; tutoriais genéricos ilustrados em cinco categorias (básicos, dados, estratégias, específicos e avançados); a referência completa da linguagem, com exemplos; uma galeria com trabalhos enviados pela comunidade de usuários; e um fórum de discussão. Em outro sítio<sup>37</sup>, indicado pelos mantenedores do projeto, é possível encontrar uma série de artigos que exploram aplicações variadas do *NodeBox*.

#### 5.1.5 Ambiente de Programação *Processing*

Criado por Casey Reas e Ben Fry, sob orientação de John Maeda, no MIT-Massachusetts Institute of Technology - Media Lab, *Processing* integra uma linguagem de programação baseada na linguagem Java, um ambiente de desenvolvimento e uma metodologia de ensino. Segundo os autores, a ferramenta

---

<sup>34</sup> <http://tinyurl.com/cfdghowto> - acesso em 13/3/2018.

<sup>35</sup> <http://drawbot.com/> - acesso em 6/12/2018.

<sup>36</sup> <http://opensoundcontrol.org/> - acesso em 6/12/2018.

<sup>37</sup> <http://research.nodebox.net/index.php/Abstracts> - acesso em 6/12/2018.

foi criada para ensinar os fundamentos da programação de computadores, em um contexto visual orientado à geração e processamento de imagens, podendo ser utilizada por estudantes, artistas, *designers* e pesquisadores, tanto para esboçar e prototipar ideias, quanto para o desenvolvimento de aplicações completas (Reas; Fryr 2007). O código-fonte do ambiente é aberto e a distribuição disponível para os sistemas operacionais Linux, Microsoft Windows e Apple Mac OS X.

A linguagem propõe-se facilitar o aprendizado de técnicas de computação gráfica e de interação humano-computador por meio de elementos primitivos para desenho vetorial e matricial (2D e 3D), processamento de imagens e cores, eventos de teclado e *mouse*, comunicação em rede e programação orientada a objetos.

Há uma série de bibliotecas adicionais criadas pela comunidade de usuários<sup>38</sup> que diversificam as possibilidades de uso da linguagem em áreas como computação física, processamento e geração de áudio e vídeo, visão computacional, simulações físicas e protocolos de comunicação. Existe a possibilidade de exportar os programas escritos em *Processing* como aplicações nativas, para celulares e *tablets* Android.

Essa ferramenta considera-se uma versão simplificada da linguagem de programação Java. A escrita de código orientado a objetos é opcional, podendo ser utilizada ou não, conforme o interesse do aprendiz e demanda dos projetos. Nesse sentido, a linguagem *Processing* oferece uma situação para o desenvolvimento de conceitos do PC, conforme o aprofundamento do estudante nas práticas e perspectivas abertas pelo uso do ambiente.

O sítio do projeto é o principal ponto de partida para os interessados em aprender a linguagem, concentrando a referência oficial da linguagem, tutoriais ilustrados sobre conceitos básicos ou por tópicos, uma lista atualizada de livros específicos sobre o tema, uma galeria com projetos selecionados pelos mantenedores e um fórum de discussão para os usuários. Existe uma série de livros disponíveis, com um deles, pelo menos, editado em português<sup>39</sup>. Dentre todas as linguagens investigadas, essa apresenta a comunidade mais ativa, com o maior número de referências e a maior diversidade de usos, de modo que outras iniciativas, como *NodeBox* e *Context Free*, manifestam terem sido influenciadas pela linguagem *Processing* e buscam oferecer uma experiência semelhante, por meio de outros conceitos, práticas e perspectivas do PC.

Além das mais utilizadas também se pode citar as seguintes plataformas: AgentSheet/AgentCubes, CHERP ([ase.tufts.edu](http://ase.tufts.edu) – acesso em 4/12/2018), Alice (basawapatna.com), MATLAB e Kodu. A ferramenta Kodu ([kodugamelab.com](http://kodugamelab.com) acesso em 4/12/2018) permite que crianças criem jogos no PC e no Xbox através de uma VPL simples<sup>40</sup>. A plataforma Agent foi apresentada em 2013, é voltada para o nível de ensino K-12 e possui uma abordagem teórica. Ainda, pode-se citar como

---

<sup>38</sup> <http://processing.org/reference/libraries/> - acesso em 6/12/2018.

<sup>39</sup> <http://processing.org> - acesso em 6/12/2018.

<sup>40</sup> Lye e Koh (2014) entendem que as VPLs são mais adequadas para o contexto da educação básica, porque reduzem a carga cognitiva exigida pela sintaxe das linguagens tradicionais.

ambientes acadêmicos brasileiros: Ambiente Computacional Para Crianças das Séries Iniciais Aprenderem Programação – NEWPROG+ (Cleziel Franzoni da Costa, Carpegieri Terezani, Orivaldo de Lira Tavares) (<http://www.br-ie.org/pub/index.php/sbie/article/view/7571/5367> acesso em 4/12/2018) e Um Ambiente Online para Crianças Aprenderem Programação de Computadores - NEWPROG ([http://portais4.ufes.br/posgrad/teses/tese\\_8358\\_Disserta%E7%E3o\\_Carpegieri\\_Final.pdf](http://portais4.ufes.br/posgrad/teses/tese_8358_Disserta%E7%E3o_Carpegieri_Final.pdf) acesso em 4/12/2018).

## 5.2 Jogos *Plugged*

Foram encontrados vários jogos *plugged* para ensino das diferentes áreas do ensino fundamental e médio. No geral se encontram agrupados em portais, cuja utilização é paga. Foram selecionados três portais ativos, sendo que um deles é específico para o ensino da língua portuguesa, outro visa diferentes línguas e um terceiro é genérico. Os jogos são individuais. Outro motivo para a seleção desses portais é que as versões *plugged* apresentam características distintas de jogos similares em suas versões *unplugged*.

Jogos do tipo *forca*, que podem ser associados com autômatos finitos, são muito utilizados para o ensino de línguas. Nesses jogos, a regra é:

- Clique em uma das letras ou palavras, dependendo do objetivo do jogo. Se a letra existir em alguma palavra, a letra será exibida. Caso contrário, seu bonequinho ficará cada vez mais preso na forca. A cada erro uma parte do corpo do bonequinho é desenhada: cabeça, tronco, braço esquerdo, braço direito, perna esquerda e perna direita.
- O aluno necessita acertar as palavras e para isso pode errar até x vezes (onde o número de vezes pode ser variado).

Por exemplo, em portais educacionais, há jogos tipo *forca*, como é o caso dos portais:

- <http://www.atividadeseducativas.com.br/index.php?lista=portugu%EAs> - acesso em 4/12/2018;
- <https://www.soportugues.com.br/secoes/jogos.php#> - acesso em 4/12/2018.

Nesse tipo de jogo, os alunos precisam juntar letras para formar palavras selecionadas. Ver Figura 5.1 (acesso aos portais em 25/3/2018).

Existem inúmeros portais com atividades para o ensino médio, no entanto, poucos trazem jogos computacionais. A grande maioria apresenta exercícios do tipo escolha a resposta certa, assinale as opções erradas, enumere a primeira coluna de acordo com a segunda, etc., onde os alunos recebem pontos toda a vez que acertam os exercícios. Esses portais consideram essas atividades como jogos.





Figura 5.1: Jogo das letras

Fonte: portal AE atividades infantis

### 5.2.1 Jogos *Plugged* Não-Determinísticos

Optou-se por trazer jogos *plugged* que são programados através do uso da IA, o que lhes confere um comportamento não-determinístico e muitas vezes inteligente, ou seja, aprendem a reconhecer as estratégias do oponente e também aprendem com seus jogadores. Ou seja, na pesquisa, jogos não-determinísticos estão associados ao uso da IA no seu desenvolvimento. São jogos comerciais que utilizam, geralmente, *play station*.

### 5.2.2 Jogo *AlphaGo*

Essa versão do antigo jogo de tabuleiro chinês Go (é um jogo com mais de 2,5 mil anos de idade e originário da China) começou a ser projetada em 2014, pela equipe *Google DeepMind*<sup>41</sup>. Seus criadores dizem que durante o a partida com o campeão chinês Ke Jie (19 anos) o desfecho do jogo era tão imprevisível que a máquina teve de operar em sua máxima capacidade. Nos P primeiros 100 movimentos, foi o mais próximo que alguém conseguiu jogar contra a versão máster do *AlphaGo*. Após esses movimentos, o computador passou a desenvolver novos movimentos que desafiam milênios de sabedoria tradicional e ainda pode continuar crescendo. O *AlphaGo* que jogou contra Ke Jie é uma versão melhorada da que derrotou, em 2015, por 5 a 0, o jogador Fan Hui, campeão da Europa e que, em 2016, ganhou por 4 a 1 do sul-coreano Lee Sedol, 18 vezes campeão mundial.

Na partida, ambos os jogadores dispõem de três horas cada um para pensar as suas jogadas, e nos primeiros lances do jogo, *AlphaGo* jogou com maior rapidez, já que, após os primeiros 15 movimentos, Ke consumiu mais de 40 (quarenta) minutos do seu relógio, contra apenas 13 da IA da Google. A Google afirma que o

---

<sup>41</sup> DeepMind junta *Data Mining* (técnica de mineração de dados) com redes neurais (uma tecnologia da Inteligência Artificial) para tornar programas de computador inteligentes. Ou seja, modificam seus algoritmos de acordo com o seu aprendizado.

rápido desenvolvimento da IA no Go, o jogo mais complexo do mundo, antecipa que esse setor da tecnologia vai dar grandes passos.

O uso da IA para o desenvolvimento de jogos só é viável comercialmente, atualmente, para diversão. Ou seja, não são jogos educacionais e, também, não são encontrados, na literatura, diretamente vinculados ao PC. Espera-se que com o desenvolvimento dessa tecnologia e a sua consequente popularização, ela chegue até as aplicações educacionais. A Google é a empresa que domina o mercado dos jogos computacionais inteligentes.

No PC, esse jogo pode ser utilizado para demonstrar o funcionamento de algoritmos não determinísticos e dinâmicos (aprendem com seu funcionamento), ou seja, aportam ideias do PC diferentes das convencionais.

O Go é um jogo milenar de tabuleiro e, portanto, pode ser jogado também de forma *unplugged* (ver Figura 5.2).



Figura 5.2: Jogo Go

Fonte: Google

### 5.2.3 Jogo MS. *Pac-Man*

*Pac-Man* também é um jogo de tabuleiro tradicional, mas, o não-determinismo da versão do jogo da Google só é possível graças a sua implementação computacional, com uso de tecnologias da IA. Nessa versão, o movimento dos fantasmas é aleatório e os caminhos são modificados (6 possibilidades com 4 cores distintas) a cada etapa (níveis de complexidade do jogo). Essa implementação utiliza *play station*. Na versão *unplugged* obviamente, o tabuleiro não se modifica, é estático. Pode ser jogado de forma individual ou em duplas (ver Figura 3.32). A IA utilizada nessa implementação é menos avançada do que no caso do Go. Mesmo assim, no PC o jogo pode destacar o uso de algoritmos não-determinísticos para a seleção das posições onde os fantasmas são colocados no tabuleiro.



Figura 5.3: Ms. Pac-Man

Fonte: Google

## 5.2.4 Jogo Coding Is Good

A descrição do jogo *CodingIsGood*, na versão *plugged* (ver Figura 3.33) é a mesma já apresentada na versão *unplugged* (item 4.1.10 - Jogo *Coding Is Good*).



Figura 5.4: Jogo Coding Is Good, versão plugged

Fonte: MathAndCoding (2017)

## 5.3 Livros para o PC *Plugged*

Os livros voltados especificamente para o PC *plugged* referem-se ao ensino de linguagens de programação e há vários livros voltados para o ensino de programação para jovens e crianças. Todos seguem ideias similares, ou seja, a comunicação em uma linguagem menos técnica e a apresentação de desafios que fazem parte do universo da geração que já nasceu no mundo digital. O foco principal desses livros, normalmente, é o desenvolvimento de *APPs* para equipamentos móveis. Entretanto, como o PC *plugged* não é o objetivo principal deste trabalho, optou-se apenas por trazer alguns exemplos recentes.

### 5.3.1 Livro Guia Mangá de Bancos de Dados

O livro Guia Mangá de Banco de Dados, escrito por **Takahashi et al (2009)**, relata a história de como um grupo de jovens organiza seu negócio com o auxílio de um Banco de Dados (BD), através de uma história em quadrinhos (mangá). Além da história, o livro apresenta exercícios no final de cada capítulo. Os conceitos de PC abordados são: o porquê e como usar um BD, terminologias e modelos de BD, linguagens de BD e estrutura de sistemas de BD. Esse livro é recomendado para jovens do ensino médio (ver Figura 5.5).



Figura 5.5: Exemplo do “Guia Mangá de Banco de Dados”

Fonte: <https://novatec.com.br/livros/manga-banco-de-dados/> ( acesso em 4/12/2018)



### 5.3.2 Livro *Head First Learn to Code*

*Head First Learn to Code* trata-se de um livro para ensino de linguagem de programação. A diferença para tantos outros existentes no mercado está no fato da linguagem utilizada para introduzir o assunto para jovens. Segundo o autor, com base nas pesquisas mais recentes em ciência cognitiva e teoria da aprendizagem, o programa *Head First Learn to Code* (ver Figura 5.6) usa um formato visualmente rico para engajar a mente, em vez de uma abordagem pesada, difícil de ser entendida pelos jovens. Com o *Head First Learn to Code*, o aluno aprende a pensar computacionalmente e a escrever código para fazer com que seu computador, dispositivo móvel ou qualquer coisa com uma CPU, execute seu código. Usando a linguagem de programação Python, o livro busca ensinar passo a passo os principais conceitos de programação, bem como muitos tópicos fundamentais da Ciência da Computação, como estruturas de dados, armazenamento, abstração, recursão e modularidade [Eric Freeman \(2018\)](#).

Este livro tem previsto seu lançamento em português ainda em 2018.

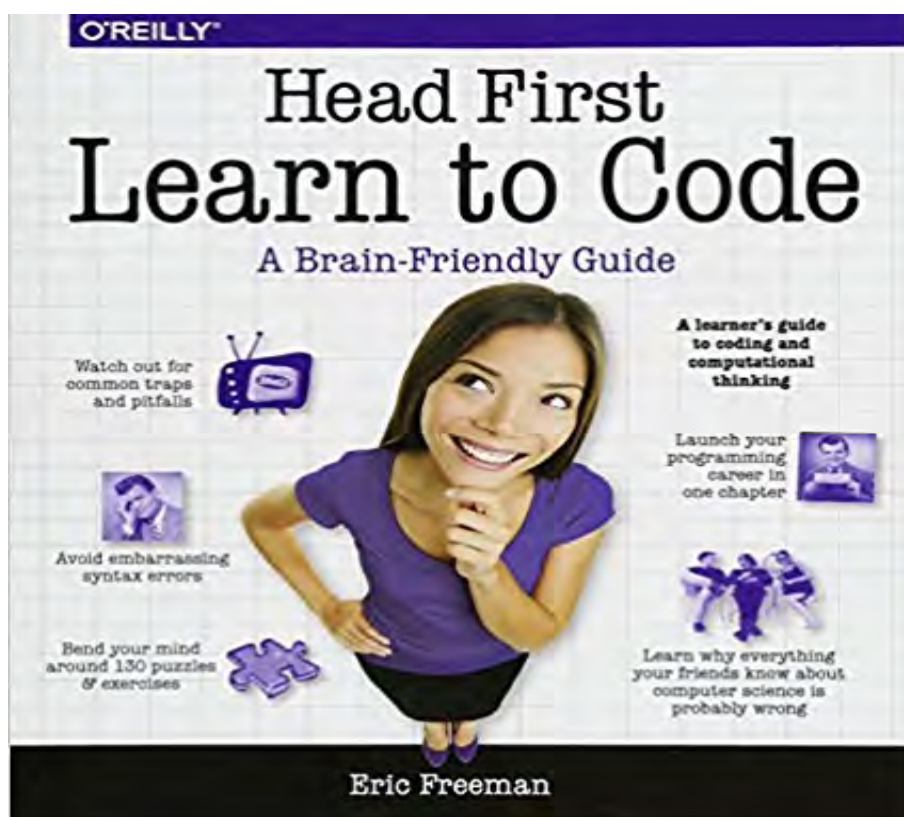


Figura 5.6: Livro “Head First Learn to Code”

Fonte: [Eric Freeman \(2018\)](#)

### 5.3.3 Livro *Coding as a Playground: Programming and Computational Thinking in the Early Childhood Classroom*

Já a associação *Association for the Advancement of Computing in Education* indica o livro *Coding as a Playground: Programming and Computational Thinking in the Early Childhood Classroom* (ver Figura 5.7), de Marina Umaschi Bers (**Bers, 2018**). Bers explica como ela passou a entender o conceito de codificação como um *playground*, “Eu sempre fui fascinada com as semelhanças entre as linguagens escritas e da programação e com o potencial de codificação para se tornar uma alfabetização que pode realmente impactar a maneira como pensamos e trazemos mudanças educacionais e sociais.” (**Bers, 2018**). Ou seja, Bers propõe a codificação como alfabetização, enfatizando o processo de aprender a codificar em uma idade jovem por meio de diversão, brincadeiras e criatividade. Existem exemplos, *links* para pesquisa e extensos recursos curriculares gratuitos. Os capítulos “Novas Línguas para Crianças Pequenas”, abordam o desenvolvimento de *Scratch Jr*, *KIBO Robots* e de uma estrutura para avaliação do uso de tecnologia nos primeiros anos de educação. Essa proposta traz uma visão interessante sobre a pesquisa e a codificação, usando blocos e robótica de chão para crianças.

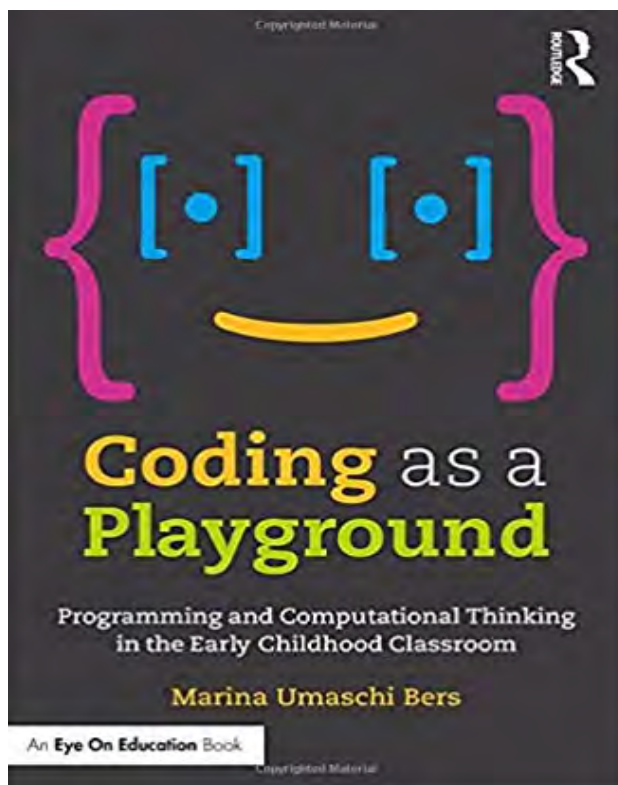


Figura 5.7: Capa do livro *Coding as a Playground*

Fonte: Marina Umaschi Bers

### 5.3.4 Livro *Hello Ruby Adventures in Coding*

O livro *Hello Ruby Adventures in Coding* (ver Figura 3.37), escrito por **Liukas (2015)**, que também escreve livros para o PC *unplugged*, relata a história de uma menininha (a heroína da história) criativa que deve encontrar soluções para diversos problemas, utilizando o PC. A história é contada até a metade do livro e, na seção seguinte, são propostas atividades, utilizando os quatro pilares do PC, para serem exercitados pela criança com a ajuda de adultos, com base em elementos contidos no livro (*e.g.* personagens e exemplos). Não há, até o momento, uma tradução do livro para o português.

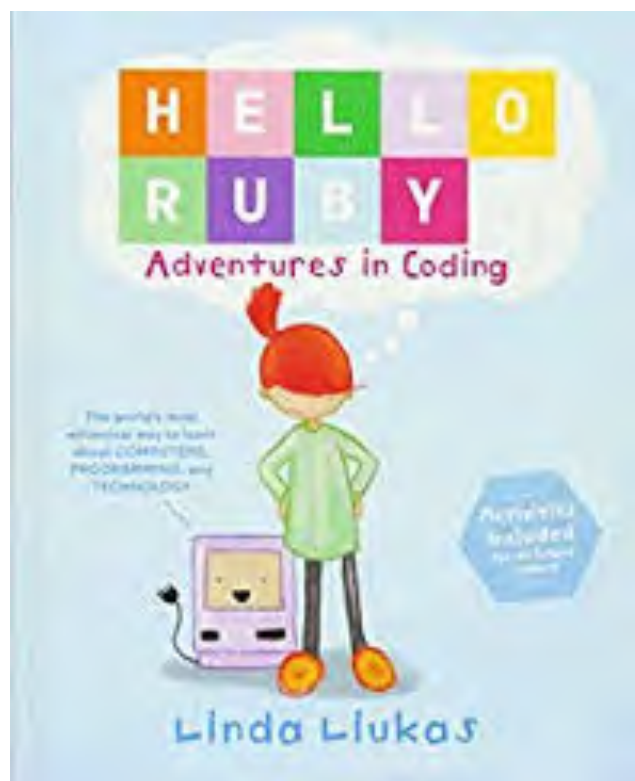


Figura 5.8: Capa do livro Hello Ruby

Fonte: [sítio onde o livro foi encontrado á venda](#) (2018)



## 6. Avaliação de Materiais Educacionais Voltados ao Desenvolvimento do Pensamento Computacional nas Escolas Brasileiras

A seguir será apresentado um resumo dos principais resultados da Avaliação de Materiais Educacionais Voltados ao Desenvolvimento do Pensamento Computacional nas Escolas Brasileiras. O relatório com detalhes da avaliação encontra-se no item 15 - Anexos.

A partir da aplicação dos métodos de busca descritos em 2.6 - Avaliação de Materiais foram avaliados 7 (sete) materiais desplugados e 10 (dez) materiais plugados, totalizando 17 (dezesete) avaliações. A Tabela 6.1 apresenta uma visão geral dos materiais avaliados, sua categorização (plugado ou desplugado), língua na qual estão disponíveis, custo e o percentual de critérios atendidos com relação ao desenvolvimento do Pensamento Computacional, de acordo com cada etapa de ensino.

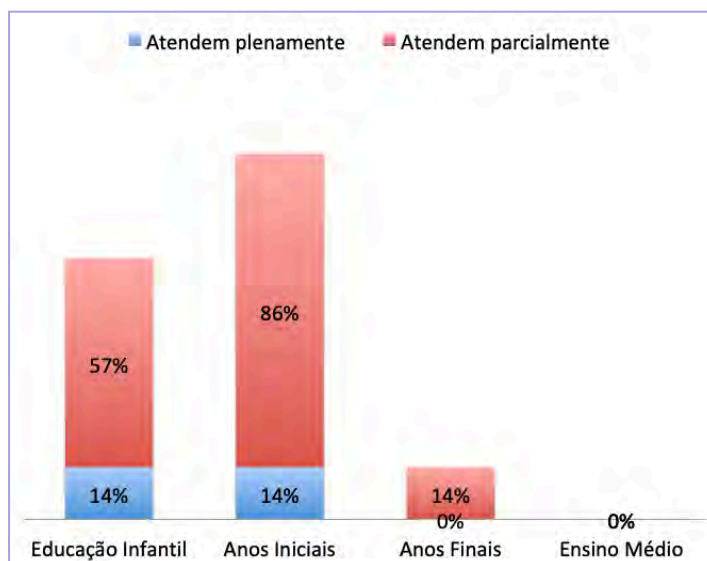
*Tabela 6.1: Visão geral dos materiais avaliados*

		Língua	Custo	Educação Infantil	Anos Iniciais	Anos Finais	Ensino Médio
Desplugadas	Computer Science Unplugged	Português	Livre	17%	100%	80%	25%
	Turma da Mônica	Português	Livre	83%	67%	20%	17%
	Estacionamento Algorítmico	Português	Livre	83%	67%	20%	17%
	Cody e Ruby	Inglês	Livre	67%	50%	30%	17%
	Tetris	Português	Livre	0%	67%	30%	0%
	Flexicards	Português	Livre	100%	67%	20%	25%
	Moon	Espanhol/Inglês	Livre	67%	83%	10%	42%
Plugadas	Alice	Português	Livre	100%	100%	100%	50%
	Scratch	Português	Livre	100%	100%	100%	50%
	Portugol	Português	Livre	50%	67%	60%	25%
	AgentCubes	Inglês	R\$200	100%	67%	60%	42%
	Kodu	Inglês	Livre	0%	33%	0%	17%
	AppInventor	Português	Livre	0%	50%	40%	42%
	CodeAcademy	Inglês	R\$80/mês	0%	100%	70%	25%
	Processing	Inglês	Livre	67%	100%	100%	100%
	LightBot	Inglês	R\$8	50%	33%	10%	8%
	Tinkercad	Inglês	Livre	100%	100%	100%	100%

A análise dos dados do quadro permite observar algumas informações relevantes:

- Há um número razoável de materiais disponíveis em língua portuguesa (5 materiais desplugados, 4 materiais plugados);
- Nenhum dos materiais em língua portuguesa possui custo de utilização;
- Globalmente, os materiais plugados alcançam maiores percentuais de critérios atendidos, para desenvolvimento do pensamento computacional nas diferentes etapas de ensino.

A Figura 6.1 mostra o percentual de critérios atendidos com relação ao desenvolvimento do Pensamento Computacional somente para *os materiais desplugados avaliados*, de acordo com as diferentes etapas de ensino. Os resultados mostram que os referidos materiais atendem mais adequadamente as etapas da Educação Infantil e Anos Iniciais do Ensino Fundamental.



*Figura 6.1: Percentual de critérios atendido pelo conjunto total de materiais desplugados avaliados de acordo com etapa de ensino*

Já a Figura 6.2 mostra o percentual de critérios atendidos com relação ao desenvolvimento do Pensamento Computacional somente para os *materiais desplugados avaliados disponíveis em português*, de acordo com as diferentes etapas de ensino. Os resultados dessa análise permitem concluir que o conjunto disponível em língua portuguesa é ainda limitado, com número pequeno de materiais capaz de atender plenamente os critérios para desenvolvimento do Pensamento Computacional.

A Figura 6.3 mostra o percentual de critérios atendidos com relação ao desenvolvimento do Pensamento Computacional para *os materiais plugados avaliados*, de acordo com as diferentes etapas de ensino. Os resultados mostram que há um conjunto de materiais que atende satisfatoriamente todas as etapas da Educação Básica, sendo o potencial desses materiais um pouco mais restritivo para o Ensino Médio.

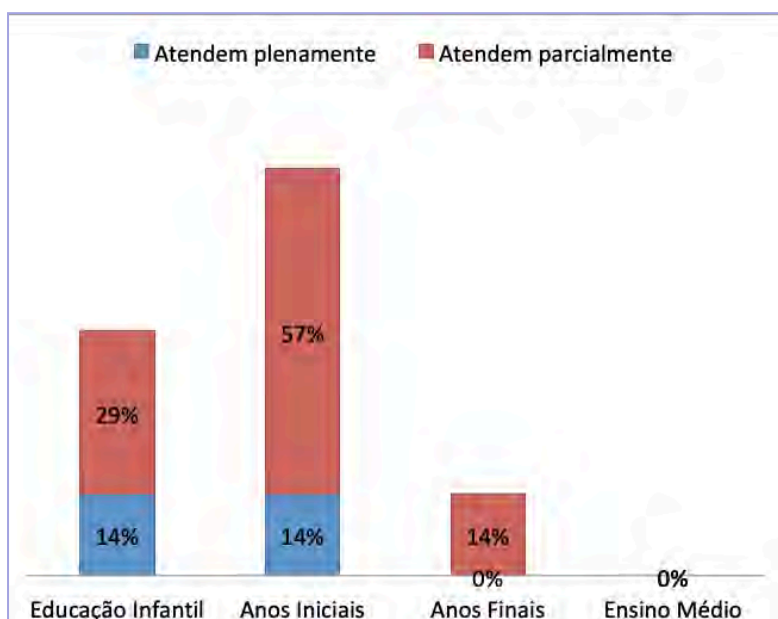


Figura 6.2: Percentual de critérios atendido pelos materiais desplugados avaliados disponíveis em português, de acordo com etapa de ensino

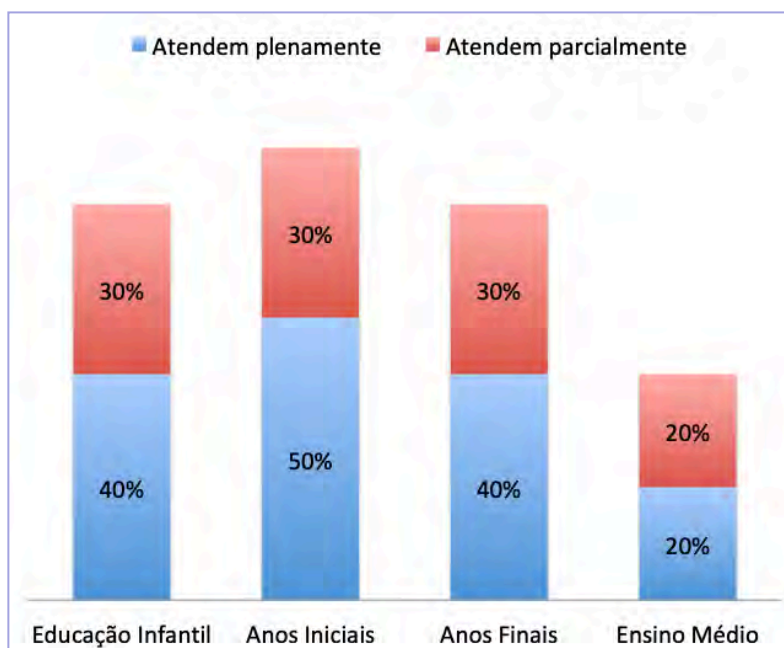


Figura 6.3: Percentual de critérios do Pensamento Computacional atendido pelos materiais plugados avaliados de acordo com etapa de ensino

Já Figura 6.4 mostra o percentual de critérios atendidos com relação ao desenvolvimento do Pensamento Computacional somente para os *materiais*

*plugados avaliados disponíveis em português*, de acordo com as diferentes etapas de ensino. Esse conjunto, apesar de pequeno, atende de maneira mais satisfatória os critérios para desenvolvimento do Pensamento Computacional nas diferentes etapas da Educação Básica.

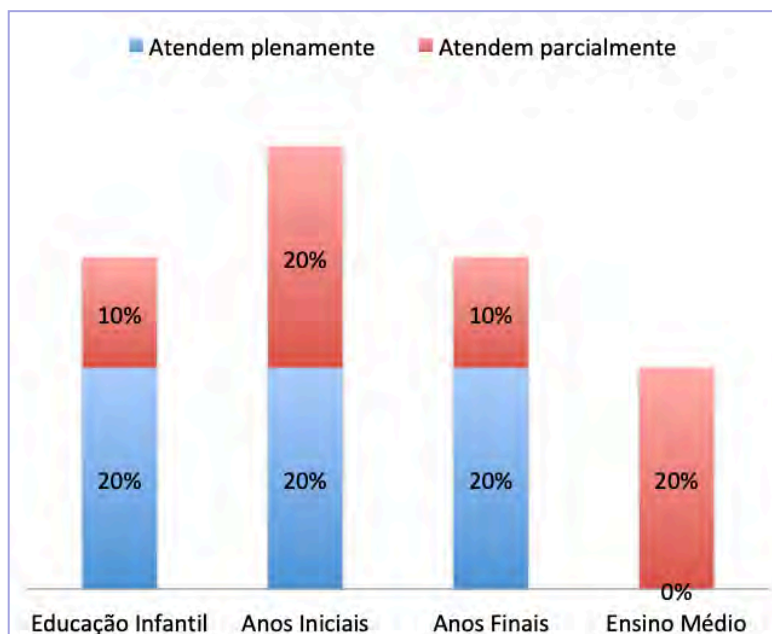


Figura 6.4: Percentual de critérios do Pensamento Computacional atendido pelos materiais plugados avaliados de acordo com etapa de ensino

Ao considerar-se o conjunto de materiais plugados e desplugados, tem-se então um conjunto mais significativo de materiais com características que atendem tanto requisitos para desenvolvimento do pensamento computacional, quanto critérios de qualidade, necessários para sua utilização nas diferentes etapas da Educação Básica. Apesar de não ser muito abrangente, entende-se que esse conjunto de materiais tenha bom potencial para sustentar políticas mais amplas de uso de tecnologia para desenvolvimento do Pensamento Computacional nas diferentes etapas da Educação Básica.

Por fim, cabe ressaltar que nenhum dos materiais avaliados tem características ou conteúdo que destoem da legislação e normas oficiais que regulamentam a Educação Básica, com exceção do critério relacionado à acessibilidade dos materiais por pessoas com deficiência ou com mobilidade reduzida. Mas sabe-se que muitos materiais educacionais hoje produzidos no País e em circulação nas escolas não são capazes de atender na íntegra esse critério. Nessas situações, é importante que o professor seja capaz de propor adaptações e/ou alternativas, para poder integrar esses materiais em suas práticas e assim contemplar todos os estudantes, considerando-se as particularidades de cada caso.

## 7. Crítica às Abordagens e à Pesquisa do Pensamento Computacional

Existe uma forte tendência a definir o PC usando uma forma puramente sequencial, sendo pouco exploradas as facilidades paralelas/concorrentes ou não-deterministas e muito menos as afetivas, o que induz a possivelmente perigosa e restritiva mudança para um raciocínio não natural da criança e do jovem. Como já lembrado, o pensamento humano costuma ser concorrente (como caminhar e assistir a um vídeo), não determinista (como optar por um entre os diversos caminhos em um deslocamento) e afetivo (como a compra de um bem não baseada em "argumentos racionais"). Ou seja, existe uma forte tendência simplista das atuais implementações do PC de seguir modelos e padrões que não refletem de forma adequada o contexto real.

Nenhum dos materiais encontrados nesta pesquisa segue, por exemplo, uma ideia da programação declarativa e/ou paralela. Utilizá-los poderá levar a um direcionamento “bitolante” da forma de raciocinar dos alunos. Veja-se os pilares que suportam a PC. Sendo assim, é importante que a comunidade de usuários do PC busque o planejamento e a produção de artefatos para os mais diferentes produtos e não apenas no contexto da programação tradicional.

No que se refere à pesquisa no Brasil e no mundo, se forem considerados os números gerais em artigos que abordam metodologias educacionais e tecnologia, não apenas do PC, como o desenvolvimento de métodos educacionais, tem-se a seguinte distribuição:

em 1º os EUA, com Scopus (28%) e WOS (17%);  
em 2º a China, com Scopus (10%) e WOS (8%);  
em 3º a Espanha, com Scopus (7%) e WOS (8%);  
em 4º o Reino Unido, com Scopus (6%) e WOS (7%);  
em 5º o Canadá e o Japão, com 5%, em ambas as bases;  
em 6º o Brasil, com 3% da pesquisa, em ambas as bases.

Esses dados deixam claro que as principais potências mundiais têm investido na área (EUA e China) e a União Europeia, se considerada a soma de Espanha e Reino Unido. O Brasil, em particular, investe pouco se forem consideradas as necessidades educacionais do País.

## 8. Avaliação do uso do Pensamento Computacional

Avaliar uma metodologia de ensino como o PC é um tema polêmico e sem consenso acadêmico, tanto no como introduzi-la, quanto no que toca à aprendizagem dos alunos (talvez esse último aspecto seja o mais difícil de ser avaliado). Este item busca apresentar experiências no processo de avaliação do PC, tanto na sua forma *plugged*, como na forma *unplugged*.

No que se refere aos alunos, o *National Research Council* (2011) destaca a falta de metodologia adequada para avaliação do PC em escolas do ensino fundamental, séries iniciais, onde é majoritariamente utilizado, assim como pesquisas formais com relação à compreensão de alunos sobre conceitos envolvidos no PC. O PC em sua implementação *unplugged* é mais difícil de ser avaliado, exatamente pela falta de parâmetros concretos. Na sua forma *plugged*, existem indicadores concretos para avaliar, pelo menos na sua forma mais utilizada, que é o ensino de programação. A engenharia de *software* fornece várias métricas para avaliar a qualidade do código.

Por exemplo, *Taub et al* (2012) levaram a abordagem *unplugged* para a sala de aula, mais especificamente as atividades propostas no livro de *Bell et al* (1997) e, após a aplicação, através do uso de questionários e entrevistas, investigou-se os efeitos dessas atividades na visão de alunos do ensino fundamental do país. Os resultados mostram que:

*“embora a maior parte dos estudantes entenderam o que é a Computação, eles também entendiam que o computador era a essência da Computação e não apenas uma ferramenta”.*

Com objetivos e resultados semelhantes, outro projeto de Computação *unplugged* foi implementado com duração de um ano, para estudantes do ensino médio, com o objetivo de "estimular a próxima geração de alunos de graduação em pesquisa na área da Computação" (*Feaster et al*, 2011). Os resultados mostram que o programa não teve impacto percebido sobre os alunos, nem propiciou uma melhor compreensão do conteúdo envolvido, ou sobre as suas atitudes em relação à Computação.

Já em outro estudo, envolvendo um grupo de pesquisadores (*Lambert e Guiffre*, 2009) que trabalhou com turmas de quarto ano, com o objetivo de aumentar o interesse dos estudantes pela área da Computação, com uso de atividades *unplugged*, os resultados, baseados em pré e pós-teste, mostram uma melhoria na confiança e interesse, tanto na área da Computação, como na de Matemática. Mas, como se percebe as atividades de avaliação são realizadas, na maioria dos casos, de forma empírica.

Para Grover (*Grover*, 2013), sem a devida avaliação, o PC, na sala de aula, não terá como seguir o caminho de sucesso em nenhum currículo, ou seja, além da necessidade de se avaliar a eficácia de qualquer abordagem curricular ao integrar o



PC, necessita-se definir métricas que permitam avaliar a aprendizagem dos alunos a respeito dos conceitos envolvidos no ensino do PC. Quando encontrados relatos da avaliação, esses estão voltados para a habilidade de programação, por exemplo, das linguagens mais utilizadas como Python e, de programação visual, o Scratch. Como já foi dito neste texto, avaliar a qualidade de código ou as habilidades necessárias para programar é um assunto já com bastante estudo na Computação. No entanto, a avaliação do PC *unplugged* é bastante difícil, talvez até pelo seu curto tempo de existência.

De forma geral, a avaliação de conceitos da Computação em escolas de ensino fundamental e médio tem sido um desafio para os educadores, conforme o relatório do *National Research Council* (2011) e também segundo Boucinha (Boucinha et al, 2017).

Ou seja, se faz necessária a criação de *frameworks*, métodos ou métricas para avaliar, de forma igualitária, o desempenho dos estudantes quando envolvidos em atividades relacionadas ao PC, porém não existe ainda um consenso a respeito de quais parâmetros e de como avaliar a aquisição dos conceitos do PC (Barcelos; Silveira, 2012), (Seiter; Foreman, 2013) e (Brennan, 2011). Ou seja, a maioria dos artigos aponta para a necessidade de se realizar algo, mas não apresentam casos concretos de avaliação, mesmo que de forma empírica.

Nesse contexto, Barcelos e Silveira (2012) avançam o estado da arte, ao propor a avaliação dos estudantes baseada em três dimensões e quinze categorias, para análise dos indicadores, para ser utilizada em conjunto com atividades em programação visual Scratch. A proposta de Barcelos e Silveira envolve dimensões, categorias e indicadores. Mas, mesmo assim, o foco está na habilidade de programação, em particular, e não na aquisição de conceitos de Computação. Trata-se apenas de um exemplo.

Os itens são dois: conceitos Computacionais e práticas Computacionais. As categorias de análise se dividem entre os conceitos e as práticas. As categorias de análise estão divididas entre categorias de comando, que envolvem movimento, aparência, som, caneta, controle, sensores, operações, variáveis. Para verificar essas categorias tem-se indicadores como, por exemplo, blocos de programação de cada categoria de comando presentes nos projetos Scratch.

Já para o item Práticas computacionais, tem-se como exemplos de categorias de análise incremental e interativa: teste e depuração; reutilização; abstração e modularização. Os indicadores para essas categorias são: testemunhos (discurso escrito e falado) sobre, por exemplo, a realização de experiências, tentativa e erro, e descobertas, sem seguir um método científico.

Na mesma linha, Seiter e Foreman (2013) apresentam uma proposta simplificada, através de uma generalização de 13 (treze) indicadores, sendo que cada indicador possui até 3 (três) níveis de desempenho, voltados para o uso na programação visual Scratch.

O uso de três níveis de desempenho também são encontrados na proposta de Rodriguez (2015). São eles: “insatisfatório”, “parcialmente proficiente” e “proficiente”. Nessa proposta existe apenas um tema: perspectivas Computacionais

e 3 (três) categorias de análise: expressar, conectar e questionar. Os indicadores continuam sendo discurso escrito e falado, através de ações como criar diálogo, criar personagens, recriar eventos, criar um jogo, movimentar, escrever, comunicar, atuar em conjunto, colocar questões, explicar, ensinar, responder, fazer, novas cores, demonstrar, interagir com colegas. Através do uso desses indicadores, o aluno é classificado como tendo um desempenho insatisfatório, parcialmente proficiente ou proficiente. As ações avaliadas referem-se sempre a instruções de linguagens de programação como *Se Então*, situações de repetição (*loops*) e procedimentos. No entanto, se for olhado de forma mais ampla, essas situações podem estar presentes em jogos, atividades descritivas, ou mesmo em códigos de programas.

Nesse sentido, é possível citar também os estudos que investigam a eficiência de linguagens de programação (visual e código) com crianças, tais como, Román-González *et al* (2015), Román-González *et al* (2016), Grover e Basu (2017) e Franklin *et al* (2017), porém carecem de abordagens *unplugged*.

Já Wing (2011) defende a avaliação utilizando conceitos mais amplos da Computação, ao verificar o desempenho dos estudantes em atividades relativas ao PC. Os indicadores são divididos em três grupos classificados em Indicador e Questão e/ou Definição:

1. Indicador: eficiência que tem como definição rapidez, espaço de armazenamento e consumo de energia;
2. Indicador: exatidão que tem como definição resposta correta e processo correto;
3. Indicador: resultado que tem como definição simplicidade e elegância, usabilidade, facilidade de manutenção, adaptabilidade e custo.

Como se pode perceber, muitos desses critérios de avaliação são subjetivos, como, por exemplo, simplicidade e elegância.

Berry (2013) apresenta sugestões aos professores de planejamento de aulas (esse ponto talvez seja sua contribuição mais significativa), recursos que podem ser utilizados, metodologias de ensino, e indica sugestões de como os professores podem avaliar o progresso dos estudantes ao realizar atividades com o PC. O relatório apresenta os seguintes pontos:

1. Não é trivial para os professores avaliar o conhecimento e a compreensão dos alunos com base apenas nos resultados das tarefas práticas. O autor lembra que, se os alunos trabalham de forma colaborativa, pode ser difícil identificar a contribuição de cada estudante. Além disso, se o ensino da Computação estiver incorporado (de forma transdisciplinar) em outras disciplinas ou assuntos abordados em sala de aula, muitas vezes é complicado separar o aprendizado de conceitos computacionais de outros assuntos. Esse último caso é o mais utilizado nas implementações *unplugged* conhecidas.
2. O autor também sugere 6 (seis) técnicas de avaliação que podem ser usadas, em conjunto com critérios previamente definidos, pelos professores

(técnicas que nem sempre são viáveis, pois envolvem algum domínio da tecnologia que os alunos não têm acesso, em geral, no uso dessa metodologia *unplugged*, mas que pode ser útil para o caso *plugged*), a saber:

- Auto Avaliação: os alunos podem manter um *blog* ou registro de vídeo de seu trabalho, contendo comentários reflexivos, ao lado de exemplos do que eles fizeram.
- Avaliação por pares: os estudantes, trabalhando com um parceiro, precisam revisar e ajudar a corrigir, algoritmos/programas, ou fornecer *feedback* crítico e construtivo ao seu parceiro. Esse método é o mesmo utilizado pelos desenvolvedores de *software* profissional.
- Questionamentos abertos: uma forma de avaliar aqueles conhecimentos dos estudantes sobre os conceitos abrangidos pelo programa de estudo e pode não ser de fácil identificação no trabalho realizado. Alguns exemplos de perguntas são: Por que o *Youtube* indica esses vídeos? Como o seu programa funciona? Por que o Google colocou esse resultado no topo? Por que esse site pode não ser seguro?, etc. Para se utilizar esse tipo de perguntas, é necessário o uso de computador.
- Discussão/Argumentação com pares: envolve o uso pelos estudantes de perguntas abertas entre alunos, mantendo o foco no que aprenderam e não apenas no que realizaram.
- Definição de metas: as habilidades de gerenciamento de projetos, como planejamento, organização, motivação de colegas e alocação de recursos, são importantes em projetos do mundo real e podem ser aplicadas em situações de aprendizagem. A decomposição de problemas complexos em subproblemas é usada nessa técnica.
- KWL (*Know, Want and Learned* – Sabe, Quer e Aprendido): tentar identificar o que os estudantes já sabem, o que eles querem e o que eles efetivamente aprenderam, é uma técnica que possibilita identificar a evolução dos alunos, porém difícil de ser implementada, para o caso de identificar o que os alunos efetivamente aprenderam.
- *Feedback* automático: essa técnica utiliza páginas e tutorias da Internet que são interativas, para que os alunos possam aprender linguagens de programação. A interatividade é fundamental para o *feedback* imediato de sucesso ou falha, na resolução de um desafio.

Lambert e Guiffre (2009) também realizaram uma tentativa de medir se houve um incremento no interesse pela área da Computação ou Matemática, através de 3 (três) sessões de 30 (trinta) minutos. Mas os autores não utilizaram esse experimento para verificar se houve mudanças nas habilidades relacionadas ao PC. Na mesma linha, Faber et al (2017) desenvolveram um conjunto de seis aulas para serem aplicadas em 26 (vinte e seis) escolas, da Educação Básica, na Holanda. O *feedback* de professores e alunos foi positivo a respeito das aulas pré planejadas, mas o autor não utilizou uma métrica para aferir a satisfação dos mesmos ou dados

estatísticos sobre o desempenho desses alunos, no que se refere ao PC desses alunos das 26 escolas que participaram da pesquisa.

Como se pode ver nas propostas apresentadas, as avaliações de alunos ou professores são realizadas de maneira qualitativa ou com escalas de baixa precisão e de forma empírica, omitindo uma avaliação quantitativa e cientificamente forte.

Por outro lado, **Campos et al (2014)** realizaram uma das primeiras tentativas de medir, através de pré e pós-testes, o uso do PC com crianças. Os testes foram aplicados antes e após apenas 3 (três) aulas com atividades desenvolvidas por **Bell et al (1997)**, em seu livro. O teste é composto por quatro perguntas, do tipo:

*Preencher o pontilhado com o elemento que complete cada sequência: 1 2 4 8*

.....

*Também são utilizados símbolos para substituir letras como, por exemplo, substituir as consoantes da palavra CASA por retângulos e as vogais por corações.*

Os resultados das aplicações desses testes sugerem que não houve uma melhoria estatisticamente significativa e/ou válida da aprendizagem dos alunos com a adoção do PC, nem no entendimento dos conceitos de Computação, nem nas disciplinas envolvidas: Matemática e Ciências. Porém, segundo o autor, esse experimento permitiu uma percepção da necessidade de melhorar o instrumento de avaliação.

Outra pesquisa que trata da avaliação do PC no ensino fundamental, na Itália, foi relatada por **Scaico et al (2012)**, a qual descreve três sessões de PC *unplugged* para estudantes de quinto e nono anos e logo após a realização de um teste relativo ao tema abordado. Mas, mesmo nesse caso, não foi feito um pré-teste para verificar alterações no desempenho dos alunos e compará-lo com o pós-teste. Esse exemplo da avaliação reflete a maioria dos casos encontrados na literatura. A avaliação é realizada com um número muito reduzido de estudantes e, muitas vezes, após apenas um conjunto de atividades que envolvem o uso do PC. Não são sistemáticas e nem contínuas.

No que se refere à avaliação do PC *unplugged*, merece destaque a proposta de **Rodriguez (2015)**, **Rodriguez et al (2016)** e **Rodriguez et al (2017)**, já mencionada neste item sobre avaliação, que procura avaliar estudantes através de 5 (cinco) atividades *unplugged*, utilizando uma adaptação dos níveis de pensamento de Bloom (Taxonomia de Bloom), ou seja, em três níveis (*proficiente, parcialmente proficiente e insatisfatório*). Rodriguez utilizou um teste composto por perguntas abertas que não passaram por um processo de validação estatística (certificar-se de que as perguntas não são tendenciosas) e o aplicou em turmas de sétimo ano do ensino fundamental, onde foi possível verificar que os alunos aprenderam habilidades vinculadas ao PC com atividades *unplugged*.

Os trabalhos aqui citados são parte de um conjunto de pesquisas que buscam apresentar a utilização atual do PC, porém não utilizam uma solução direta, de fácil aplicação e com um processo de validação formal, para que se possa atingir um resultado mais preciso e confiável. A revisão de estudos que fornecem evidências

sobre a efetividade da Computação *unplugged*, para desenvolver habilidades vinculadas ao PC, mostra que esse tema encontra-se ainda em aberto.

A forma como as gerações atuais e futuras irão interagir com a tecnologia fará toda a diferença. A literatura, através de pesquisas em educação computacional, sugere que a programação pode constituir um método para ensino de Matemática e Ciências (Guzdial, 2016). A pesquisa realizada por Tew *et al* (2008) afirma que, após o aprofundamento de conhecimento de seus estudantes em conceitos da Computação, alunos que tinham dificuldades em outras disciplinas começaram a correlacionar esses conceitos com os temas abordados nas outras disciplinas e tiveram um rendimento superior, inclusive comparável aos dos melhores alunos. E, de acordo com Guzdial, uma forma de incentivar estudantes a entrar no mercado de trabalho ocorre através do domínio da Computação, o que gera um impacto significativo em suas carreiras. Se os estudantes têm acesso à Computação desde a Educação Básica, essa aproximação é facilitada.

## 9. Integração na Educação Básica

A *Base Nacional Comum Curricular (BNCC)* é um instrumento de gestão pedagógica e tem como objetivo definir os conhecimentos essenciais aos quais todos os estudantes brasileiros têm o direito de ter acesso e se apropriar durante sua trajetória na educação Infantil, fundamental e média (pública ou privada). A Base é parte de uma nova proposta de Currículo e orienta a formulação do projeto Político-Pedagógico das escolas<sup>42</sup>.

A BNCC (disponível em: [http://basenacionalcomum.mec.gov.br/wp-content/uploads/2018/12/BNCC\\_14dez2018\\_site.pdf](http://basenacionalcomum.mec.gov.br/wp-content/uploads/2018/12/BNCC_14dez2018_site.pdf), acesso em 14/12/2018) faz referências explícitas ao *Pensamento Computacional* e a diversos aspectos correlatos como *Cultura Digital*, *TDIC (tecnologias digitais da informação e comunicação)*, *Redes Sociais*, *algoritmo*, etc., os quais são destacados ao longo capítulo, em geral na forma de citações.

### 9.1 Introdução da BNCC

Entre diversas propostas, a Introdução da BNCC determina que, ao longo da Educação Básica (Educação Infantil, Ensino Fundamental e Ensino Médio), os estudantes irão desenvolver “*Dez Competências Gerais*”, tanto cognitivas quanto sócio emocionais, que incluem o exercício da curiosidade intelectual, o uso das tecnologias digitais de comunicação e a valorização da diversidade dos indivíduos. Pode-se destacar as seguintes competências que seguem a linha do Pensamento Computacional, sendo:

*Valorizar e utilizar os conhecimentos historicamente construídos sobre o **mundo físico, social, cultural e digital** para entender e explicar a realidade, continuar aprendendo e colaborar para a construção de uma sociedade justa, democrática e inclusiva;*

*Utilizar diferentes linguagens – verbal, corporal, visual, sonora e **digital** –, bem como conhecimentos das linguagens artística, matemática e científica, para se expressar e partilhar informações, experiências, ideias e sentimentos em diferentes contextos e produzir sentidos que levem ao entendimento mútuo;*

---

<sup>42</sup><http://www.buddys.com.br/> - acesso em 4/12/2018

<http://www.ctrlplay.com.br/> - acesso em 4/12/2018

<http://www.dragonbyte.com.br/> - acesso em 4/12/2018

<http://escolapixel.com.br/> - acesso em 4/12/2018

<http://www.happycode.com.br/> - acesso em 4/12/2018

<http://konfidegeeks.com.br/>

<http://www.madcode.com.br/> - acesso em 4/12/2018



*Compreender, utilizar e criar **tecnologias digitais de informação e comunicação (TDIC)** de forma crítica, significativa, reflexiva e ética nas diversas práticas sociais (incluindo as escolares) para se comunicar, acessar e disseminar informações, produzir conhecimentos, resolver problemas e exercer protagonismo e autoria na vida pessoal e coletiva.*

Ao tratar o compromisso com a Educação Integral, destaca que, “a sociedade contemporânea impõe um olhar inovador e inclusivo a questões centrais do processo educativo: o que aprender, para que aprender, como ensinar, como promover redes de aprendizagem colaborativa e como avaliar o aprendizado. Neste sentido, no novo cenário mundial:

*Requer o desenvolvimento de competências para aprender a aprender, saber lidar com a informação cada vez mais disponível, atuar com discernimento e responsabilidade nos contextos das **culturas digitais**, aplicar conhecimentos para resolver problemas, ter autonomia para tomar decisões, ser proativo para identificar os dados de uma situação e buscar soluções, conviver e aprender com as diferenças e as diversidades.*

Na progressão das aprendizagens essenciais do Ensino Fundamental para o Ensino Médio, o conjunto das competências específicas e habilidades definidas para o Ensino Médio concorre para o desenvolvimento das competências gerais da Educação Básica e está articulado às aprendizagens essenciais estabelecidas para o Ensino Fundamental.

## 9.2 Etapa do Ensino Fundamental

Na etapa do Ensino Fundamental, no contexto da Educação Básica, há crianças e adolescentes que, ao longo desse período, passam por uma série de mudanças relacionadas a aspectos físicos, cognitivos, afetivos, sociais, emocionais, entre outros. Nesse contexto:

*...a **cultura digital** tem promovido mudanças sociais significativas nas sociedades contemporâneas. Em decorrência do avanço e da multiplicação das tecnologias de informação e comunicação e do crescente acesso a elas pela maior disponibilidade de computadores, telefones celulares, tablets e afins, os estudantes estão dinamicamente inseridos nessa cultura, não somente como consumidores. Os jovens têm se engajado cada vez mais como protagonistas da **cultura digital**, envolvendo-se diretamente em novas formas de interação multimidiática e multimodal e de atuação social em rede, que se realizam de modo cada vez mais ágil. Por sua vez, essa cultura também apresenta forte apelo emocional e induz ao imediatismo de respostas e à efemeridade das informações, privilegiando análises superficiais e o uso de imagens e formas de expressão mais sintéticas, diferentes dos modos de dizer e argumentar característicos da vida escolar.*

*É importante que a instituição escolar preserve seu compromisso de estimular a reflexão e a análise aprofundada e contribua para o desenvolvimento, no estudante, de uma atitude crítica em relação ao conteúdo e à multiplicidade de ofertas midiáticas e digitais. Contudo, também é imprescindível que a escola compreenda e incorpore mais as novas linguagens e seus modos de funcionamento, desvendando possibilidades de comunicação (e também de manipulação), e que eduque para usos mais democráticos das tecnologias e para uma participação mais consciente na **cultura digital**. Ao aproveitar o potencial de comunicação do **universo digital**, a escola pode instituir novos modos de promover a aprendizagem, a interação e o compartilhamento de significados entre professores e estudantes*

Ainda:

*A compreensão dos estudantes como sujeitos com histórias e saberes construídos nas interações com outras pessoas, tanto do entorno social mais próximo quanto do universo da **cultura midiática e digital**, fortalece o potencial da escola como espaço formador e orientador para a cidadania consciente, crítica e participativa.*

## 9.2.1 Linguagens

Na Área de Linguagens, explicita a questão da linguagem digital nas *Competências Específicas*:

*Utilizar diferentes linguagens – verbal (oral ou visual-motora, como Libras, e escrita), corporal, visual, sonora e **digital** –, para se expressar e partilhar informações, experiências, ideias e sentimentos em diferentes contextos e produzir sentidos que levem ao diálogo, à resolução de conflitos e à cooperação*

Na Componente Língua Portuguesa, a BNCC justifica o fato de “dedicar espaço maior no trecho introdutório destinado aos *novos letramentos e à cultura digital* é devido tão somente ao fato de que sua articulação ao currículo é mais recente e ainda pouco usual, ao contrário da consideração dos letramentos da letra já consolidados”:

*Não se trata de deixar de privilegiar o escrito/impresso nem de deixar de considerar gêneros e práticas consagrados pela escola<sup>30</sup>, tais como notícia, reportagem, entrevista, artigo de opinião, charge, tirinha, crônica, conto, verbete de enciclopédia, artigo de divulgação científica etc., próprios do letramento da letra e do impresso, mas de contemplar também os **novos letramentos, essencialmente digitais**.*

O componente Língua Portuguesa da BNCC:

*dialoga com documentos e orientações curriculares produzidos nas últimas décadas, buscando atualizá-los em relação às pesquisas recentes da área e às transformações das práticas de linguagem ocorridas neste século, devidas em*

*grande parte ao desenvolvimento das **tecnologias digitais da informação e comunicação (TDIC)**.*

Sobre as práticas de linguagem contemporâneas com textos cada vez mais multissemióticos e multimidiáticos e com novas formas de produzir, de configurar, de disponibilizar, de replicar e de interagir:

*As novas ferramentas de edição de textos, áudios, fotos, vídeos tornam acessíveis a qualquer um a produção e disponibilização de textos multissemióticos nas redes sociais e outros ambientes da Web. Não só é possível acessar conteúdos variados em diferentes mídias, como também produzir e publicar fotos, vídeos diversos, podcasts, infográficos, enciclopédias colaborativas, revistas e livros **digitais** etc. Depois de ler um livro de literatura ou assistir a um filme, pode-se postar comentários em **redes sociais** específicas, seguir diretores, autores, escritores, acompanhar de perto seu trabalho; podemos produzir playlists, vlogs, vídeos-minuto, escrever fanfics, produzir e-zines, nos tornar um booktuber, dentre outras muitas possibilidades. Em tese, a **Web** é democrática: todos podem acessá-la e alimentá-la continuamente.*

*Essa consideração dos novos e multiletramentos; e das práticas da **cultura digital** no currículo não contribui somente para que uma participação mais efetiva e crítica nas práticas contemporâneas de linguagem por parte dos estudantes possa ter lugar, mas permite também que se possa ter em mente mais do que um “usuário da língua/das linguagens”, na direção do que alguns autores vão denominar de designer: alguém que toma algo que já existe (inclusive textos escritos), mescla, remixa, transforma, redistribui, produzindo novos sentidos, processo que alguns autores associam à criatividade.*

Dessa forma, a BNCC procura:

*contemplar à **cultura digital**, diferentes linguagens e diferentes letramentos, desde aqueles basicamente lineares, com baixo nível de hipertextualidade, até aqueles que envolvem a hipermídia.*

Da mesma maneira, imbricada à questão dos multiletramentos considera, como uma de suas premissas, a diversidade cultural:

*Sem aderir a um raciocínio classificatório reducionista, que desconsidera as hibridizações, apropriações e mesclas, é importante contemplar o cânone, o marginal, o culto, o popular, a cultura de massa, a **cultura das mídias**, a **cultura digital**, as culturas infantis e juvenis, de forma a garantir uma ampliação de repertório e uma interação e trato com o diferente.*

O **Eixo Leitura** no contexto da BNCC é tomada em um sentido mais amplo:

*dizendo respeito não somente ao texto escrito, mas também a imagens estáticas (foto, pintura, desenho, esquema, gráfico, diagrama) ou em movimento (filmes, vídeos etc.) e ao som (música), que acompanha e cossignifica em muitos **gêneros digitais**.*

No tratamento das práticas leitoras, quando aborda a reconstrução e reflexão sobre as condições de produção e recepção dos textos:

*Refletir sobre as transformações ocorridas nos campos de atividades em função do desenvolvimento das tecnologias de comunicação e informação, do uso do hipertexto e da hipermídia e do surgimento da Web 2.0: novos gêneros do discurso e novas práticas de linguagem próprias da **cultura digital**, transmutação ou reelaboração dos gêneros em função das transformações pelas quais passam o texto (de formatação e em função da convergência de mídias e do funcionamento hipertextual), novas formas de interação e de compartilhamento de textos/ conteúdos/informações, reconfiguração do papel de leitor, que passa a ser também produtor, dentre outros, como forma de ampliar as possibilidades de participação na **cultura digital** e contemplar os novos e os multiletramentos*

*Analisar as diferentes formas de manifestação da compreensão ativa (réplica ativa) dos textos que circulam nas redes sociais, blogs/microblog, sites e afins e os gêneros que conformam essas práticas de linguagem, como: comentário, carta de leitor, postem rede social, gif, meme, fanfic, vlogs variados, political remix, charge digital, paródias de diferentes tipos, vídeos-minuto, e-zine, fanzine, fanvídeo, vidding, gameplay, walkthrough, detonado, machinima, trailer honesto, playlists comentadas de diferentes tipos etc., de forma a ampliar a compreensão de textos que pertencem a esses gêneros e a possibilitar uma participação mais qualificada do ponto de vista ético, estético e político nas práticas de linguagem da **cultura digital**.*

E quando aborda estratégias e procedimentos de leitura:

*Manejar de forma produtiva a **não linearidade da leitura de hipertextos** e o manuseio de várias janelas, tendo em vista os objetivos de leitura.*

A demanda cognitiva das atividades de leitura deve aumentar progressivamente desde os anos iniciais do Ensino Fundamental até o Ensino Médio. Esta complexidade se expressa pela articulação (entre outros):

*...da consideração da **cultura digital** e das **TDIC** (tecnologias digitais da informação e comunicação).*

No **Eixo da Produção de Textos**, entre os exemplos de práticas de linguagem relacionadas à interação e à autoria, inclui:

*divulgar conhecimentos específicos por meio de um verbete de **enciclopédia digital colaborativa**.*

O tratamento das práticas de produção de textos compreende dimensões inter-relacionadas às práticas de uso e reflexão, tais como:

*Refletir sobre diferentes contextos e situações sociais em que se produzem textos e sobre as diferenças em termos formais, estilísticos e linguísticos que esses contextos determinam, incluindo-se aí a multisssemiose e características*

*da conectividade (uso de hipertextos e hiperlinks, dentre outros, presentes nos textos que circulam em **contexto digital**).*

Ainda, o tratamento das práticas de produção de textos compreende, na alimentação temática:

*Selecionar informações e dados, argumentos e outras referências em **fontes confiáveis impressas e digitais**, para que o texto a ser produzido tenha um nível de aprofundamento adequado e contemple a sustentação das posições defendidas.*

Na BNCC, a organização das práticas de linguagem (leitura de textos, produção de textos, oralidade e análise linguística/semiótica) é por campos de atuação: Campo da vida cotidiana (somente anos iniciais), Campo artístico-literário, Campo das práticas de estudo e pesquisa, Campo jornalístico-midiático e Campo de atuação na vida pública, sendo que esses dois últimos aparecem fundidos nos anos iniciais do Ensino Fundamental, com a denominação Campo da vida pública.

*A **cultura digital** perpassa todos os campos, fazendo surgir ou modificando gêneros e práticas. Por essa razão, optou-se por um tratamento transversal da **cultura digital**, bem como das **TDIC**, articulado a outras dimensões nas práticas em que aparecem. De igual forma, procurou-se contemplar formas de expressão das culturas juvenis, que estão mais evidentes nos campos artístico-literário e jornalístico-midiático, e menos evidentes nos campos de atuação na vida pública e das práticas de estudo e pesquisa, ainda que possam, nesse campo, ser objeto de pesquisa e ainda que seja possível pensar em um vídeo-minuto para apresentar resultados de pesquisa, slides de apresentação que simulem um game ou em formatos de apresentação dados por um número mínimo de imagens que condensam muitas ideias e relações, como acontece em muitas das formas de expressão das culturas juvenis.*

Nas competências específicas da língua portuguesa:

*Mobilizar práticas da **cultura digital**, diferentes linguagens, mídias e ferramentas digitais para expandir as formas de produzir sentidos (nos processos de compreensão e produção), aprender e refletir sobre o mundo e realizar diferentes projetos autorais.*

Ao longo do texto da BNCC sobre a língua Portuguesa no Ensino Fundamental nos anos iniciais e finais são referenciados mais de 70 (setenta) outros pontos onde termos correlatos a aspectos do Pensamento Computacional.

## 9.2.2 Arte

No Ensino Fundamental, o componente curricular Arte está centrado nas seguintes linguagens: as *Artes Visuais*, a *Dança*, a *Música* e o *Teatro*. Cada uma das quatro linguagens do componente curricular constitui uma unidade temática. Além dessas, uma última unidade temática, *Artes Integradas*, explora as relações e

articulações entre as diferentes linguagens e suas práticas, inclusive aquelas possibilitadas pelo uso das **novas tecnologias de informação e comunicação**.

Entre as habilidades dos anos iniciais do Ensino Fundamental, tem-se:

*Explorar diferentes tecnologias e **recursos digitais** (multimeios, animações, jogos eletrônicos, gravações em áudio e vídeo, fotografia, softwares etc.) nos processos de criação artística.*

E, entre as habilidades dos anos finais do Ensino Fundamental, tem-se:

*Desenvolver processos de criação em artes visuais, com base em temas ou interesses artísticos, de modo individual, coletivo e colaborativo, fazendo uso de materiais, instrumentos e **recursos** convencionais, alternativos e **digitais**.*

*Explorar e analisar elementos constitutivos da música (altura, intensidade, timbre, melodia, ritmo etc.), por meio de **recursos tecnológicos** (games e **plataformas digitais**), jogos, canções e práticas diversas de composição/criação, execução e apreciação musicais.*

*Identificar e manipular diferentes **tecnologias e recursos digitais** para acessar, apreciar, produzir, registrar e compartilhar práticas e repertórios artísticos, de modo reflexivo, ético e responsável.*

### 9.2.3 Língua Inglesa

O estudo da língua inglesa tem um *caráter formativo* no sentido em que possibilita “o acesso aos saberes linguísticos necessários para engajamento e participação, contribuindo para o agenciamento crítico dos estudantes e para o exercício da cidadania ativa, além de ampliar as possibilidades de interação e mobilidade, abrindo novos percursos de construção de conhecimentos e de continuidade nos estudos”.

Ensinar inglês com essa finalidade tem, para o currículo, três implicações importantes:

- rever as relações entre língua, território e cultura, na medida em que os falantes de inglês já não se encontram apenas nos países em que essa é a língua oficial;
- ampliar a visão de letramento, ou melhor, dos *multiletramentos*, concebida também nas **práticas sociais do mundo digital** – no qual saber a língua inglesa potencializa as possibilidades de participação e circulação – que aproximam e entrelaçam diferentes semioses e linguagens (verbal, visual, corporal, audiovisual), em um contínuo processo de significação contextualizado, dialógico e ideológico;
- compreender que determinadas crenças – como um “inglês melhor” para se ensinar, ou um “nível de proficiência” a ser alcançado pelo aluno – precisam ser relativizadas.



No *Eixo Leitura*, destaca que:

*O trabalho com gêneros verbais e híbridos, potencializados principalmente pelos **meios digitais**, possibilita vivenciar, de maneira significativa e situada, diferentes modos de leitura, bem como diferentes objetivos de leitura.*

Entre as competências específicas de Língua Inglesa, tem-se que:

*Comunicar-se na língua inglesa, por meio do uso variado de linguagens em **mídias impressas ou digitais**, reconhecendo-a como ferramenta de acesso ao conhecimento, de ampliação das perspectivas e de possibilidades para a compreensão dos valores e interesses de outras culturas e para o exercício do protagonismo social.*

No *Eixo Conhecimentos Linguísticos*, entre os objetos de conhecimento, tem-se:

*Usos de linguagem em **meio digital**: “internetês”.*

Entre as Habilidades da Língua Inglesa – 9º ano, tem-se:

*Reconhecer, nos novos **gêneros digitais** (blogues, mensagens instantâneas, tweets, entre outros), novas formas de escrita na constituição das mensagens.*

## 9.2.4 Matemática

A BNCC explicitamente referencia Pensamento Computacional na Área da Matemática quando se refere aos processos matemáticos:

*Os processos matemáticos de resolução de problemas, de investigação, de desenvolvimento de projetos e da modelagem podem ser citados como formas privilegiadas da atividade matemática, motivo pelo qual são, ao mesmo tempo, objeto e estratégia para a aprendizagem ao longo de todo o Ensino Fundamental. Esses processos de aprendizagem são potencialmente ricos para o desenvolvimento de competências fundamentais para o letramento matemático (raciocínio, representação, comunicação e argumentação) e para o desenvolvimento do **pensamento computacional**.*

Entre as competências específicas da matemática, tem-se:

*Utilizar processos e ferramentas matemáticas, inclusive **tecnologias digitais** disponíveis, para modelar e resolver problemas cotidianos, sociais e de outras áreas de conhecimento, validando estratégias e resultados.*

*Enfrentar situações-problema em múltiplos contextos, incluindo-se situações imaginadas, não diretamente relacionadas com o aspecto prático-utilitário, expressar suas respostas e sintetizar conclusões, utilizando diferentes registros e linguagens (gráficos, tabelas, esquemas, além de texto escrito na língua materna e outras linguagens para descrever **algoritmos, como fluxogramas, e dados**)*

Na unidade temática Números, nos Anos Iniciais do Ensino Fundamental, tem-se:

*No tocante aos cálculos, espera-se que os alunos desenvolvam diferentes estratégias para a obtenção dos resultados, sobretudo por estimativa e cálculo mental, além de **algoritmos** e uso de calculadoras.*

Um aspecto a considerar na aprendizagem na unidade temática Álgebra, como também aquelas relacionadas a Números, Geometria e Probabilidade e Estatística é que:

*podem contribuir para o desenvolvimento do **pensamento computacional** dos alunos, tendo em vista que eles precisam ser capazes de traduzir uma situação dada em outras linguagens, como transformar situações-problema, apresentadas em língua materna, em fórmulas, tabelas e gráficos e vice-versa.*

Nesse contexto, ao tratar de aulas de matemática, destaca que:

*Associado ao **pensamento computacional**, cumpre salientar a importância dos **algoritmos** e de seus **fluxogramas**, que podem ser objetos de estudo nas aulas de Matemática. Um algoritmo é uma sequência finita de procedimentos que permite resolver um determinado problema. Assim, o algoritmo é a decomposição de um procedimento complexo em suas partes mais simples, relacionando-as e ordenando-as, e pode ser representado graficamente por um fluxograma. A **linguagem algorítmica** tem pontos em comum com a linguagem algébrica, sobretudo em relação ao conceito de variável.*

*Outra habilidade relativa à álgebra que mantém estreita relação com o **pensamento computacional** é a identificação de **padrões** para se estabelecer generalizações, propriedades e algoritmos.*

Relativamente às habilidades da Área Matemática, para os vários anos tem-se:

a) 1º ano

*Descrever, após o reconhecimento e a explicitação de um padrão (ou regularidade), os elementos ausentes em sequências recursivas de números naturais, objetos ou figuras.*

b) 2º ano

*Descrever um padrão (ou regularidade) de sequências repetitivas e de **sequências recursivas**, por meio de palavras, símbolos ou desenhos.*

*Descrever os elementos ausentes em sequências repetitivas e em **sequências recursivas** de números naturais, objetos ou figuras.*

c) 3º ano

*Reconhecer figuras congruentes, usando sobreposição e desenhos em malhas quadriculadas ou triangulares, incluindo o uso de **tecnologias digitais**.*

d) 4º ano

*Resolver e elaborar problemas com números naturais envolvendo adição e subtração, utilizando estratégias diversas, como cálculo, cálculo mental e **algoritmos**, além de fazer estimativas do resultado.*

*Resolver e elaborar problemas envolvendo diferentes significados da multiplicação (adição de parcelas iguais, organização retangular e proporcionalidade), utilizando estratégias diversas, como cálculo por estimativa, cálculo mental e **algoritmos**.*

*Resolver e elaborar problemas de divisão cujo divisor tenha no máximo dois algarismos, envolvendo os significados de repartição equitativa e de medida, utilizando estratégias diversas, como cálculo por estimativa, cálculo mental e **algoritmos**.*

*Realizar pesquisa envolvendo variáveis categóricas e numéricas e organizar dados coletados por meio de tabelas e gráficos de colunas simples ou agrupadas, com e sem uso de **tecnologias digitais**.*

e) 5º ano

*Resolver e elaborar problemas de adição e subtração com números naturais e com números racionais, cuja representação decimal seja finita, utilizando estratégias diversas, como cálculo por estimativa, cálculo mental e **algoritmos**.*

*Resolver e elaborar problemas de multiplicação e divisão com números naturais e com números racionais cuja representação decimal é finita (com multiplicador natural e divisor natural e diferente de zero), utilizando estratégias diversas, como cálculo por estimativa, cálculo mental e **algoritmos**.*

*Reconhecer, nomear e comparar polígonos, considerando lados, vértices e ângulos, e desenhá-los, utilizando material de desenho ou **tecnologias digitais**.*

*Reconhecer a congruência dos ângulos e a proporcionalidade entre os lados correspondentes de figuras poligonais em situações de ampliação e de redução em malhas quadriculadas e usando **tecnologias digitais**.*

*Realizar pesquisa envolvendo variáveis categóricas e numéricas, organizar dados coletados por meio de tabelas, gráficos de colunas, pictóricos e de linhas, com e sem uso de **tecnologias digitais**, e apresentar texto escrito sobre a finalidade da pesquisa e a síntese dos resultados.*

f) 6º ano

*Construir **algoritmo** em linguagem natural e representá-lo por **fluxograma** que indique a resolução de um problema simples (por exemplo, se um número natural qualquer é par).*

*Construir figuras planas semelhantes em situações de ampliação e de redução, com o uso de malhas quadriculadas, plano cartesiano ou **tecnologias digitais**.*

Construir **algoritmo** para resolver situações passo a passo (como na construção de dobraduras ou na indicação de deslocamento de um objeto no plano segundo pontos de referência e distâncias fornecidas etc.).

Determinar medidas da abertura de ângulos, por meio de transferidor e/ou **tecnologias digitais**.

Interpretar e desenvolver **fluxogramas** simples, identificando as relações entre os objetos representados (por exemplo, posição de cidades considerando as estradas que as unem, hierarquia dos funcionários de uma empresa etc.).

g) 7º ano

Resolver e elaborar problemas com números naturais, envolvendo as noções de divisor e de múltiplo, podendo incluir máximo divisor comum ou mínimo múltiplo comum, por meio de estratégias diversas, sem a aplicação de **algoritmos**.

Resolver um mesmo problema utilizando diferentes **algoritmos**.

Representar por meio de um fluxograma os passos utilizados para resolver um grupo de problemas

Classificar sequências em **recursivas** e não **recursivas**, reconhecendo que o conceito de **recursão** está presente não apenas na matemática, mas também nas artes e na literatura.

Descrever, por escrito e por meio de um **fluxograma**, um **algoritmo** para a construção de um triângulo qualquer, conhecidas as medidas dos três lados.

Descrever, por escrito e por meio de um **fluxograma**, um **algoritmo** para a construção de um polígono regular (como quadrado e triângulo equilátero), conhecida a medida de seu lado.

h) 8º ano

Resolver e elaborar problemas, envolvendo cálculo de porcentagens, incluindo o uso de **tecnologias digitais**.

Identificar a regularidade de uma sequência numérica ou figural **não recursiva** e construir um **algoritmo** por meio de um **fluxograma** que permita indicar os números ou as figuras seguintes.

Identificar a regularidade de uma sequência numérica **recursiva** e construir um **algoritmo** por meio de um **fluxograma** que permita indicar os números seguintes.

Descrever, por escrito e por meio de um **fluxograma**, um **algoritmo** para a construção de um hexágono regular de qualquer área, a partir da medida do ângulo central e da utilização de esquadros e compasso.

i) 9º ano

*Resolver e elaborar problemas que envolvam porcentagens, com a ideia de aplicação de percentuais sucessivos e a determinação das taxas percentuais, preferencialmente com o uso de **tecnologias digitais**, no contexto da educação financeira.*

*Descrever, por escrito e por meio de um **fluxograma**, um **algoritmo** para a construção de um hexágono regular de qualquer área, a partir da medida do ângulo central e da utilização de esquadros e compasso.*

## 9.2.5 Ciências da Natureza

A sociedade contemporânea está fortemente organizada com base no desenvolvimento científico e tecnológico. Ciência e tecnologia vêm se desenvolvendo de forma integrada com os modos de vida que as diversas sociedades humanas organizaram ao longo da história

Portanto, a área de Ciências da Natureza tem um compromisso com o desenvolvimento do **letramento científico**, que envolve a capacidade de compreender e interpretar o mundo (natural, social e tecnológico), mas também de transformá-lo com base nos **aportes teóricos e processuais das ciências**.

Nessa perspectiva, Ciências da Natureza, precisa assegurar o acesso à diversidade de conhecimentos científicos, bem como a aproximação gradativa aos principais **processos, práticas e procedimentos da investigação científica**.

Dentro da ideia de que o processo investigativo é central na formação do aluno, destaca-se, relativamente ao levantamento, análise e representação:

*Desenvolver e utilizar **ferramentas, inclusive digitais**, para coleta, análise e representação de dados (imagens, esquemas, tabelas, gráficos, quadros, diagramas, mapas, **modelos, representações de sistemas, fluxogramas, mapas conceituais, simulações, aplicativos etc.**).*

*Desenvolver soluções para problemas cotidianos usando diferentes **ferramentas, inclusive digitais**.*

No que se refere às competências específicas de Ciências da Natureza, tem-se:

*Analisar, compreender e explicar características, fenômenos e processos relativos ao **mundo natural, social e tecnológico (incluindo o digital)**, como também as relações que se estabelecem entre eles, exercitando a curiosidade para fazer perguntas, buscar respostas e criar soluções (inclusive tecnológicas) com base nos conhecimentos das Ciências da Natureza.*

*Utilizar diferentes **linguagens e tecnologias digitais de informação e comunicação** para se comunicar, acessar e disseminar informações, produzir conhecimentos e resolver problemas das Ciências da Natureza de forma crítica, significativa, reflexiva e ética.*

Relativamente às habilidades da Área Ciência da Natureza, para os vários anos, tem-se:

a) 5º ano

*Identificar algumas constelações no céu, com o apoio de recursos (como mapas celestes e **aplicativos digitais**, entre outros), e os períodos do ano em que elas são visíveis no início da noite.*

b) 6º ano

*Concluir, com base na análise de ilustrações e/ou **modelos** (físicos ou **digitais**), que os organismos são um complexo arranjo de sistemas com diferentes níveis de organização*

c) 7º ano

*Analisar historicamente o uso da **tecnologia, incluindo a digital**, nas diferentes dimensões da vida humana, considerando indicadores ambientais e de qualidade de vida.*

## 9.2.6 Ciências Humanas

No decorrer do Ensino Fundamental, os **procedimentos de investigação** em Ciências Humanas devem contribuir para que os alunos desenvolvam a capacidade de observação de diferentes indivíduos, situações e objetos que trazem à tona dinâmicas sociais em razão de sua própria natureza (tecnológica, morfológica, funcional).

No que se refere às competências específicas de Ciências Humanas, tem-se:

*Analisar o **mundo social, cultural e digital** e o meio técnico-científico-informacional com base nos conhecimentos das Ciências Humanas, considerando suas variações de significado no tempo e no espaço, para intervir em situações do cotidiano e se posicionar diante de problemas do mundo contemporâneo.*

*Utilizar as linguagens cartográfica, gráfica e iconográfica e diferentes gêneros textuais e **tecnologias digitais de informação e comunicação** no desenvolvimento do raciocínio espaço-temporal relacionado a localização, distância, direção, duração, simultaneidade, sucessão, ritmo e conexão.*

Quando trata da Geografia nos Anos Finais do Ensino Fundamental, afirma que:

*é preciso que os alunos ampliem seus conhecimentos sobre o uso do espaço em diferentes situações geográficas regidas por normas e leis historicamente instituídas, compreendendo a transformação do espaço em território usado – espaço da ação concreta e das relações desiguais de poder, considerando*



também o **espaço virtual** proporcionado pela **rede mundial de computadores** e das **geotecnologias**.

Relativamente às habilidades de Geografia, tem-se, para o 7º ano:

*Interpretar e elaborar mapas temáticos e históricos, inclusive utilizando **tecnologias digitais**, com informações demográficas e econômicas do Brasil (cartogramas), identificando padrões espaciais, regionalizações e analogias espaciais.*

No que se refere às competências específicas de História, tem-se:

*Produzir, avaliar e utilizar **tecnologias digitais de informação e comunicação** de modo crítico, ético e responsável, compreendendo seus significados para os diferentes grupos ou estratos sociais.*

Relativamente às habilidades de História, para os vários anos, tem-se:

a) 4º ano

*Identificar as transformações ocorridas nos meios de comunicação (cultura oral, imprensa, rádio, televisão, cinema, **internet e demais tecnologias digitais de informação e comunicação**) e discutir seus significados para os diferentes grupos ou estratos sociais.*

b) 9º ano

*Analisar as transformações nas relações políticas locais e globais geradas pelo desenvolvimento das **tecnologias digitais de informação e comunicação**.*

## 9.3 Etapa do Ensino Médio

A BNCC do Ensino Médio se organiza em continuidade ao proposto para a Educação Infantil e o Ensino Fundamental, centrada no desenvolvimento de competências e orientada pelo princípio da educação integral. Portanto, as competências gerais da Educação Básica orientam igualmente as aprendizagens dessa etapa.

Quando trata das *finalidades do Ensino Médio na contemporaneidade*, afirma que a escola precisa se estruturar para, entre outros aspectos:

*garantir a contextualização dos conhecimentos, articulando as dimensões do trabalho, da **ciência**, da **tecnologia** e da cultura;*

*viabilizar o acesso dos estudantes às **bases científicas e tecnológicas** dos processos de produção do mundo contemporâneo, **relacionando teoria e prática** – ou o conhecimento teórico à resolução de problemas da realidade social, cultural ou natural.*

Adiante, complementa que, além da escola viabilizar o acesso, o Ensino Médio deve garantir aos estudantes a *compreensão dos fundamentos científico-tecnológicos*, e que deve possibilitar aos estudantes:

*compreender e utilizar os conceitos e teorias que compõem a **base do conhecimento científico-tecnológico**, bem como os **procedimentos metodológicos e suas lógicas**;*

*apropriar-se das **linguagens científicas** e utilizá-las na comunicação e na disseminação desses conhecimentos;*

*apropriar-se das **linguagens das tecnologias digitais** e tornar-se fluentes em sua utilização.*

Especificamente, na progressão do Ensino Fundamental para o Ensino Médio relativamente à Área da Matemática e suas Tecnologias:

*A área de Matemática, no Ensino Fundamental, centra-se na compreensão de conceitos e procedimentos em seus diferentes campos e no desenvolvimento do **pensamento computacional**, visando à resolução e formulação de problemas em contextos diversos. No Ensino Médio, na área de Matemática e suas Tecnologias, os estudantes devem consolidar os conhecimentos desenvolvidos na etapa anterior e agregar novos, ampliando o leque de recursos para resolver problemas mais complexos, que exijam maior reflexão e abstração.*

Quando trata dos itinerários formativos – estratégicos para a flexibilização da organização curricular do Ensino Médio, pois possibilitam opções de escolha aos estudantes, tem-se que:

*matemática e suas tecnologias: aprofundamento de conhecimentos estruturantes para aplicação de diferentes conceitos matemáticos em contextos sociais e de trabalho, estruturando arranjos curriculares que permitam estudos em resolução de problemas e análises complexas, funcionais e não-lineares, análise de dados estatísticos e probabilidade, geometria e topologia, **robótica, automação, inteligência artificial, programação, jogos digitais, sistemas dinâmicos**, dentre outros, considerando o contexto local e as possibilidades de oferta pelos sistemas de ensino*

### 9.3.1 As Tecnologias Digitais e a Computação

A BNCC do Ensino Médio dedica toda uma seção à questão das Tecnologias Digitais e a Computação e os principais aspectos são aqui destacados.

Quando afirma que contemporaneidade é fortemente marcada pelo desenvolvimento tecnológico:

*Tanto a **computação** quanto as **tecnologias digitais de informação e comunicação (TDIC)** estão cada vez mais presentes na vida de todos.*

*Grande parte das **informações** produzidas pela humanidade está **armazenada digitalmente**. Isso denota o quanto o mundo produtivo e o cotidiano estão sendo movidos por **tecnologias digitais**, situação que tende a se acentuar fortemente no futuro.*

Ao destacar que essa constante transformação ocasionada pelas tecnologias, bem como sua repercussão na forma como as pessoas se comunicam, impacta diretamente no funcionamento da sociedade, destaca que:

*É preciso garantir aos jovens aprendizagens para atuar em uma sociedade em constante mudança, prepará-los para **profissões que ainda não existem**, para usar **tecnologias que ainda não foram inventadas** e para **resolver problemas que ainda não conhecemos**. Certamente, grande parte das futuras profissões envolverá, direta ou indiretamente, **computação e tecnologias digitais**.*

Diferentes dimensões que caracterizam a computação e as tecnologias digitais são tematizadas, tanto no que diz respeito a conhecimentos e habilidades, quanto a atitudes e valores:

**pensamento computacional:** *envolve as capacidades de compreender, analisar, definir, modelar, resolver, comparar e automatizar problemas e suas soluções, de forma metódica e sistemática, por meio do desenvolvimento de algoritmos;*

**mundo digital:** *envolve as aprendizagens relativas às formas de processar, transmitir e distribuir a informação de maneira segura e confiável em diferentes artefatos digitais – tanto físicos (computadores, celulares, tablets etc.) como virtuais (internet, redes sociais e nuvens de dados, entre outros) –, compreendendo a importância contemporânea de codificar, armazenar e proteger a informação;*

**cultura digital:** *envolve aprendizagens voltadas a uma participação mais consciente e democrática por meio das tecnologias digitais, o que supõe a compreensão dos impactos da revolução digital e dos avanços do mundo digital na sociedade contemporânea, a construção de uma atitude crítica, ética e responsável em relação à multiplicidade de ofertas midiáticas e digitais, aos usos possíveis das diferentes tecnologias e aos conteúdos por elas veiculados, e, também, à fluência no uso da tecnologia digital para expressão de soluções e manifestações culturais de forma contextualizada e crítica.*

Essas dimensões também foram contempladas nos objetivos de aprendizagem e desenvolvimento da Educação Infantil e nas competências específicas e habilidades dos diferentes componentes curriculares do Ensino Fundamental, respeitadas as características dessas etapas. No Ensino Médio, dada a intrínseca relação entre as culturas juvenis e a **cultura digital**, torna-se imprescindível ampliar e aprofundar as aprendizagens construídas nas etapas anteriores. Afinal, os jovens estão dinamicamente inseridos na **cultura digital**, como

consumidores e como protagonistas. Portanto, na BNCC dessa etapa, o foco passa a estar:

*no reconhecimento das potencialidades das **tecnologias digitais** para a realização de uma série de atividades relacionadas a todas as áreas do conhecimento, a diversas práticas sociais e ao mundo do trabalho.*

São definidas competências e habilidades, nas diferentes áreas, que permitem aos estudantes:

*buscar **dados e informações** de forma crítica nas diferentes **mídias, inclusive as sociais**, analisando as vantagens do uso e da evolução da tecnologia na sociedade atual, como também seus riscos potenciais;*

*apropriar-se das **linguagens da cultura digital**, dos novos letramentos e dos multiletramentos para explorar e produzir conteúdos em diversas mídias, ampliando as possibilidades de acesso à ciência, à tecnologia, à cultura e ao trabalho;*

*usar diversas **ferramentas de software e aplicativos** para compreender e produzir conteúdos em diversas mídias, simular fenômenos e processos das diferentes áreas do conhecimento, e elaborar e explorar diversos registros de representação matemática;*

*utilizar, propor e/ou implementar soluções (processos e produtos) envolvendo diferentes tecnologias, para identificar, analisar, modelar e solucionar problemas complexos em diversas áreas da vida cotidiana, explorando de forma efetiva o **raciocínio lógico**, o **pensamento computacional**, o espírito de investigação e a criatividade.*

### 9.3.2 Línguas e suas Tecnologias

A área de Línguas e suas Tecnologias tem a responsabilidade de propiciar oportunidades para a consolidação e a ampliação das habilidades de uso e de reflexão sobre as linguagens – artísticas, corporais e verbais (oral ou visual-motora, como Libras, e escrita) –, que são objeto de seus diferentes componentes (Arte, Educação Física, Língua Inglesa e Língua Portuguesa).

Para orientar uma abordagem integrada dessas linguagens e de suas práticas, a área:

*propõe que os estudantes possam vivenciar experiências significativas com práticas de linguagem em diferentes **mídias** (impressa, **digital**, analógica), situadas em campos de atuação social diversos*

Relativamente à Arte, é fundamental que os estudantes possam assumir o papel de protagonistas como apreciadores e como artistas, criadores e curadores:

*Assim, devem poder fazer uso de materiais, instrumentos e **recursos** convencionais, alternativos e **digitais**, em diferentes meios e tecnologias.*

Por sua vez, a Língua Inglesa:

*além dessa visão intercultural e “desterritorializada” da língua inglesa – que, em seus usos, sofre transformações oriundas das identidades plurais de seus falantes –, consideraram-se também as práticas sociais do **mundo digital**, com ênfase em multiletramentos.*

*permite aos estudantes explorar a presença da multiplicidade de seu uso na **cultura digital**, nas culturas juvenis e em estudos e pesquisas, como também ampliar suas perspectivas em relação à sua vida pessoal e profissional.*

Para orientar uma abordagem integrada dessas linguagens e de suas práticas, a área:

*propõe que os estudantes possam vivenciar experiências significativas com práticas de linguagem em diferentes **mídias** (impressa, **digital**, analógica), situadas em campos de atuação social diversos.*

Considerando que uma semiose é um sistema de signos em sua organização própria é importante que os jovens, ao explorarem as possibilidades expressivas das diversas linguagens:

*possam realizar reflexões que envolvam o exercício de análise de elementos discursivos, composicionais e formais de enunciados nas diferentes semioses – visuais, sonoras, verbais e corporais. Afinal, muito por efeito das novas **tecnologias digitais da informação e da comunicação (TDIC)**, os textos e discursos atuais organizam-se de maneira híbrida e multissemiótica, incorporando diferentes sistemas de signos em sua constituição.*

*Assim, propostas de trabalho que potencializem aos estudantes o acesso a saberes sobre o **mundo digital** e a práticas da **cultura digital** devem também ser priorizadas, já que, direta ou indiretamente, impactam seu dia a dia nos vários campos de atuação social e despertam seu interesse e sua identificação com as **TDIC**. Sua utilização na escola não só possibilita maior apropriação técnica e crítica desses recursos, como também é determinante para uma aprendizagem significativa e autônoma pelos estudantes.*

Nessa perspectiva, para além da cultura da palavra escrita, que deve continuar tendo centralidade na educação escolar:

*é preciso considerar a cultura digital, os multiletramentos e os novos letramentos, entre outras denominações que procuram designar novas práticas sociais de linguagem.*

Não são apenas novos gêneros que surgem ou se transformam (como *post*, *tweet*, meme, *mashup*, *playlist* comentada, reportagem multimidiática, relato multimidiático, *vlog*, videominuto, *political remix*, tutoriais em vídeo, entre outros), mas também de:

*novas ações, procedimentos e atividades (curtir, comentar, redistribuir, compartilhar, taggear, seguir/ ser seguido, remediar, remixar, curar,*

*coleccionar/descoleccionar, colaborar etc.) que supõem o desenvolvimento de outras habilidades.*

Não se trata de substituição ou de simples convivência de mídias, mas de levar em conta:

*como a coexistência e a convergência das mídias transformam as próprias mídias e seus usos e potencializam novas possibilidades de construção de sentidos.*

Ao alterar o fluxo de comunicação de um para muitos – como na TV, rádio e mídia impressa – para de muitos para muitos:

*as possibilidades advindas das **tecnologias digitais de informação e comunicação (TDIC)** permitem que todos sejam produtores em potencial, imbricando mais ainda as práticas de leitura e produção (e de consumo e circulação/recepção). Não só é possível para qualquer um redistribuir ou comentar notícias, artigos de opinião, postagens em vlogs, machinemas, AMVs e outros textos, mas também escrever ou performar e publicar textos e enunciados variados, o que potencializa a participação.*

Em que pese o potencial participativo e colaborativo das **TDIC**, a abundância de informações e produções requer, ainda, que os estudantes:

*desenvolvam habilidades e critérios de curadoria e de apreciação ética e estética, considerando, por exemplo, a profusão de notícias falsas (fake news), de pós-verdades, do cyberbullying e de discursos de ódio nas mais variadas instâncias da internet e demais mídias.*

Entre as competências específicas de Linguagens e suas Tecnologias, tem-se:

*Mobilizar práticas de **linguagem no universo digital**, considerando as dimensões técnicas, críticas, criativas, éticas e estéticas, para expandir as formas de produzir sentidos, de engajar-se em práticas autorais e coletivas, e de aprender a aprender nos campos da ciência, cultura, trabalho, informação e vida pessoal e coletiva.*

Relativamente a essa competência específica, as seguintes habilidades são apresentadas:

*Explorar **tecnologias digitais da informação e comunicação (TDIC)**, compreendendo seus princípios e funcionalidades, e utilizá-las de modo ético, criativo, responsável e adequado a práticas de linguagem em diferentes contextos*

*Avaliar o impacto das **tecnologias digitais da informação e comunicação (TDIC)** na formação do sujeito e em suas práticas sociais, para fazer uso crítico dessa mídia em práticas de seleção, compreensão e produção de discursos em ambiente digital.*



*Utilizar diferentes linguagens, mídias e ferramentas digitais em processos de produção coletiva, colaborativa e projetos autorais em ambientes digitais.*

*Apropriar-se criticamente de processos de pesquisa e busca de informação, por meio de ferramentas e dos novos formatos de produção e distribuição do conhecimento na cultura de rede.*

Ao longo do texto da BNCC sobre a Linguagens e suas Tecnologias no Ensino Médio são referenciados cerca de 30 (trinta) outros pontos onde termos correlatos a aspectos do Pensamento Computacional são encontrados.

### 9.3.3 Matemática e suas Tecnologias

A BNCC da área de Matemática e suas Tecnologias propõe a consolidação, a ampliação e o aprofundamento das aprendizagens essenciais desenvolvidas no Ensino Fundamental. Em particular, a BNCC propõe que os estudantes utilizem tecnologias, como calculadoras e planilhas eletrônicas, desde os anos iniciais do Ensino Fundamental. Tal valorização possibilita que, ao chegarem aos anos finais, eles possam ser estimulados a desenvolver o **pensamento computacional**, por meio da interpretação e da elaboração de **algoritmos**, incluindo aqueles que podem ser representados por **fluxogramas**.

Em continuidade a essas aprendizagens, no Ensino Médio o foco é a construção de uma visão integrada da Matemática, aplicada à realidade. Consequentemente, é preciso levar em conta as vivências cotidianas– impactadas de diferentes maneiras pelos avanços tecnológicos, pelas exigências do mercado de trabalho, pela potencialidade das **mídias sociais**, entre outros.

*Nesse contexto, destaca-se ainda a importância do **recurso a tecnologias digitais e aplicativos** tanto para a investigação matemática como para dar continuidade ao desenvolvimento do **pensamento computacional**, iniciado no Ensino Fundamental.*

A área de Matemática e suas Tecnologias tem a responsabilidade de aproveitar todo o potencial já constituído no Ensino Fundamental, bem como promover a ampliação do letramento matemático. Assim, novos conhecimentos devem estimular processos mais elaborados de reflexão e de abstração, que deem sustentação a modos de pensar que permitam aos estudantes formular e resolver problemas em diversos contextos com mais autonomia e recursos matemáticos. Para que esses propósitos se concretizem, os estudantes devem:

*desenvolver habilidades relativas aos **processos de investigação, de construção de modelos e de resolução de problemas**.*

Relativamente às competências específicas de Matemática e suas tecnologias, tem-se:

*Utilizar estratégias, conceitos, definições e procedimentos matemáticos para interpretar, **construir modelos e resolver problemas** em diversos contextos,*

*analisando a plausibilidade dos resultados e a adequação das soluções propostas, de modo a construir argumentação consistente.*

*Compreender e utilizar, com flexibilidade e precisão, diferentes registros de representação matemáticos (algébrico, geométrico, estatístico, **computacional** etc.), na busca de solução e comunicação de resultados de problemas.*

Ao longo do texto da BNCC sobre a Matemática e suas Tecnologias no Ensino Médio são referenciados cerca de 20 (vinte) outros pontos onde termos correlatos a aspectos do Pensamento Computacional são encontrados.

### 9.3.4 Ciências da Natureza e suas Tecnologias

Nas sociedades contemporâneas, muitos são os exemplos da presença da Ciência e da Tecnologia, e de sua influência no modo como vivemos, pensamos e agimos. Nesse contexto, a Ciência e a Tecnologia tendem a ser encaradas não somente como ferramentas capazes de **solucionar problemas**, tanto os dos indivíduos como os da sociedade, mas também como uma abertura para novas visões de mundo.

Diante da diversidade dos usos e da divulgação do conhecimento científico e tecnológico na sociedade contemporânea, torna-se fundamental a apropriação, por parte dos estudantes, de linguagens específicas da área das Ciências da Natureza e suas Tecnologias. O Ensino Médio deve, portanto, promover a compreensão e a apropriação desse modo de “se expressar” próprio das Ciências da Natureza pelos estudantes.

Essa perspectiva está presente nas competências específicas e habilidades da área. Das três competências específicas, destaca-se:

*Investigar situações-problema e avaliar aplicações do conhecimento científico e tecnológico e suas implicações no mundo, utilizando procedimentos e linguagens próprios das Ciências da Natureza, para propor soluções que considerem demandas locais, regionais e/ou globais, e comunicar suas descobertas e conclusões a públicos variados, em diversos contextos e por meio de diferentes **mídias e tecnologias digitais de informação e comunicação (TDIC)**.*

No detalhamento das competências, habilidades e outros aspectos ao longo do texto da área de Ciências da Natureza e suas Tecnologias, são referenciados cerca de 15 (quinze) outros pontos onde termos correlatos a aspectos do Pensamento Computacional são encontrados.

### 9.3.5 Ciências Humanas e Sociais Aplicadas

A BNCC da área de Ciências Humanas e Sociais Aplicadas propõe a ampliação e o aprofundamento das aprendizagens essenciais desenvolvidas no Ensino Fundamental.

É necessário que a Área de Ciências Humanas e Sociais Aplicadas:

*favoreça o protagonismo juvenil investindo para que os estudantes sejam capazes de mobilizar diferentes linguagens (textuais, imagéticas, artísticas, gestuais, **digitais**, tecnológicas, gráficas, cartográficas etc.), valorizar os trabalhos de, recorrer a diferentes formas de registros e engajar-se em práticas cooperativas, para a **formulação e resolução de problemas**.*

Referente às habilidades das competências específicas da área, tem-se:

*Identificar, contextualizar e criticar tipologias evolutivas e oposições dicotômicas (cidade/campo, cultura/ natureza, civilizados/bárbaros, razão/emoção, material/**virtual** etc.), explicitando suas ambiguidades.*

*Utilizar as linguagens cartográfica, gráfica e iconográfica, diferentes gêneros textuais e **tecnologias digitais de informação e comunicação** de forma crítica, significativa, reflexiva e ética nas diversas práticas sociais, incluindo as escolares, para se comunicar, acessar e difundir informações, produzir conhecimentos, resolver problemas e exercer protagonismo e autoria na vida pessoal e coletiva.*

## 9.4 A BNCC e o Ensino de Computação Através do PC

Segundo Nunes (Nunes 2017), poder-se-ia introduzir na BNCC os seguintes conteúdos:

<i>Nível de Ensino</i>		<i>Objetivos da aprendizagem</i>
Ensino Fundamental	1º ao 3º ano	Habilidades Cognitivas
	4º ao 6º ano	Matemática Discreta e Lógica,
	7º ao 9º ano	Fundamentos da Ciência da Computação, História da Ciência da Computação e Análise Combinatória
Ensino Médio	10º ao 12º ano	Linguagens de Programação e Estruturas de Dados

Tabela 9.1: Proposta de Nunes 2011

Fonte: (Nunes 2011)

Essa proposta de Nunes traz exatamente a ideia de que não existe uma dissociação entre Computação e o PC. Esse enfoque torna-se importante quando

parece existir uma perigosa dissociação entre PC e Computação, como se pode perceber em vários estudos citados neste texto. Se, em uma próxima BNCC, Computação se tornar uma área de conhecimento, o PC entra através da Computação e pode ser integrado com todas as demais áreas.

Ainda existem iniciativas do ensino universitário para que o PC seja inserido nas escolas brasileiras. Em relatório publicado pela CAPES, aponta-se a necessidade de estudar o assunto e sua adoção (Navaux *et al*, 2016).

Diversas iniciativas de introdução ao PC têm sido realizadas nos últimos anos, envolvendo escolas e instituições de ensino superior (público e privado) em diferentes níveis da educação, como por exemplo: Barcelos e Silveira (2012), Scaico *et al* (2012), Barreto (2013), França e Amaral (2013), Ribeiro *et al* (2013), Andrade *et al* (2013), Viel *et al* (2014), Campos *et al* (2014), Kologeski *et al* (2016), dentre outros.

## 9.5 Análise Apresentada pela SBC para Itens Específicos da BNCC

Observe-se que, no texto que segue, a análise apresentada pela Sociedade Brasileira de Computação (SBC) foi realizada sobre versão anterior da BNCC e não sobre a utilizada neste texto (ver Item 9 - Integração na Educação Básica), a qual baseou a discussão anterior neste capítulo.

Em seu documento técnico (SBC, 2018b), a SBC comenta a especificidade e a contextualização das atividades sugeridas na BNCC. Muitas das atividades constantes na BNCC fazem parte do referencial bibliográfico do PC. Entretanto, sua vinculação a determinados temas, da Matemática, por exemplo, é criticada pela SBC. Da mesma forma, o uso de estratégias atualmente consideradas ultrapassadas no ensino de Computação, também é apontada na análise da SBC.

A seguir faz-se uma apresentação dos posicionamentos da SBC (SBC 2018b) frente ao proposto na BNCC, sobre o tema PC. Considera-se relevante apresentar a visão da SBC, pois essa entidade é brasileira e, portanto, conhecedora do contexto do País. O texto a seguir apresenta basicamente as críticas feitas pela SBC às propostas da BNCC. O documento base para tais comentários é: Nota Técnica da Sociedade Brasileira de Computação sobre a BNCC9EF e a BNCC-EM, de 2018.

A SBC apresenta críticas e contribuições específicas para diferentes pontos da proposta, como apresentado a seguir, tanto para a BNCC-EF, como para BNCC-EM. Ainda, segundo a SBC, o documento foi elaborado:

*com o objetivo de demonstrar sua inadequação e solicitar providências para uma correspondente revisão que, se não realizada, pode prejudicar o ensino na Educação Básica no Brasil.*

O documento apresenta, primeiro, uma análise das habilidades relacionadas à computação na BNCC ensino fundamental. Nele, a SBC divide o tema entre PC e Mundo digital e discute as 11 habilidades apresentadas, na versão 3, da BNCC. As principais considerações da SBC referentes a esses pontos são transcritas a seguir:

A construção de algoritmos não é ensinada: Não há nenhuma habilidade ou objeto de conhecimento relacionado ao aprendizado de construção de algoritmos, que é o tópico central do PC. É ingênuo acreditar que o aluno aprenderá somente “fazendo”, sem ser apresentado de forma sistemática às abstrações necessárias e nem às técnicas de construção de algoritmos.

Linguagem muito específica: É sugerida uma linguagem muito específica para a representação de algoritmos (fluxograma), o que é inadequado para uma base comum curricular, que deve deixar a escolha de linguagens específicas para as implementações. Na área de Computação, como surgem novas linguagens para representar algoritmos com grande frequência, não se define linguagens específicas nem em currículos (que são mais concretos que diretrizes).

Linguagem inadequada: Fluxograma é uma linguagem criada na década de 60/70, não é uma linguagem que segue o paradigma de programação estruturada e não estimula o uso das principais técnicas de solução de problemas através de algoritmos (decomposição, generalização, transformação). A inclusão de conceitos como “fluxograma” no Ensino Fundamental não somente prejudica o desenvolvimento do PC, bem como certamente trará graves problemas na aprendizagem de algoritmos.

Habilidades mal formuladas: Várias habilidades são extremamente específicas e sua real necessidade é questionável.

Falta de relação entre habilidade e objeto de conhecimento: algumas habilidades não tem uma relação evidente com o objeto de conhecimento ao qual estão relacionadas.

Pensamento computacional não é desenvolvido: Não há nenhum objeto de conhecimento ou habilidade que trabalhe os princípios do PC. Essa habilidade deve ser construída ao longo dos anos e de maneira sistemática e incremental.

Mundo Digital não é apresentado: Na BNCC, todos os conhecimentos sobre o Mundo Digital foram ignorados, a despeito de ser sempre ressaltada a importância da BNCC na formação do cidadão completamente inserido no Mundo Digital. Portanto, os conteúdos de habilidades relacionados à **Computação** não são de fato desenvolvidos na versão homologada da BNCC-EF.

Após estabelecer essas considerações, a SBC apresenta seu entendimento para PC, que é:

**Pensamento computacional:** Habilidade de compreender, definir, modelar, comparar, solucionar, automatizar e analisar problemas (e soluções) de forma metódica e sistemática.

O pensamento computacional não tem como objetivo traduzir uma situação dada em outra linguagem, ou transformar situações problema em tabelas e gráficos. Pensamento computacional é uma habilidade relacionada à construção de soluções para problemas, envolvendo a descrição e generalização dos

*processos de solução, bem como sua automatização e análise. Utilizam-se sim linguagens para descrever as soluções, porém a ênfase é no processo de construção da solução em si.*

Também questiona a abordagem apresentada na BNCC para algoritmos:

*Algoritmos podem ser representados por fluxogramas, porém, como discutido anteriormente, esta não é a representação mais adequada. Existe uma gama de outras linguagens visuais que podem ser empregadas para este fim, e que possuem características muito mais desejáveis do ponto de vista didático-pedagógico. A analogia entre Álgebra e Algoritmo é bastante questionável. A Álgebra é uma área da Matemática que estuda manipulações simbólicas, permitindo que se descrevam relações entre grandezas de forma genérica, através do uso de variáveis, termos e equações. O conceito de variável na Álgebra é usado para permitir a expressão sintática de relações sem a necessidade de listar instâncias concretas, ou seja, uma variável é um nome que usamos para referenciar um valor qualquer. Em Computação, o conceito de variável é diverso, tanto pode ser similar ao conceito algébrico (que é o caso, por exemplo, em paradigmas funcionais), quanto pode representar um lugar ou posição de memória onde um valor é guardado (em paradigmas imperativos). O fato de usarmos variáveis tanto na construção de Algoritmos quanto na Álgebra não torna estas duas áreas similares. Os objetivos são completamente distintos.*

Tais colocações são feitas com base no texto da BNCC (pag. 269) sobre Introdução à Matemática.

Ainda, no mesmo documento a SBC comenta sobre as habilidades da Matemática apresentadas na BNCC, 6º ano onde se pode ler "Objeto de conhecimento: Fluxograma para determinar a paridade de um número natural (EF06MA04) Construir algoritmo em linguagem natural e representá-lo por fluxograma que indique a resolução de um problema simples (por exemplo, se um número natural qualquer é par)." A SBC questiona esta colocação através de vários pontos:

*Por que um objeto de conhecimento tão específico "fluxograma para determinar a paridade de um número natural"? Algoritmos para outras questões matemáticas não são necessários? Por que um fluxograma? Não deveria ser solicitada uma linguagem específica na base. Seria melhor o termo "linguagem visual para descrição de processos" (fluxograma é uma delas). Fluxograma segue o paradigma de linguagens não-estruturadas, que sabidamente não é adequado para exercitar as boas práticas de resolução de problemas.*

Ainda considerando o 6º ano, mas agora para o que a BNCC coloca para Geometria: "Objeto de conhecimento: construção de retas paralelas e perpendiculares, fazendo uso de réguas, esquadros e softwares (EF06MA23) Construir algoritmos para resolver situações passo a passo (como na construção de dobraduras ou na indicação de deslocamento de um objeto no plano segundo pontos de referencia e distância fornecidas etc.)". A SBC comenta que essa habilidade não está relacionada com o objeto de conhecimento.



No mesmo sentido, ainda para o 6º ano, o documento questiona a proposta que conecta o PC e Probabilidade e Estatística, especificamente sobre o texto da BNCC “Objeto de conhecimento: Diferentes tipos de representações: gráficos e fluxogramas (IEF06MA34) Interpretar e desenvolver fluxogramas simples, identificando as relações entre os objetivos representados (por exemplo, posição de cidades considerando as estradas que as unem, hierarquia dos funcionários de uma empresa, etc.).” Quanto a essa colocação da BNCC, a SBC contrapõe que,

*um fluxograma não serve para representar informação em si, mas processos. Os exemplos dados poderiam ser representados por organogramas, grafos, ou seja, abstrações para representar dados/informações, mas não por fluxogramas.*

Para o proposto na BNCC, para o 7º ano, a SBC faz observações sobre “Objeto de conhecimento: Fração e seus significados: como parte de inteiros, resultado da divisão, razão e operador (EF07MA07) - Representar por meio de um fluxograma os passos utilizados para resolver um grupo de problemas.” Coloca a SBC:

*“Por que fluxograma? Mesmos comentários de itens anteriores. Esta habilidade não está relacionada com o objeto de conhecimento.”*

No que se refere ao uso dos fluxogramas, as observações são as mesmas já feitas para o 6º ano.

Quanto a proposta para Álgebra, “Objeto de conhecimento: Linguagem algébrica: variável e incógnita. (EF07MA14) Classificar sequências em recursivas e não recursivas, reconhecendo que o conceito de recursão está presente não apenas na matemática, mas também nas artes e na literatura”. A SBC observa que:

*o conceito de variável e incógnita não está relacionado à habilidade de classificar sequências recursivas e não recursivas.*

Por último, ainda para o 7º ano, para o conceito de Geometria, “Objeto de conhecimento: Triângulos: construção, condição de existência e soma das medidas dos ângulos internos (EF07MA26) - Descrever, por escrito e por meio de um fluxograma, um algoritmo para a construção de um triângulo qualquer, conhecidas as medidas dos três lados”. E, também para o “Objeto de conhecimento: Polígonos regulares: quadrado e triângulo equilátero (EF07MA28) - Descrever, por escrito e por meio de um fluxograma, um algoritmo para a construção de um polígono regular (como quadrado e triângulo equilátero), conhecida a medida de seu lado.” A SBC comenta que:

*novamente, são habilidades extremamente específicas, que envolvem conseguir construir um algoritmo concreto e usar para descrevê-lo uma linguagem específica. Habilidades deveriam ser genéricas. Além disso, fluxogramas não deveriam ser requeridos.*

Para o 8º ano, no que se refere à Álgebra, “Objeto de conhecimento: sequências recursivas e não recursivas (EF08MA10) Identificar a regularidade de uma sequência numérica ou figural não recursiva e construir um algoritmo por meio de um fluxograma que permita indicar os números ou as figuras seguintes.

(EF08MA11) Identificar a regularidade de uma sequência numérica recursiva e construir um algoritmo por meio de um fluxograma que permita indicar os números seguintes.” Quanto a essas sugestões da BNCC, as observações apontadas pela SBC são:

“uma sequência recursiva é uma sequência que tem uma lei de formação. Consequentemente, não é possível construir um algoritmo para determinar o próximo número de uma sequência não recursiva, como sugere a habilidade EF08MA10, (pois ela não tem lei de formação, só é possível construir algoritmos para processos que tem lei de formação” e, também, é novamente questionada a indicação do uso de um fluxograma (EF08MA11).

Com relação à Geometria, ainda no 8º ano, “Objeto de conhecimento: construções geométricas: ângulos de 90°, 60°, 45° e 30° e polígonos regulares (EF08MA16) - Descrever, por escrito e por meio de um fluxograma, um algoritmo para a construção de um hexágono regular de qualquer área, a partir da medida do ângulo central e da utilização de esquadros e compasso.” A esse respeito, a SBC questiona:

*por que este algoritmo concreto? Por que fluxograma?*

Para o 9º ano também são feitas observações sobre as propostas da BNCC com relação à Geometria, no que se refere ao “Objeto de conhecimento: Polígonos regulares (EF09MA15) - Descrever, por escrito e por meio de um fluxograma, um algoritmo para a construção de um polígono regular cuja medida do lado é conhecida, utilizando régua e compasso, como também softwares.” As observações são as mesmas que no caso do proposto para o 8º ano, ou seja:

*por que esse algoritmo concreto? Por que fluxograma?*

Já para as habilidades para a Computação, apresentadas na BNCC para o ensino médio, a SBC apresenta as seguintes observações:

- “(EM13MAT315) Reconhecer um problema algorítmico, enunciá-lo, procurar uma solução e expressá-la por meio de um algoritmo, com o respectivo fluxograma”:

*Um problema não é algorítmico, a solução pode ser algorítmica.*

*Fluxogramas aparecem novamente. Como já exposto nas críticas à BNCC-EF, fluxogramas são representações arcaicas e inadequadas de algoritmos, que não estimulam (e atrapalham) a utilização das principais técnicas de solução de problemas.*

*Reconhecer se um problema tem ou não solução algorítmica é difícil. Para se provar que não existe possibilidade de resolver um problema de forma algorítmica, precisa-se primeiro definir o que é “forma algorítmica” e como se representa “um problema”. Ou seja, está-se falando de um nível mais alto de raciocínio, no qual se fala sobre algoritmos. Esse nível é próximo da Filosofia. Aqui se depara frequentemente com paradoxos (por exemplo, o que acontece se*

*um algoritmo recebe um outro algoritmo como entrada?). A SBC acredita que esse tópico é sim relevante na formação do estudante, pois estimula o desenvolvimento da análise crítica, em especial considerando questões próximas da Filosofia e questionando a própria Computação em si (ou seja, questionando os limites da Computação). Porém não é possível abordá-lo de uma forma ingênua, é necessário que a fundamentação teórica seja apresentada ao aluno para que a análise seja passível de ser trabalhada.”*

- “(EM13MAT406) Utilizar os conceitos básicos de uma linguagem de programação na implementação de algoritmos escritos em linguagem corrente e/ou matemática”:

*A redação desta habilidade remete a uma ideia inadequada sobre o real objetivo de se ensinar Computação. O objetivo não é utilizar uma linguagem de programação para representar algoritmos, e sim criar algoritmos para resolver problemas, e ser capaz de representar esses algoritmos de diversas formas (língua portuguesa, linguagens visuais, linguagens de programação, etc.).*

*Para atingir essa habilidade, são necessários diversos conceitos referentes à representação abstrata de informações e processos que não foram trabalhados nas propostas da BNCC para o ensino fundamental (e nem para o ensino médio). A experiência no ensino de algoritmos por décadas mostra que essa maneira (foco na linguagem) é inadequada para desenvolver a habilidade de criar algoritmos, e normalmente fadada ao fracasso.*

## 9.6 Integrar o PC ao Ensino Básico

Na prática, existem vários estudos sobre a integração do PC na educação básica (fundamental e médio para a abrangência deste estudo), seja ele *unplugged* ou *plugged*, em diferentes países, com diferentes realidades socioeconômicas. O objetivo deste item é apresentar algumas dessas propostas.

Neste sentido cabe apontar a pesquisa de Valente (2016), que realiza um levantamento entre diferentes autores e define seis categorias de abordagens no ensino do PC na Educação Fundamental, a saber:

1. *Atividades sem o uso das tecnologias*: utilizar abordagens lúdicas, truques de magia e competições para mostrar às crianças o tipo de pensamento que é esperado por um cientista da Computação.
2. *Programação em Scratch*: linguagem de programação baseada em blocos visuais, projetados para facilitar a manipulação da mídia por programadores novatos.
3. *Robótica pedagógica*: utilização de aspectos/abordagens da robótica industrial em um contexto no qual as atividades de construção, automação e controle de dispositivos robóticos propiciam a aplicação concreta dos conceitos em um ambiente de ensino e de aprendizagem. (Nessa área o

exemplo de escolas piloto em Portugal é bastante relevante. Nota dos autores).

4. *Produção de narrativas digitais*: consiste no uso das TICs na produção de narrativas que tradicionalmente são orais ou impressas, também conhecidas como histórias digitais, narrativas interativas ou *digital storytelling*<sup>43</sup>.
5. *Criação de jogos*: desenvolvimento de sistemas compostos pela estética do visual e som, narrativa contando a história do jogo, mecânica de regras do jogo e a tecnologia usada para produzir um jogo eletrônico. Nas plataformas voltadas para o PC *plugged* existem algumas específicas para o desenvolvimento de jogos.
6. *Uso de simulações*: uso de *softwares* que criam um ambiente para observar fenômenos que sejam difíceis, ou até mesmo não passíveis de serem desenvolvidos no mundo real.

Cada uma das abordagens possui uma característica diferente, para se chegar ao objetivo comum: o ensino do PC. Note-se que todas as abordagens necessitam do uso de equipamentos e *softwares* específicos, exceto a primeira.

Os estudos científicos que são apresentados a seguir possuem abrangência ampla, ou seja, tratam da educação básica, posto que envolvem a educação fundamental e a média.

De acordo com estudos realizados por Barcelos e Silveira (2012), uma estratégia para inserção do PC no ensino básico pode ocorrer através de disciplinas pré-existentes no atual currículo, como, por exemplo, a Matemática, incentivando seu uso também em outras disciplinas, para poder atingir um público cada vez maior. CSTA/ISTE (2009) também realizaram um levantamento com diversas propostas de adoção de forma inter, multi e transdisciplinar.

Outros exemplos de integração do PC na Educação Fundamental são descritos em livros, como Bunce (2015), Burrett (2016), Patterson (2016) e Bird *et al* (2017), através de lições que utilizam computadores; Caldwell e Smith (2017) e Brackmann (2017), sem a necessidade de usar qualquer equipamento eletrônico. O trabalho de Brackmann foi aplicado em escolas brasileiras, em um estudo piloto, para sua tese de doutorado no PGIE/UFRGS (Pós-Graduação em Informática na Educação/UFRGS).

Entretanto, mais importante que esses exemplos é o processo na aplicação do PC, independentemente da disciplina, ou seja, o ponto de partida de qualquer atividade:

- É a exploração de problemas, propondo situações em que os alunos desenvolvam algum tipo de estratégia para resolvê-las;

---

<sup>43</sup> A arte de contar histórias, utilizando ferramentas tecnológicas, como *tablets*, *desktops*, câmeras digitais e *smartphones*: essa é a definição mais simples para o [digital storytelling](#), ou a contação digital de histórias.

- O enunciado do problema exerce função importante, pois esse é fornecido ao aluno para que ele interprete e estruture a situação que lhe é apresentada para ser resolvida;
- O método de busca da solução envolve a realização de aproximações sucessivas rumo à resolução do problema, utilizando o que o aluno já aprendeu para a resolução de outros problemas;
- O aluno constrói um conceito para a definição de um problema, mas também, constrói conceitos articulados com outros que façam sentido num conjunto de problemas similares (que compartilham alguma etapa da sua solução).

Esse conjunto de orientações para aplicar o PC em situações de ensino/aprendizagem do currículo escolar pode servir de guia para que os professores busquem utilizar o PC nas suas disciplinas.

No caso específico do Brasil, destaca-se o trabalho possivelmente pioneiro de **Fernandes (2001)**, **Fernandes & Menezes (2002)** e **Bavaresco et al (2018)**, intitulado “Ciência da Computação para Crianças” (anterior à consolidação do termo Pensamento Computacional) em que é apresentada uma proposta metodológica para o ensino de Ciência da Computação no ensino básico, elaborada a partir de pesquisas e estudos e contemplando a aplicação da mesma.

Também, houve preocupação em trabalhar de forma interdisciplinar a demonstração e ensino dos conceitos fundamentais da Ciência da Computação. Por exemplo, algoritmos são utilizados para desenvolvimento de cálculos, noções de lógica e conjuntos, história da Filosofia, conjugação de verbos etc. Outra preocupação foi transformar esses conceitos em aplicações em que fossem utilizados elementos os mais simples possíveis e presentes no cotidiano. A transformação desses conceitos em aplicações úteis mostrou-se de fundamental importância no desenvolvimento do trabalho.

Conforme Fernandes, uma importante aplicação do trabalho foi no comparativo de crianças do Colégio de Aplicação da UFRGS com universitários do Curso de Bacharelado em Ciência da Computação, também da UFRGS, sendo que:

- no Colégio de Aplicação, o ingresso é por sorteio, representando todas as camadas sociais e culturais; a amostragem foi de 12 crianças com idades variando de 10 a 12 anos; ao todo foram 10 encontros, totalizando cerca de 15 horas de desenvolvimento;
- no Bacharelado em Ciência da Computação, o ingresso é via um dos mais disputados e seletivos vestibulares e esse curso é considerado um dos melhores cursos de Ciência da Computação do Brasil; a amostragem foi de 40 alunos entre quarto e quinto semestre, representando cerca de 1500 horas de curso.

Como resultado geral verificou-se que:

*o desenvolvimento das crianças tende a ser mais rápido que o dos universitários.*

De fato, crianças selecionadas por sorteio e com um centésimo da carga horária atingiram níveis próximos dos alunos do Bacharelado. Especificamente, para problemas de lógica:

*Cerca de 70% das crianças resolviam os problemas muito rapidamente;*

*Cerca de 50% das crianças eram mais rápidas do que 25% dos alunos do Bacharelado, para os mesmos problemas.*

O Brasil é um país grande e com muitas diferenças regionais. Talvez o projeto para o ensino da Computação possa ser único, mas sua implantação necessita observar essas diferenças, como existência ou não de energia elétrica, existência ou não de conexão, existência ou não de equipamentos e principalmente a disponibilidade de professores capacitados para trabalhar esses conceitos com ou sem o uso de tecnologia. Implementações diferenciadas às vezes são salutares, como se pode ver nas experiências de vários países, que estão brevemente apresentadas neste documento.



## 10. Formação de Professores

A formação do professor é um dos fatores predominantes para a adoção adequada do PC, seja *unplugged* ou *plugged*. De forma ampla, a integração das TICs no currículo e na formação inicial dos professores é um desafio, pois durante o processo de tornar o PC ou outra metodologia voltada para a inclusão digital, como uma competência para o século XXI, os professores necessitam fluência nessas metodologias e nas tecnologias que possam estar vinculadas a essas propostas. Isso envolve refletir sobre a diversidade de conceitos, teorias, modelos e práticas que podem ser aplicados nas disciplinas contempladas nos currículos dos diferentes países, que optem por adotá-las.

Esta discussão surge dado o movimento popular e o ímpeto dos governos para expandir as oportunidades de aprendizado da Ciência da Computação nas salas de aula do ensino fundamental e médio, incluindo a iniciativa *Computer Science For All*, lançada em janeiro de 2016 nos EUA:

<https://obamawhitehouse.archives.gov/blog/2016/01/30/computer-science-all> (acesso em 18/12/2018)

Existem várias indicações de como e de que quais tecnologias são as mais adequadas para cada região do mundo. A Unesco apresenta uma política ampla igualitária, focada na utilização de tecnologias básicas, como as de comunicação, por exemplo, uso do e-mail, e em tecnologias voltadas para a formação profissional, como o uso de editores de texto, planilhas eletrônicas, etc. Por outro lado, a Fundação *Horizon* (que atua na UE) aponta o ensino do uso de tecnologias específicas para cada região, sendo que essa recomendação pode envolver tecnologias de ponta, como as impressoras 3D, sistemas de IA altamente personalizados, Realidade Virtual e Realidade Aumentada. Certamente, para cada caso, a formação dos professores será diferente.

De qualquer forma, o foco deste texto está na formação dos professores para a utilização do PC em suas disciplinas, de maneira geral e não especificamente na disciplina de Computação. Como é possível observar, existem poucos artigos (apenas dois autores, totalizando 5 (cinco) artigos ou textos) que tratam da capacitação de professores. Sendo que desses, 3 (três) são de um mesmo autor, Curzon, do Reino Unido. Ainda, entre esses artigos encontrados, quando se trata da avaliação do processo de formação dos professores, os mesmos utilizaram uma avaliação qualitativa, baseada na opinião dos participantes. Não foram encontrados textos que abordassem o resultado prático que essas iniciativas provocaram nos professores, em sala de aula.

A seleção de artigos buscou focar em textos que apresentam a formação de professores tendo como base a metodologia do PC. Mas na maioria dos casos a formação foca em apenas um pequeno aspecto do PC, como por exemplo, a utilização de um jogo e, a partir dele, que conceitos da Ciência da Computação podem ser abordados com os alunos.

O tema tem sido tratado de forma individual pelos autores de livros, ou pelas associações vinculadas com a área. Ou seja, cada autor desenvolve conteúdos, geralmente em forma do MOOC (contendo vídeos, textos e exemplos), voltados para a capacitação dos professores, no uso particular de seu livro ou da proposta apoiada por uma determinada fundação/associação. O item 15 - Anexos traz exemplos desses casos.

## 10.1 Foco na Formação para o PC *Unplugged*

O foco deste trabalho está no uso da abordagem do PC *unplugged*. Logo, para capacitação de professores nessa metodologia. Cabe ressaltar que existem muitos sítios de livros que oferecem a capacitação para os professores que desejam utilizá-los em sala de aula. Essa formação é oferecida através de um MOOC e os professores que a realizam não são avaliados. Ainda, é importante lembrar que, para realizarem suas capacitações através de MOOC, os professores necessitam possuir equipamentos disponíveis, bem como conhecimentos mínimos de Informática. Dentre os poucos textos que foram encontrados na literatura sobre o tema, pode-se destacar o de [Curzon et al \(2014\)](#). No experimento de Curzon, a formação aconteceu através de uma série de oficinas, organizadas com o objetivo de explorar a eficácia dos métodos *unplugged*, para familiarizar os professores com conceitos da Computação.

Após as oficinas, foi realizada uma avaliação, baseada em formulários com respostas livres, que podem ser resumidas nas seguintes respostas:

*“as atividades unplugged proporcionam uma sessão inspiradora e divertida para os professores; eles consideraram esta metodologia útil e interessante para ser utilizada em sala de aula; a atividade contribuiu para melhorar sua confiança para uma possível aplicação da metodologia”.*

Na mesma linha, em trabalho prévio, realizado no Reino Unido, Curzon (2013) descreve como as atividades *unplugged* incorporadas em *histórias mágicas* podem ser usadas para capacitar os professores no uso do PC. Após uma oficina de desenvolvimento profissional de 2 (duas) horas para professores, 100% dos participantes afirmaram que o evento lhes ofereceu ideias úteis e viáveis para a sala de aula. Curzon também fornece um manual básico de orientação para a utilização do PC.

Um outro experimento similar foi relatado por [Faber et al \(2017\)](#), onde, após um total de 14.040 (quatorze mil e quarenta) horas-aula de ensino de PC em 26 (vinte e seis) escolas na Espanha, a conclusão foi que a abordagem *unplugged* (um dos poucos trabalhos que trata da avaliação da capacitação dos professores na abordagem *unplugged*) é uma alternativa em relação à programação em computadores. Essa conclusão carece de maiores detalhes que a comprovem na prática do dia a dia dessas escolas. Os autores também conduziram experimentos no Brasil.

Como pode ser visto, os relatos anteriores buscam observar o entusiasmo e o interesse dos participantes pela Computação, mas não avaliam o quanto os participantes compreenderam sobre os conceitos abordados e, mesmo, sobre o uso da metodologia. Esta carência de processo de avaliação existe ainda, tanto para a capacitação dos professores, como para aprendizagem dos alunos. Dentre os conceitos de PC, na maioria dos casos, o foco dos experimentos está na capacitação para a programação e essa habilidade é abordada somente durante a formação dos professores e não na sua prática escolar.

Com a revisão de literatura pode-se constatar que existe a necessidade de mais pesquisas que sejam cientificamente válidas, que forneçam maiores evidências sobre a eficácia das atividades *unplugged* no desenvolvimento das habilidades vinculadas ao PC, especialmente quando se trata das escolas dos anos iniciais de formação, onde essa forma do PC é mais utilizada. Essa carência de avaliação acontece tanto para o caso da formação dos professores, como para a eficácia do método para formação dos alunos em conceitos de Computação.

Quando se trata da formação/capacitação dos professores para o uso do PC em suas disciplinas, sejam elas de Ciências (Física, Química e Biologia), Matemática, Informática ou Ciência da Computação, é preciso apresentar o que é o PC e talvez esse seja o único caso em que as aulas serão sobre o PC, propriamente dito.

Pode-se concluir que a formação dos professores, na maioria dos casos, é realizada através do uso de um MOOC. Isso é verdade também, para a formação específica, como, por exemplo, a formação de como o professor pode utilizar um determinado livro em suas aulas.

Como já colocado, grande parte dos aplicativos, jogos e livros voltados para o PC possuem tutoriais, com passo a passo, galerias de exemplos, etc. Entretanto, a maioria desses materiais está disponíveis apenas em língua inglesa. Ainda, esses tutoriais não estão acompanhados de um processo de avaliação. Em alguns casos, isso se resume à informação, opcional, que o professor aluno fornece, em termos de 'respostas abertas' sobre a sua capacitação.

## 10.2 Proposta para a Formação de Professores

No Brasil existe a Licenciatura em Computação (79 (setenta e nove) cursos, tanto presenciais como a distância) que forma professores capacitados para desenvolver atividades de ensino relacionadas com PC, e que não são adequadamente aproveitados. Igualmente cabe destacar que mesmo podendo-se contar com esses docentes, é necessária a formação de professores, para que utilizem adequadamente a proposta do PC, em sala de aula. As 79 Licenciaturas estão distribuídas por região como segue (ver também a Figura 10.1), conforme dados do endereço <http://emec.mec.gov.br> (acesso em 4/12/2018):

Sul	11
Sudeste	19
Centro-Oeste	11
Nordeste	27
Norte	11
<i>TOTAL</i>	<i>79</i>

### LICENCIATURAS EM COMPUTAÇÃO POR REGIÃO

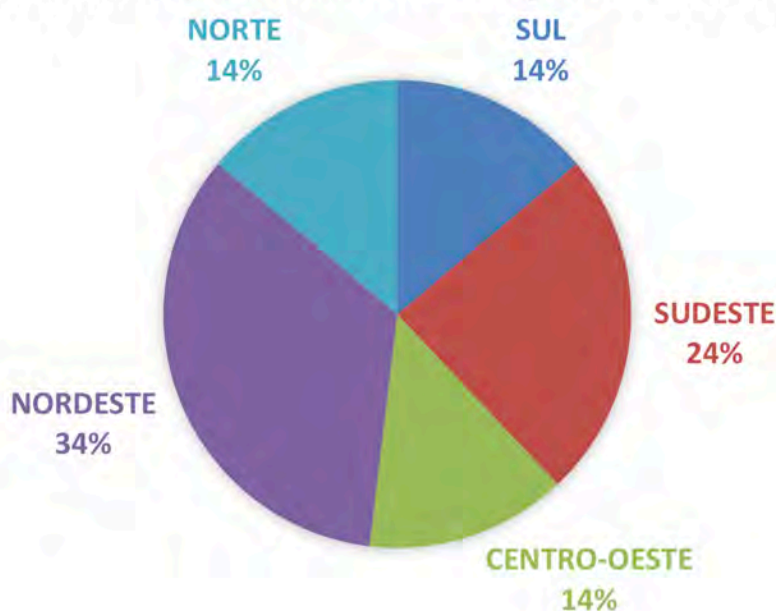


Figura 10.1: Localização dos cursos de licenciatura no Brasil

E, também, é importante o desenvolvimento de materiais e abordagens sobre o emprego do PC, para suporte a esses professores. Sem uma capacitação eficiente, os materiais não serão utilizados de forma adequada em sala de aula. Na opinião dos autores, a capacitação para o uso do PC *unplugged* é principalmente pedagógica, mas uma introdução aos conceitos de Computação também se faz necessária. Já para o uso do PC *plugged*, pelo fato de envolver *software*, que nem sempre possui documentação em português, a formação técnica, além da pedagógica, é relevante.

Apresentamos neste capítulo ideias e sugestões para a capacitação de professores nos cursos de licenciatura, em geral, e na Licenciatura em Computação, em particular, dado o fato de que o Brasil possui esse tipo de curso.

Na literatura encontrada não é feita grande diferença entre a forma de capacitar professores durante sua formação ou professores em serviço. No entanto, a grande maioria dos autores defende que a melhor estratégia é a capacitação durante os cursos de Licenciatura. Nesse contexto, é fundamental que os professores em formação compreendam o PC no contexto da área do conhecimento que estarão estudando, para posteriormente ensinar. Isso requer que eles tenham uma compreensão profunda de sua própria disciplina e conhecimento de como os conceitos do PC se relacionam com o que os alunos irão aprender em sala de aula. Portanto, é importante que os professores formadores possibilitem que os seus alunos, futuros professores, “construam esses conhecimentos para se sentirem à vontade para aplicá-los em suas futuras aulas”.

Um estudo feito por Yadav (Yadav *et al* 2014) examinou a influência de um módulo de PC, de uma semana, sobre a compreensão e as atitudes dos professores em formação em relação à incorporação do pensamento computacional em suas futuras salas de aula. Professores em formação matriculados em um curso introdutório de Psicologia Educacional foram divididos em dois grupos. Um grupo (o grupo de controle) não experimentou o módulo de PC, enquanto o segundo grupo (o grupo experimental) aprendeu especificamente sobre as ideias de PC através do módulo. Os autores descobriram que a maioria dos professores em formação no grupo de controle via o PC como integração de tecnologia na sala de aula, enquanto a maioria dos participantes do grupo experimental desenvolveu sua compreensão do PC como uma abordagem de solução de problemas, usando algoritmos e heurísticas. Os resultados também sugeriram que os professores em formação do grupo experimental eram mais capazes de articular como integrar o PC nas salas de aula do ensino fundamental e médio, em comparação com o grupo de controle. Os resultados do estudo indicam o potencial para integrar o PC para os professores em formação nos cursos existentes de Licenciatura.

Os criadores desse experimento usaram exemplos da vida cotidiana, bem como exemplos específicos da disciplina de cada futuro professor, para destacar o PC. Por exemplo, utilizaram um exemplo de informar direções do ponto A ao ponto B, para destacar o que é um algoritmo (uma rota passo a passo), a eficiência dos algoritmos (como fornecer a melhor maneira de chegar de A a B), a abstração (como efetivamente informar qualquer direção) e automação (como projetar um sistema similar ao *Google Maps*).

Em outro exemplo, eles (Yadav *et al* 2012) mostraram a ideia de processamento paralelo através do exemplo sobre a maneira mais rápida para dois amigos comprarem ingressos de cinema quando três linhas de acesso à bilheteria estiverem disponíveis. Estes módulos de PC estão disponíveis em [http://cs4edu.cs.purdue.edu/comp\\_think](http://cs4edu.cs.purdue.edu/comp_think) (acesso em 6/12/2018). Nele também é possível encontrar outros exemplos de como as construções do PC podem ser apresentadas para os professores. Diferentemente do experimento anterior, o estudo de (Yadav *et al* 2013) foi conduzido em um curso geral de formação de professores, para todas as áreas do conhecimento.

Mas como sabemos, existem muitos professores já formados, em atividade e que talvez necessitem serem familiarizados com as ideias e competências do PC, através de cursos preexistentes, como os de formação continuada em *tecnologias educacionais* (título genérico utilizado internacionalmente para disciplinas com esse objetivo). Esses cursos são, normalmente, desconectados das teorias e métodos de ensino que os professores em formação nas licenciaturas aprendem, concentrando-se em tecnologia (como ferramentas da Web 2.0, para ensinar). Nesses casos a proposta pode ser a de revisar o conteúdo desses cursos preexistentes para fornecer aos professores oportunidades de pensar computacionalmente e experimentar o PC como um conjunto genérico de habilidades e competências que não dependem, necessariamente, de computadores ou de outras tecnologias educacionais. Redesenhar cursos de introdução à tecnologia educacional em torno do aprendizado de conceitos e capacidades do PC é também uma oportunidade para as áreas de Ciência da Computação e Educação trabalharem em conjunto.

Para Taylor (1980), muitas das primeiras iniciativas de introduzir a Computação no ensino fundamental e médio focam em programação simples, destinada a ajudar os alunos a “aprender algo sobre como os computadores funcionam e como funciona o seu próprio pensamento.” Cerca de uma década atrás, no entanto, os cursos de tecnologia educacional se distanciaram dessa visão e começaram a se concentrar no uso de ferramentas de *software* pré-projetadas na sala de aula pelos próprios alunos. Ou seja, os cursos poderiam ser personalizados, para grupos de professores, com base em suas áreas de conhecimento e vinculados ao seu dia-a-dia na sala de aula.

Em linha com essa proposta, a estrutura do Conhecimento Tecnológico Pedagógico Tecnológico (TPACK) é um modelo útil para integrar o PC, no qual as ideias relacionadas são estreitamente interligadas ao assunto e as abordagens pedagógicas que os professores em formação ensinarão em suas futuras salas de aula, ou que professores já em serviço poderão adotar no seu dia-a-dia. A TPACK amplia a ideia de Shulman (1986) do conhecimento pedagógico de conteúdo, ao incluir o conhecimento que os professores precisam para ensinar, de forma eficaz, usando a tecnologia.

Segundo a TPACK, os professores formados de acordo com a essa proposta aprendem sobre a integração efetiva da tecnologia no contexto de sua área de conhecimento e da pedagogia. Ainda, segundo a TPACK (<http://periodicos.uem.br/ojs/index.php/ImagensEduc/article/view/34615> - acesso em 4/12/2018), cursos de métodos científicos, em programas de formação de professores, também oferecem uma oportunidade para ajudar os professores em formação a incorporar o PC no contexto de suas áreas temáticas. Os cursos de Metodologia Científica permitem que eles adquiram novas maneiras de pensar sobre ensino e aprendizagem em uma área específica e oferecem oportunidades para “desenvolver maneiras pedagógicas de fazer, agir e ser como professor”. Um curso de métodos tece “conhecimento conjunto sobre: a disciplina de cada professor, o conhecimento sobre as crianças e como elas aprendem, sobre o papel



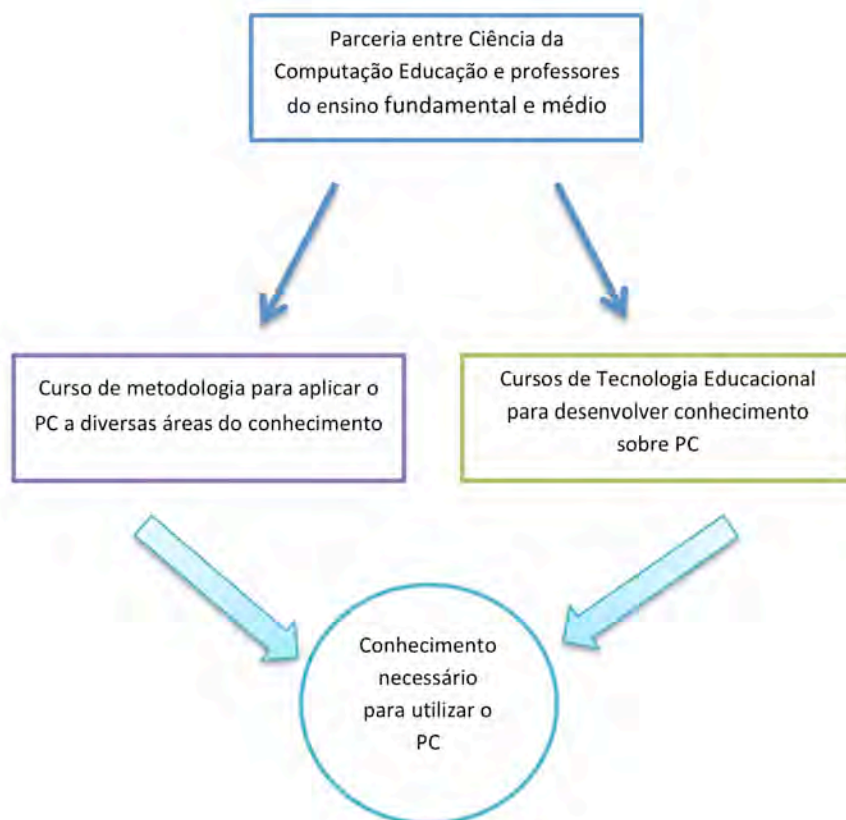
do professor e sobre a vida em sala de aula e seu papel na aprendizagem do aluno.” Nesse contexto, um curso de métodos também pode ser um lugar onde os professores exploram ideias de PC no contexto de sua disciplina específica (especializações de áreas). Por exemplo, em um curso de Métodos de Ensino de Inglês, futuros professores e professores já em sala de aula poderiam aprender a incorporar algoritmos, em uma atividade de escrita, pedindo aos alunos que escrevam uma receita detalhada - uma série de instruções passo a passo - para uma comida favorita. Da mesma forma, professores em um curso de métodos de estudos sociais poderiam aprender a incorporar análise de dados e reconhecimento de padrões, fazendo com que os alunos coletassem e analisassem estatísticas populacionais e usassem esses resultados para identificar e representar tendências. As ferramentas de análise de dados poderiam ser tão simples quanto o *Piktochart*, que permite que os alunos representem dados e informações por meio de infográficos, ou ferramentas mais avançadas, como o *Google Charts*, que permite aos alunos representar dinamicamente os dados, usando gráficos personalizáveis e interativos. Professores em um curso de Ciência poderiam ser introduzidos ao raciocínio computacional através de modelos computacionais, para demonstrar ideias científicas e fenômenos para seus alunos. Os modelos computacionais também poderiam ser usados para testar hipóteses, bem como soluções para problemas. Por sua vez, os alunos poderiam usar ferramentas, como o *NetLogo* e o *Scalable Game Design*, para desenvolverem modelos computacionais, à medida que se envolvessem na realização de simulações e *design* de jogos, por exemplo.

Os conceitos gerais de PC adquiridos no curso de tecnologia educacional e as práticas de PC específicas da disciplina, adquiridas nos cursos de métodos, ajudariam os professores em formação a conectar o PC ao conteúdo que ensinarão em suas salas de aula. Dessa forma, os construtos do conhecimento pedagógico da disciplina específica e do conhecimento pedagógico do conteúdo tecnológico forneceriam suporte para o desenvolvimento do conhecimento do PC dos professores.

Especificamente, os cursos de tecnologia educacional serviriam de base para o desenvolvimento de conhecimento de conteúdo para o PC. Esse conhecimento permitiria que os professores explorassem ideias centrais do PC, por que essas ideias são centrais e como os construtos do PC são semelhantes ou diferem de outros conceitos paralelos (como o pensamento matemático). Com cursos de métodos de ensino, eles aprenderiam a integrar o PC no contexto de áreas específicas. Isso permitiria que eles aprendessem como representar e formular o PC no tema de sua disciplina, em particular, e torná-lo compreensível para os alunos. Envolvendo a formação de professores em ideias de PC no contexto do ensino de suas áreas de conhecimento, os formadores de professores poderiam assegurar que o PC tornar-se-ia parte do seu próprio conjunto de ferramentas de vocabulário e resolução de problemas.

A Figura 10.2 fornece uma estrutura inicial em torno da qual os formadores de professores poderiam começar a moldar as experiências iniciais dos professores

em formação, nos programas de formação de professores do ensino fundamental e médio.



*Figura 10.2: Estrutura para abordar o PC na formação de professores*

Fonte: [Yadav\\_Stephenson\\_CACM, 2017](#).

No ensino fundamental, a incorporação de exercícios de PC à alfabetização oferece uma transição direta para os candidatos a professores. Por exemplo, os professores poderiam explorar como incluir a abstração na análise de temas dentro de prosa ou poemas, usando detalhes textuais ou resumindo o texto. Eles também poderiam construir um plano de aula, para seus alunos, que incorporasse a análise de dados e a representação de dados, identificando palavras que retratam sentimentos e comparando como elas são representadas em diferentes versões da mesma história.

Da mesma forma, os professores em formação poderiam incorporar o PC em planos de aula para as aulas de Linguagem, no nível secundário, sugerindo que os alunos coletassem e integrassem dados/informações de várias fontes, para representar visualmente temas comuns (mapas conceituais construídos com a ferramenta do Google ou de forma manual). Professores de Ciência do nível médio poderiam incluir coleta, análise e representação de dados em qualquer atividade na qual os estudantes coletassem dados e identificassem e representassem padrões nesses dados. Por fim, os professores de estudos sociais poderiam explorar como

usar grandes conjuntos de dados (como dados do censo), para permitir que os alunos explorassem e identificassem padrões e discutissem as implicações do acesso crescente a grandes quantidades de dados pessoais.

Embora os cursos existentes de formação de professores ofereçam oportunidades para introduzi-los no PC, alguns programas podem considerar o desenvolvimento de cursos autônomos e/ou programas de certificação, que permitem que os professores do ensino fundamental e médio descubram o escopo dos conceitos e capacidades do PC, bem como o envolvimento com ferramentas computacionais que fomentam o desenvolvimento de competências do PC. Idealmente, esses cursos seriam desenvolvidos por docentes da Educação e da Ciência da Computação, colaborando para identificar os apropriados resultados de aprendizado e os recursos disponíveis.

Como integrar o PC em qualquer currículo envolve a exposição de professores e alunos a conceitos e práticas usados por Cientistas da Computação, é importante que os professores da área da Educação trabalhem em estreita colaboração com os de Ciência da Computação. Da mesma forma, o corpo docente tem uma compreensão diferenciada do currículo e políticas educacionais do ensino fundamental e médio que são fundamentais para garantir que os esforços atuais do PC sejam bem sucedidos. Nesse mesmo sentido, em um dos relatórios de 2016 da *Computing Research Association* (<https://cra.org/events/snowbird-2016/> - acesso em 18/12/2018) destacou a necessidade dos profissionais da Ciência da Computação estabelecer conexões interdisciplinares com colegas de outras disciplinas (como educação, psicologia educacional e licenciaturas em geral). Essas colaborações incluíram cursos de codesenvolvimento e coensino que preparam professores para ensinar o PC; veja *Yadav e Korb (2012)*. Além disso, estes profissionais podem colaborar em pesquisas sobre o “ensino” (sic) do PC Isso permitiria que cientistas da computação e educadores colaborassem no desenvolvimento de atividades conectadas e não conectadas. Os professores aprenderiam, então, como usar esse conhecimento para ensinar as crianças a pensar computacionalmente no contexto de uma área específica por meio de cursos de métodos.

Além disso, muitos recursos disponíveis poderiam ser incorporados em uma tecnologia educacional ou curso de metodologia científica que poderia ajudar os professores em formação a conectar o PC às suas vidas diárias e aos contextos de sala de aula. Por exemplo, os professores em formação podem realizar atividades do livro *CS Unplugged* (<http://csunplugged.org> acesso em 4/12/2018), muitas das quais ensinam habilidades de pensamento computacional sem precisar de um computador e são facilmente adaptadas a outros temas (um exemplo destas atividades é apresentado neste Capítulo). Os professores também poderiam usar o *Scratch* - um ambiente de programação que permite aos alunos criar programas arrastando e soltando blocos representando construções centrais - para criar programas e animações simples. O *Scratch* é brevemente apresentado no item PC *Plugged*.

Reconhecendo a necessidade de os professores lidarem com o PC em seus currículos e práticas, várias organizações, incluindo a CSTA, a ISTE

(<https://www.iste.org/> - acesso em 7/12/2018) e a *National Science Teachers Association*, também estão desenvolvendo e compartilhando ferramentas e recursos para professores atuais e futuros. O website *Exploring Computational Thinking* do Google (<http://g.co/exploringCT> - acesso em 4/12/2018) oferece mais de 130 planos de aula e exemplos; uma coleção de vídeos demonstrando como os conceitos de PC são usados na solução de problemas do mundo real; e um curso on-line “Pensamento Computacional para Educadores” (<http://g.co/computationalthinking> - acesso em 4/12/2018). Desde 2014, o Grupo de Pesquisa em Ciência da Computação da Universidade de Adélizide, na Austrália, tem feito parceria com o Google para criar cursos introdutórios para implementar o Currículo de Tecnologias Digitais da Austrália e ensinar Ciências da Computação e PC nos níveis fundamental e médio. Esses cursos estão, explicitamente ligados ao currículo australiano (<https://csdigitaltech.appspot.com> - acesso em 4/12/2018). Esses recursos proporcionam um ponto de partida para os educadores de professores, da Austrália, incorporarem ideias de PC e relacioná-las a uma área específica na qual os professores continuarão a ensinar em salas de aula.

Todos os documentos consultados referem, de alguma forma, que é mais fácil introduzir alunos em formação ainda nos cursos de Licenciatura, ao PC, do que professores já formados e em serviço. Por este motivo destacam a importância urgente de readequação das diferentes licenciaturas para abordarem o “pensar computacionalmente” e também para a utilização de tecnologias de forma natural. Ainda, mesmo as propostas para atividade *unplugged* se encontram disponíveis em sítios e as instruções para seu uso também.

## 11. Panorama Global da Adoção de Políticas para o Ensino de Computação

A seguir será apresentado:

- a) Um panorama da adoção de Políticas Públicas para o ensino de Computação, incluindo, em alguns casos, o PC *unplugged* e *plugged* na Educação Básica, em diferentes países. É importante salientar que nem sempre existe correspondência entre as faixas de idade, nos diferentes países, para essas nomenclaturas. Nem mesmo a nomenclatura é padronizada;
- b) Algumas instituições não governamentais com iniciativas ou sugestões para diferentes formas de incluir a Ciência da Computação e as tecnologias, em geral, no Ensino Fundamental. Cabe lembrar que estas instituições possuem alcance limitado, muitas vezes atendendo à interesses particulares e não substituem a atuação do Estado.

### 11.1 Países

Diferentes países adotam diferentes formas de introduzir a Computação no ensino. Em alguns casos, mais de uma forma é adotada, como se pode constatar nos exemplos apresentados a seguir.

A disciplina de Computação é diferente das aulas de Informática (que focam geralmente no uso de editores de texto, navegadores, planilhas eletrônicas, etc.) e o uso de habilidades da área da Ciência da Computação pode agregar benefícios educacionais (habilidades de reflexão e solução de problemas, compreensão dos limites e alcances da tecnologia digital) e econômicos (a necessidade de profissionais com boa formação). Embora Computação e Informática sejam conceitos diferentes, as políticas educacionais de alguns países confundem esses conceitos. Na nossa opinião, isso acontece pelo fato de o PC focar, em grande parte de suas implementações práticas, no desenvolvimento de habilidades para a programação.

Apesar de até o momento ter-se falado no PC *unplugged*, os programas dos países desenvolvidos deixam claro que sua implementação possui uma outra possibilidade, ou seja, a que utiliza também tecnologia (computadores e robôs) e, ainda, a opção por outras formas de introdução da Computação, como, por exemplo, a Inteligência Artificial, que traz a visão da Computação dos algoritmos evolutivos, através da aprendizagem de máquina e da programação declarativa, por exemplo. O ensino da Computação *plugged* é empregado prioritariamente por países desenvolvidos.

Aqui se apresenta, de forma resumida, resultados de uma revisão sistemática da literatura a respeito do panorama de diversos países que já adotaram ou estão em processo de adoção do ensino de Computação na educação formal. Em alguns casos, a definição da política de ensino de Computação não deixa claro se a

metodologia a ser utilizada será o PC, em outros casos, o PC resume-se ao ensino de programação e ainda em outros casos é utilizado o termo mais amplo, TICs. De qualquer forma, optou-se por trazer essas políticas para a discussão. Os países se encontram organizados em ordem alfabética e não pelo fato de adotarem explicitamente, ou não, o PC em suas políticas educacionais, para a formação em tecnologias.

Este levantamento foi motivado pelo trabalho liderado pela Microsoft (Jones et al, 2011), realizado em âmbito internacional, para averiguar a situação da introdução de disciplinas voltadas ao ensino e funcionamento de sistemas computacionais. E atualizado de acordo com relatórios da fundação Horizon e de consultoria prévia realizada por Rosa Vicari (uma das autoras) para a Confederação Nacional da Indústria, (Vicari 2017).

### 11.1.1 Alemanha

Cada um dos 16 Estados federativos da Alemanha possui um departamento de educação. Sendo que cada Estado assume um modelo diferente para a introdução da Computação nas escolas. Não existe uma padronização dos conteúdos (ementas/tópicos) específica para os alunos entre o quinto e o décimo anos. A partir de então, as disciplinas de Computação não são mais obrigatórias. No entanto, podem ser utilizadas como atividade extraclasse. A motivação do país para a adoção da Computação nas escolas foi baseada em dois objetivos:

*Permitir que os estudantes saibam lidar com artefatos e tecnologias da informação na vida cotidiana; e*

*Promover o ensino avançado da Computação após o ensino básico (técnico e superior)<sup>44</sup>.*

Todos os tópicos que fazem parte da integração da Computação no currículo são especificados no documento chamado *Einheitliche Prüfungsanforderungen – Informatik* (em tradução livre, Normatização dos Requisitos para Ciência da Computação). Alguns dos tópicos que o estudante pode escolher, segundo Brinda et al (2009), são: modelagem entidade-relacionamento; autômatos; modelagem algorítmica; interação homem-máquina; privacidade; segurança; arquitetura de computadores; computabilidade; eficiência; e linguagens de programação orientadas a objetos.

Em 2008, a Associação da Informática (*Gesellschaft für Informatik*) da Alemanha, publicou o documento guia para a segunda metade do ensino fundamental. O documento foi modelado dentro do padrão definido pelo *National Council of Teachers of Mathematics* (NCTM<sup>45</sup>) e organizado em conteúdos

---

<sup>44</sup> Na UE, a denominação dos níveis de ensino utiliza fundamental inferior e fundamental superior.

<sup>45</sup> <http://www.nctm.org/> - acesso em 4/12/2018



(informações e dados; algoritmos; linguagens e autômatos; sistemas de Informática; Informática, homem e sociedade) e processos (modelagem e implementação; raciocínio e avaliação; estrutura e inter-relacionamento; comunicação e cooperação; representação e interpretação). O documento descreve as competências para o 8º, 9º e 10º anos do ensino alemão. Cabe observar que o foco está na modelagem computacional. Um ponto forte do sistema de ensino alemão está na disponibilidade de professores com formação adequada para ministrar as aulas. A capacitação necessária estabelece dois cursos de nível superior, além de uma preparação pedagógica, teoria educacional focada na área em que atua e estágio de dois anos.

No caso da Alemanha, a proposta separa o ensino de mídias digitais e TICs das aulas de Computação. Não foi encontrada referência ao uso específico do PC como proposta para o ensino de conceitos de Computação e, em 2013, a obrigatoriedade da Computação voltou a ser pauta do governo (SCHMUNDT, 2013) e foi apresentado ao parlamento em 2015 e aprovada.

As opções são muito diversificadas, por exemplo, a Alemanha também utiliza a robótica inteligente no ensino da língua, de forma multidisciplinar. O robô NAO (<https://www.aldebaran.com/en/cool-robots/nao>) é entregue a cada criança imigrante. Ele é pré-programado com tecnologia de Inteligência Artificial e interage com os alunos de forma afetiva, visando o ensino da língua alemã.

### 11.1.2 Argentina

Em 2013, a Fundação Sadosky (<http://www.fundacionsadosky.org.ar/>, acesso em 5/12/2018, *Investigacion y Desarrollo en TIC*) publicou o documento denominado “CC-2016: Una propuesta para refundar la enseñanza de la computación en las escuelas Argentinas”, que tem como principal objetivo conscientizar a comunidade científica da importância de uma mudança do ensino dos níveis fundamental e médio, com a introdução dos princípios de Computação em sua estrutura (Sadosky, 2013). O documento argumenta que o ensino da Computação é essencial para as grandes oportunidades proporcionadas pelas tecnologias. Além disso, a entidade alerta quanto à necessidade dos estudantes desenvolverem habilidades e competências essenciais para a vida moderna (Brackmann et al, 2016). A Argentina propôs o uso da metodologia do PC para os anos iniciais do ensino. Nesse caso, é um dos poucos países que adotou a metodologia do PC *unplugged* para as séries iniciais, em todo o território.

Em 2015, o Conselho Federal de Educação Argentino (CFE Argentina, 2015) apresentou um novo documento, movido pela necessidade estratégica de desenvolvimento econômico-social da nação, estabelecendo diversos elementos, dentre eles:

- Ensinar programação como parte do currículo, ou como atividade extracurricular nas escolas, durante os anos do ensino obrigatório;

- Criar a “Rede de Escolas que Programam” (REP), categorizadas conforme disponibilidade de professores com formação adequada e oferta de aulas de programação;
- Intensificar a propagação da REP em todas as jurisdições, até alcançar todas as instituições de ensino estatais, através de iniciativas de formação docente específica e em serviço;
- Criar o prêmio anual “Clementina”, para as produções de destaque na área de Informática, desenvolvidas pelos alunos, em escolas que fazem parte da REP;
- Criar o Repositório Nacional de Produções em Informática, tornando o acesso aberto a todos os demais membros da REP.

O documento foi aprovado, tendo como base as resoluções que tratam das políticas de inclusão digital, onde são também aprovadas intensificações no uso de TICs nas escolas, “para melhoria nos processos de ensino e aprendizagem”.

No que toca à formação de professores em exercício, a Universidade Nacional de Rosário (UNR), em conjunto com a Fundação Sadosky, desenvolveu um curso *lato-sensu*, denominado *Especialización Docente de Nivel Superior en Didáctica de las Ciencias de la Computación: Aprendizaje y Enseñanza del Pensamiento Computacional y la Programación en el Nivel Primario*. O curso é composto por 400 horas aula, sendo 80 a distância e cobrindo temáticas como: introdução do PC nas escolas encarregadas dos anos iniciais da educação, atividades de PC, programação, projetos integradores, organização de computadores, robótica, segurança e privacidade dos dados e avaliação (Isaías, 2017).

### 11.1.3 Austrália

O programa da Austrália não explicita foco no PC, mas sim no ensino da programação, independente da metodologia que venha a ser utilizada para seu ensino.

O Ministério da Educação publicou, em 2015, uma reestruturação no currículo das escolas públicas (ACARA, 2015). O modelo coloca a programação como uma das principais competências (e recebe doze milhões de dólares australianos) com o intuito de incentivar o desenvolvimento de recursos curriculares para o ensino da Matemática; dar suporte para introdução da programação em todos os níveis de escolaridade; criar uma escola piloto, baseada na estratégia *P-TECH-like Schools*<sup>46</sup>; financiar Escolas de Verão para estudantes nas áreas de Ciência, Tecnologia, Engenharia e Matemática (*STEM – Science, Technology, Engineering and Mathematics*). No quinto e sexto anos, os estudantes aprendem a codificar e, a partir do sétimo, aprendem a programar.

---

<sup>46</sup> Escolas P-TECH são organizações sem fins lucrativos e possuem um currículo e metodologia de ensino inovadores, caminhando juntas com a indústria e demais instituições de ensino locais, focando na formação do aluno para o mercado de trabalho. (<http://saf.org.au/p-tech-australia/>).

É interessante apontar que, com a introdução da Computação no currículo, os alunos deixaram de ter as disciplinas de História e Geografia como obrigatórias. Essas passaram a ser opcionais (Davis, 2015). Também é de notar o objetivo das organizações P-Tech.

### 11.1.4 Coreia do Sul

Em um relatório de 2012<sup>47</sup>, o governo sul-coreano apresentou uma revisão de sua política de uso de tecnologias na Educação. Até então, a Coreia do Sul vinha utilizando, prioritariamente, Jogos Educacionais para o ensino básico. No documento, o Ministério da Educação local afirma que “já era hora de optar por sistemas mais criativos e menos lúdicos”. O país possui a Internet mais rápida do mundo e também o maior percentual de acesso por pessoa. A decisão do país foi a de investir (vinte e quatro bilhões de dólares em 2013) na compra de equipamentos (*tablets*, celulares e computadores) para os alunos, visando *e-learning* e *home-learning*. A política para *software* focou no conteúdo digital (livros e Objetos de Aprendizagem). O mesmo relatório aponta para o incentivo ao desenvolvimento próprio desses recursos.

Em 2015, o debate voltou a focar tanto na oferta como na matrícula nas aulas de Informática, devido aos seguintes motivos:

- Diminuição da carga horária exigida pelas disciplinas eletivas;
- Ausência de uma regulamentação específica das aulas de Informática;
- Ausência de uma avaliação institucional relativa às aulas de Informática. Atualmente, apenas 10% das escolas do ensino fundamental ofertam aulas de Informática e menos de 5% dos alunos possuem a oportunidade de participar delas. Para que essa realidade mude, o governo está redigindo um novo currículo, que deve entrar em vigor em 2018.

A destacar, ainda, que esse debate incluiu os temas sobre Ética na Informática e Cibercrime, como importantes para serem considerados no conteúdo (Choi, 2015).

Como a Coreia é utilizada como um exemplo de superação na área da educação, optou-se por trazer um pouco da história do emprego da tecnologia no ensino deste país. A Coreia do Sul iniciou a utilização de computadores, em sala de aula, desde a década de 1970; em 1987 foi introduzida a alfabetização digital e, a partir de 1992, o foco foi o ensino das TICs. Em 2004, 80% das escolas já possuíam aulas de Informática e 40% de codificação. Em 2007, os educadores coreanos sugeriram que os computadores não deveriam ser utilizados apenas para aulas de alfabetização digital, mas também poderiam ter um papel na solução de diversos tipos de problemas do mundo real, ou seja, utilizar uma abordagem multidisciplinar.

---

<sup>47</sup> Disponível em: [go.nmc.org/mralbert](http://go.nmc.org/mralbert) - acesso em: março/2018.

Como resultado, as escolas mudaram o nome da disciplina “Computadores” para “Informática”, onde o foco principal, paradoxalmente, passou a ser os princípios e conceitos da Computação (métodos e procedimentos, algoritmos simples e circuitos lógicos), desde o ensino fundamental, como disciplina eletiva.

### 11.1.5 Escócia

No Reino Unido não existe uma padronização do ensino fundamental em termos curriculares. Por esse motivo, optou-se por trazer também o caso particular da Escócia, onde a Computação é parte do currículo do ensino fundamental desde 1980, iniciando aproximadamente aos 14 anos de idade dos alunos. As escolas trabalhavam com o foco no ensino da programação, mas esse modelo foi alterado e o foco foi ampliado para Sistemas de Informação. Atualmente, no último ano do ensino fundamental, o estudante aprende conceitos e prática do desenvolvimento de *software* e de bancos de dados relacionais. O ensino desses dois tópicos e a sua real necessidade, nessa etapa da formação regular, tem sido questionado.

No histórico da Escócia, pode-se destacar que, em 2004, ocorreu uma troca de currículo e juntamente com essa foi realizada uma pesquisa denominada *Computer Science: What do pupils think?* que constatou que os estudantes consideravam as aulas de Informática tediosas, pois o novo currículo incentivava o ensino de aplicativos de escritório, o que gerou uma diminuição de interesse dos alunos pela disciplina. Os resultados da pesquisa provocaram uma nova mudança, em 2011, com o lançamento do *Curriculum for Excellence*, um documento que rege o sistema de ensino na Escócia dos 3 (três) aos 18 (dezoito) anos. Essa proposta está dividida em duas categorias:

1. TICs para melhoria da aprendizagem; e
2. Ciência da Computação contextualizada no desenvolvimento de competências tecnológicas e conhecimentos. Embora essa segunda categoria esteja definida dessa forma, na proposta da Escócia, também não se encontra menção explícita à adoção do PC para o ensino da Computação.

### 11.1.6 Estados Unidos da América

Em 2014, o Governo Obama emitiu um amplo plano de ação para expandir a Ciência da Computação nas salas de aula do país (*The White House*, 2016). Assinada em 2015, a Lei Federal *Every Student Succeeds Act* (ESSA) é responsável pelas políticas públicas na área da educação do país. Nesse documento são detalhadas, desde a forma como ocorrem os financiamentos, até a maneira como as escolas públicas são avaliadas. É importante lembrar que o ensino privado nos EUA não é fiscalizado pelo governo. Essa lei coloca a Ciência da Computação em condições de igualdade com outras disciplinas acadêmicas, tais como Matemática, Geografia, História, Inglês e Ciências. O documento não define como a implantação deve ocorrer, porém incentiva sua adoção e permite a obtenção de recursos para tal (âmbito federal). Mesmo após a assinatura da Lei, de acordo com Code.Org (2016)

e Guzdial (2014), o ensino da Ciência da Computação não é obrigatório, porém existem iniciativas estaduais que permitem a substituição de disciplinas de Matemática, Ciências, Língua Estrangeira e outras, por Computação.

Os estados participantes seguem o currículo proposto pela *Computer Science Teacher Association* (CSTA), denominado “*A Model Curriculum for K–12 Computer Science* (CSTA/ISTE, 2011), em que se encontram aconselhamentos, devidamente estruturados, para o ensino da Computação nas escolas, desde o jardim de infância até o último ano do ensino médio, além de exercícios que podem ser realizados em sala de aula.

Por outro lado, o lançamento do projeto Code.Org (2015) atraiu a atenção dos estudantes, pais e escolas no país, provocando impacto sobre a importância da Computação. Em pesquisa encomendada pela Google a Gallup-Google (2016), entre diversos dados coletados, destacam-se para esse trabalho: 1) 90% dos pais querem que seus filhos tenham aulas de programação nas escolas e 2) 50% dos pais acreditam que a Computação é uma competência tão importante quanto ler, escrever e fazer cálculos.

Através de propostas de leis lideradas pelo senado, o governo estadunidense liberou, em 2016, o montante de quatro bilhões de dólares em recursos para os estados e cem milhões de dólares diretamente para os Estados membros, buscando aumentar o acesso a disciplinas de Computação; capacitar professores; expandir o acesso a materiais de instrução de alta qualidade; e construir parcerias regionais efetivas, visando à melhoria do ensino da disciplina, no país, *The White House* (2016).

Em resumo, o objetivo específico é o de incentivar a oferta de disciplinas de Computação, além de formar professores e equipar as escolas para atender a demanda da área, tendo em vista que a disciplina ainda não é obrigatória. Segundo Cuny (2017), algumas cidades ou distritos como: *New York City*, *San Francisco*, *Broward County* (FL), *Rhode Island*, *Virginia* e *Chicago* tomaram medidas que tornaram a Computação obrigatória em suas escolas. Ou seja, nos EUA, existem escolas que optaram por adotar o PC, como constatado em artigos específicos comentados neste texto e outras que utilizam outras metodologias.

Cabe também lembrar, que o país possui o maior mercado mundial de *software* educacional. Na área de Tutores Inteligentes (ensino personalizado), as empresas *Alelo*, *Carnegie Learning* e *Prazas Learning Inc* esperam movimentar, em 2018, um valor de US\$ 6,7 bilhões dólares, com seus sistemas *Enskill*, *MathIA* e *Milka*, e *Thinkster Math*. Desses sistemas, apenas o *Enskill*, da *Alelo*, está voltado para o ensino de línguas e cultura. Os demais são para o ensino de matemática, para o K-12.

Os EUA encontram-se no processo de criação da próxima geração do currículo de referência para as disciplinas da área das Ciências, denominado *Next Generation Science Standards*<sup>48</sup> (NGSS). Esta proposta possui como inspiração o sistema de ensino de países como Finlândia, Canadá, Hungria, Singapura, Coreia do Sul, China

---

<sup>48</sup> <https://www.nextgenscience.org> – acesso em 5/12/2018

e Japão, tendo em vista a alta pontuação em testes internacionais de seus alunos, nas áreas de Ciência e Matemática. Alguns desses países são abordados neste texto, outros não, pois não foram encontrados documentos que permitissem uma análise de suas políticas educacionais para o ensino da Computação. Em alguns casos a inexistência se deve ao fato de o país não possuir uma política única centralizada.

### 11.1.7 Estônia

A Estônia, o país que desenvolveu o Skype, é considerado um dos mais dependentes de *Internet* no mundo, atrás somente da Coreia do Sul e dos EUA, e sua ascensão digital ocorreu alguns anos depois de sua independência, no início da década de 90 (Mansel, 2013).

O projeto de introdução à programação no currículo foi proposto na década de 90. A motivação veio através de membros do governo, após a independência, que haviam vivido nos EUA e acompanhado a ascensão da robótica industrial naquele país. No final da década de 90, através do projeto chamado *Tiger's Heap*, todas as escolas do país já possuíam Internet. O projeto também incentiva o ensino de programação para crianças, a partir dos 7 anos, levando-se em consideração a seguinte lógica:

*se você aprende regras gramaticais aos 7 ou 8 anos, o que isso difere em relação às regras de programação?*

Hoje, o ensino da programação existe em nível nacional e abrange a educação fundamental, o ensino médio e a formação técnica. Mas, a disciplina é facultativa em todos os níveis. As escolas utilizam, em sua maioria, duas plataformas: a chamada *ProgeTiiger Kogumik*<sup>49</sup>, onde são oferecidas diversas ferramentas e suporte aos professores, para incentivar o ensino da programação nas escolas; ou a chamada *Oppematerjalid*<sup>50</sup>, que oferece uma série de sugestões de uso de atividades e tecnologias de acordo com o nível escolar, área, linguagem de programação e sistema operacional (espécie de planos de aulas). Essas plataformas foram desenvolvidas no próprio país com apoio de fundos europeus (Mansel, 2013).

### 11.1.8 França

Em 2015, o Ministério da Educação francês anunciou que as escolas iriam gradualmente receber aulas de programação, a partir de setembro do mesmo ano, como atividades extracurriculares. O argumento utilizado pelo ministério foi que os alunos necessitam falar francês, contar, calcular, compor e decompor números na Matemática, porém a escola não pode ignorar a importância da digitalização em

---

<sup>49</sup> <http://www.progetiiger.ee/?q=> - acesso em 5/12 2018

<sup>50</sup> <http://www.progetiiger.ee/oppematerjalid> - acesso em 3/4/2018



todas as disciplinas, pois o aprendizado da lógica facilita o entendimento de conceitos do mundo em geral. O ministro francês da época chegou a afirmar:

“A dúvida não é mais se devemos ou não ensinar Computação e programação aos alunos, mas como, qual o propósito e em que nível o ensino deve ser introduzido no currículo” (Johnson, 2015).

De acordo com o Conselho Superior Curricular, tem-se como objetivo repensar os métodos de ensino e avaliação, além de modernizar o currículo. Para isso, foram definidos os primeiros requisitos para o ensino no país. Segundo (Fleury e Neveux, 2014) os princípios são: estabelecer os fundamentos das Linguagens de Programação e desenvolver aplicativos com a utilização de algoritmos simplificados.

A intenção de encorajar os estudantes para a carreira na Computação é uma tentativa de torná-los cidadãos do mundo, o qual, a cada dia, está mais conectado. Essas mudanças no currículo não têm a intenção de torná-los a todos programadores, mas facilitar a identificação de talentos, em um setor de competição global.

### 11.1.9 Finlândia

De autoria de Mykkänen e Liukas (2015), o guia *Koodiaapinen 2016* teve como objetivo convencer o governo da Finlândia a implantar o ensino da Computação nas escolas, desde o ensino fundamental. Seu conteúdo é composto por um passo-a-passo, leve e direto, de como utilizar nas escolas o PC. O ensino da metodologia é feito através de um MOOC<sup>51</sup>. Desde sua publicação, esse guia é a principal referência para o legislativo na adoção do ensino de Computação, no currículo, que, a partir de 2016, tornou a disciplina obrigatória desde os primeiros anos da educação.

Em 2015, o ministro da educação, Kai Mykkänen, (Weinberg, 2015) defendeu que:

as pessoas precisam dominar minimamente a linguagem do computador. Quem não investir nisso vai ficar para trás”.

Ele ainda complementa com dados:

se seu país não acelerar, acumulará um déficit de 17.000 programadores em quatro anos.

O guia *Koodiaapinen 2016* propõe as atividades de acordo com o ano escolar:

- Anos 1 e 2: utilizar o PC através de atividades lúdicas;
- Anos 3 a 6: desenvolver atividades no computador utilizando programação visual e aprender a controlá-lo sem medo de errar;

---

<sup>51</sup> <http://koodiaapinen.fi/mooc/> - acesso em 5/12/2018

- Anos 7 a 9: familiarizar os alunos com uma linguagem de programação tradicional. Alguns fatos a respeito da implantação do novo currículo incluem vínculos estranhos, como apresentado no guia *Koodiaapinen 2016*: para que um aluno de sexta série tenha um bom rendimento na disciplina de Matemática, o estudante deve saber criar um programa simples utilizando uma linguagem de desenvolvimento visual (Scratch);
- Uma formação gratuita para professores é realizada por um MOOC e que também dispõe de uma biblioteca aberta com sugestão de conteúdos e atividades.

De forma concreta, a proposta da utilização do PC nas escolas da Finlândia é um esforço conduzido pelo psicólogo educacional e pesquisador de *design* Tarmo Toikkanen, da Alto University (<http://www.aalto.fi/en/> - acesso em 5/12/2018), e construída com os esforços de voluntários e poucos recursos. Ou seja, essa iniciativa não foi institucionalizada pelo governo do país.

Nos países nórdicos, Finlândia Suécia e Noruega, por orientação do relatório *Horizon* (2017), muitas escolas optaram pelo uso de sistemas com base na Inteligência Artificial, como Sistemas Tutores Inteligentes (o mais conhecido é o Albert, utilizado em 26 escolas na Suécia) e *Smartbooks* personalizados (que vão sendo construídos de acordo com o estilo de aprendizagem e as necessidades de cada aluno). O uso dos 'livros inteligentes' elimina a utilização de um livro padronizado. Cada aluno tem o seu personalizado dentro do tema previamente informado, ao sistema computacional, que gera os livros, pelo professor. A Suécia foi o país que implementou rapidamente os conselhos do relatório *Horizon*. Esse relatório também indica o uso de Impressoras 3D, de Realidade Virtual e de Realidade Aumentada, como tecnologias educacionais.

### 11.1.10 Israel

Em Israel, um comitê foi nomeado pelo Ministério da Educação do país, em 1990, com o objetivo de construir um currículo de Computação para as escolas. Esse currículo foi implantado em 1995 e os princípios que norteiam o trabalho da comissão são, segundo Hazzan *et al* (Hazzan *et al*, 2008):

1. Ciência da Computação é uma disciplina ensinada, da mesma forma que outras disciplinas científicas;
2. o programa está concentrado nos principais conceitos e fundamentos da área;
3. dois programas diferentes são necessários, um para os estudantes, com apenas interesse geral em Computação, e outro, mais amplo e profundo, para aqueles com especial interesse pela área;
4. cada um dos dois programas têm unidades obrigatórias e eletivas;
5. questões conceituais e experimentais são entrelaçadas durante todo o programa;

6. dois paradigmas de programação bastante diferentes são idealmente ensinados, o procedimental e o declarativo (nota dos autores: esse talvez seja o principal ponto da política de Israel);
7. um laboratório de Informática, bem equipado e mantido, é obrigatório em todas as escolas;
8. livros didáticos e guias de ensino têm sua produção incentivada para todos os conteúdos do programa;
9. professores com formação adequada são os profissionais indicados para ensino da Computação nas escolas.

Mais detalhes a respeito do currículo de Computação para as escolas de Israel podem ser conferidos em Gal-Ezer et al (1995) e Gal-Ezer et al (1999), Gal-Ezer (2015) da Open University of Israel.

### 11.1.11 Grécia

Na Grécia, as crianças têm aulas de Computação desde a terceira série (oito anos de idade) em disciplinas denominadas “*Desenvolvimento de Aplicativos em Ambientes de Programação*”, com conceitos básicos de Informática, tais como: o que é um computador; partes que o compõem; seu uso na rotina diária; e como usá-lo. Além de atividades *unplugged*, as crianças também usam computadores para aprender através de jogos (e.g. Logo) e aplicativos simples de comunicação pela Internet (Balanskat; Engelhardt, 2014). A partir dos 10 (dez) anos, os estudantes aprendem a desenvolver aplicativos simples para a resolução de problemas e, nessa altura, podem decidir entre as seguintes áreas de interesse: aprofundamento da alfabetização digital; processamento de texto, imagens e arquivos; uso e criação de gráficos; programação de computadores; busca de informações na Internet e apresentação de dados; comunicação pela Internet; e computadores na vida diária.

Quando essa prática foi implantada, não houve o preparo de professores de Informática ou a criação de material didático. Entretanto como, de um modo geral, os professores já possuíam um preparo avançado em Computação, pois os professores eram formados em Engenharia de Software e Ciência da Computação, já estavam mais que habilitados para a proposta curricular. No entanto, segundo Balanskat e Engelhardt (Balanskat; Engelhardt, 2014), eles, muitas vezes, deixam a desejar na metodologia de ensino, devido à falta de conhecimentos pedagógicos em sua formação. Na segunda metade do ensino fundamental, as aulas de Informática são obrigatórias desde 1993, compreendendo dois níveis de idade. Entre 12 e 14 anos, baseadas em quatro áreas de ensino: aprofundamento em *hardware* de computadores; uso avançado de interface de usuário em sistemas operacionais; busca, processamento e apresentação de informações, utilizando a Internet e aplicativos de escritório; uso de computadores para a vida profissional ou de uso rotineiro.

Na faixa dos 15 aos 18 anos, as atividades abrangem o pensamento algorítmico, programação e amplo uso da Internet, para apresentações e projetos. Para a faixa dos 15 anos, existe material didático padronizado, mas os professores

podem criar seu próprio material, desde que atenda aos requisitos do currículo. O currículo é composto pelas mesmas áreas do ensino fundamental, porém são abordados de forma mais aprofundada. E, no último ano do ensino médio, para aqueles que optarem por uma linha tecnológica, os conceitos de Ciência da Computação tornam-se obrigatórios. Nessa etapa, a ênfase está no pensamento algorítmico, nos princípios da Ciência da Computação e nos conceitos de programação. Mais uma vez, existe material didático padrão, porém o professor pode criar seus próprios conteúdos.

### 11.1.12 Portugal

O relato sobre Portugal tem origem em visita realizada por Rosa Vicari, um dos autores deste texto, a uma das escolas piloto. Portugal selecionou um conjunto de escolas, distribuídas pelo país (zonas urbanas e rurais), de ensino fundamental, onde toda a metodologia de ensino foi transformada. Essas escolas, ao invés de mesas e cadeiras, possuem tapetes para o movimento dos robôs. Todas as disciplinas utilizam a metodologia ativa (baseada em projetos de interesses das próprias crianças). Os conceitos de cada uma das disciplinas são apresentados de forma integrada aos projetos. Semanalmente são oferecidas palestras motivacionais sobre uma área profissional.

Os professores foram previamente capacitados para essa nova realidade. Os resultados poderão ser observados com o tempo. O projeto explicita o termo PC (no caso *plugged*) e ele está presente em todas as atividades realizadas durante o período escolar. Os alunos aprendem a programar os robôs para solucionar problemas.

As bases da proposta: a Direção-Geral da Educação (DGE) tem um novo programa decorrente do projeto-piloto de Iniciação à Programação no 1.º Ciclo do ensino básico, que envolveu mais de 70.000 alunos, entre 2015 e 2017. A partir de setembro de 2018, o ensino de programação e robótica vai ser obrigatório em todas as escolas de ensino básico, através da disciplina de Tecnologias de Informação e Comunicação (TIC). Por agora, o Governo avança com um novo programa em centenas de escolas públicas e privadas, que teve início em outubro de 2018.

A iniciativa “Programação e Robótica no ensino básico” pretende dinamizar atividades de programação e robótica com os alunos do 1º ao 9º ano de escolaridade “através das ofertas complementares, das atividades de enriquecimento curricular ou da oferta de escola, opção que fica ao critério dos órgãos do agrupamento escolar”. As escolas que desejam participar dessa iniciativa precisam candidatar-se junto à DGE.

O documento base está organizado em quatro áreas das ciências da computação, Pensamento Computacional, Algoritmia, Programação e Robótica, estruturados de acordo com padrões de desenvolvimento: iniciais, intermédios e avançado, em conformidade com os estágios de cada criança e jovem. Para maiores detalhes ver <https://tek.sapo.pt/noticias/computadores/artigos/programacao-e->

[robotica-no-ensino-basico-ganha-linhas-orientadoras](#) acesso em 5/12 2018. O programa possui parceria com a Microsoft.

### 11.1.13 Reino Unido

O Currículo do Reino Unido está centrado nas Tecnologias da Informação e Comunicação e é regido pelo Currículo Nacional para TICs. Os detalhes podem ser encontrados em: <https://www.gov.uk/government/publications/national-curriculum-in-england-computing-programmes-of-study/national-curriculum-in-england-computing-programmes-of-study> - acesso em 7/12/2018.

Esse documento possui uma especificação de alto nível, porém, quando colocado em prática, pouco era aplicado de fato além dos pacotes de escritório (processadores de texto, planilhas eletrônicas, editores de apresentação, etc.). Em 2011, houve uma revisão do Currículo Nacional, onde a disciplina de Computação foi considerada obrigatória em todos os quatro níveis (*Key Stages – KS*), independentemente da idade dos alunos. Após concluído os KS, os alunos são preparados para as provas GCSE (*General Certificate of Secondary Education*), onde podem comprovar suas competências ocupacionais em diversas áreas, (certificando o conhecimento e compreensão). Caso o aluno opte pela continuidade no ensino superior, são necessários mais dois anos de estudos e a realização das provas GCE (*General Certificate of Education*), também conhecidas como *AS-A Levels*. Tanto o GCSE quanto o GCE possuem Computação como opção de certificação nas instituições homologadas para aplicação das provas (SCC, 2015) (*Teaching Jobs London, 2013*).

Rossi (2015) relata em seu trabalho que os pais são favoráveis à introdução da programação nas escolas. Em sua pesquisa no Reino Unido, aponta que 60% dos pais e 75% dos alunos preferem aulas de Python, ao invés do idioma francês, no ensino Fundamental, em um universo de 1000 crianças (5 a 11 anos de idade), 1000 crianças (11 a 16 anos de idade) e 1000 pais. Um dado curioso dessa pesquisa é o fato das crianças preferirem as aulas de programação, por essas serem “mais fáceis” em relação ao francês. Cabe ainda lembrar que os territórios que compõem o Reino Unido possuem autonomia para definir seus currículos.

Dentre os países que já estão implementando essa ideia da disciplina de programação poder substituir outra disciplina do currículo, o líder é o Reino Unido, no ensino fundamental, onde o Currículo Nacional, desde 2011, inclui a disciplina de Computação como obrigatória em todos os seus quatro níveis (*Key Stages – KS*), independentemente da idade dos estudantes.

### 11.1.14 Resumo Comparativo Entre as Políticas

Cabe salientar que a grande maioria dos países que adotaram o PC o fazem com o uso de máquinas (computadores e robô).

Um comparativo pode ser observado na Tabela 11.1.

**Tabela 11.1: Integração da Computação no currículo**

	FUNDAMENTAL	ENSINO MÉDIO INFERIOR (GERAL)	ENSINO MÉDIO SUPERIOR (VOCACIONAL)	ENSINO MÉDIO SUPERIOR (GERAL)	ENSINO SECUNDÁRIO SUPERIOR (VOCACIONAL)	DEPENDÊNCIA REGIONAL OU CURRÍCULO ADOPTADO
<ul style="list-style-type: none"> <li>● Facultativo</li> <li>● Compulsório</li> <li>○ Em processo de adoção</li> </ul>						
ALEMANHA				●	●	●
ARGENTINA	●	●	●	●	●	●
AUSTRÁLIA	●	●	●	●	●	●
ÁUSTRIA		●		●	●	●
BÉLGICA (Holanda)	●	●	●			
BULGÁRIA				●	●	
CORÉIA DO SUL	●	●	●	●	●	●
DINAMARCA		●		●	●	
ESCÓCIA	●	●	●	●	●	●
ESLOVÁQUIA	●	●	●	●	●	
ESPANHA	●	●		●		● ●
ESTADOS UNIDOS	●	●	●	●	●	●
ESTÔNIA	●	●	●	●	●	●
FINLÂNDIA	●	●				
FRANÇA	●	●		●		
GRÉCIA	●		●	●	●	●
HUNGRIA				●	●	
IRLANDA		●				●
ISRAEL	●	●	●	●	●	
LITUÂNIA		●		●		
MALTA				●		
POLÔNIA		●		●	●	●
PORTUGAL		●			●	
REINO UNIDO (INGLATERRA)	● ●	● ●		● ●		
REPÚBLICA TCHECA					●	●

Fonte: adaptada de Balanskat e Engelhardt, 2015

A tabela inclui diferentes países com diferentes sistemas de ensino, o que torna difícil uma padronização de series e idades. A nomenclatura utilizada foi a proposta por Pelgrum (2001) e Balanskast e Engelhardr (2015). No entanto, foram incluídos, também, países que não fazem parte da UE. Para os casos dos países de fora da UE, com exceção de Israel, existem diferenças de 1 (um) ou 2 (dois) anos na idade em que os alunos ingressam nas diferentes categorias (fundamental, médio inferior, etc.).



Após a revisão bibliográfica sistemática, é apresentada a Tabela 11.2, com um resumo comparativo entre os diversos países estudados, para os quais foram encontrados dados que possibilitassem a comparação.

*Tabela 11.2: Quadro Comparativo das políticas adotadas nos países estudados*

País	Ano de Adoção	Fundamental	Médio	Técnico	Disc. específica	Modo de integração
ALEMANHA	2004	F	FO	-	Não	-
ARGENTINA	2015	F	F	-	Sim	Variável
AUSTRÁLIA	2015	O	-	-	Não	-
ÁUSTRIA	2015	O	O		Não	Variável
CORÉIA DO SUL	2007	F	F	-	Não	Sim
DINAMARCA						
ESCÓCIA	1987	F	F	-	Sim	Variável
ESTADOS UNIDOS	2015	F	F	-	Sim	Variável
ESTÔNIA	1996	F	F	F	Sim	Variável
FINLÂNDIA	2016	O	-	-	Sim	Não
FRANÇA	2016	FO	O	-	Sim	Não
GRÉCIA	1993	O	O	-	Não	Sim
ISRAEL	1990	O	O	-	Sim	EE
PORTUGAL	2015	F	O	-	Sim	EE
REINO UNIDO (INGLATERRA)	2011	O	O	-	Sim	Sim
REPÚBLICA TCHECA						

F: facultativo O: Obrigatório FO: em algumas regiões, facultativo, em outras, obrigatório

EE: a escola escolhe V: Variável

Fonte: adaptada de Brackmann 2017

Para os demais países utilizados no relato não foram encontrados dados que permitissem completar essa tabela. Apenas na Estônia é encontrada menção ao ensino técnico e o ensino de conceitos de Computação.

## 11.2 Iniciativa Privada e Organizações Não-Governamentais

A seguir são apresentadas algumas instituições não governamentais com iniciativas ou sugestões para diferentes formas de incluir a Ciência da Computação e as tecnologias, em geral, no Ensino Fundamental. Cabe lembrar que estas instituições possuem alcance limitado, muitas vezes atendendo à interesses particulares e não substituem a atuação do Estado.

### Iniciativas Internacionais

#### 11.2.1 European Schoolnet

Rede de troca de experiências (<http://www.eun.org> – acesso em 18/12/2018), sem fins lucrativos, entre 31 Ministérios da Educação da Europa, com sede em Bruxelas (Bélgica), e que tem como foco principal a inovação do ensino e aprendizagem na área de TICs. Entre seus objetivos estão: criar políticas para regulamentar o ensino de Computação na Europa e identificar evidências escaláveis e com possibilidade de transferência de práticas e prioridades relacionadas à educação no continente.

#### 11.2.2 European Commission (Europa)

Os dados têm por base a publicação do Departamento de Ciência e Conhecimento da Comissão Europeia (CE), Joint Research Centre (JRC), em que foram encontrados dados a respeito de 9 (nove) países (Áustria, Rep. Tcheca, Dinamarca, Finlândia, França, Grécia, Hungria, Itália, Lituânia) que visavam fornecer *subsídios*, baseados em evidências, para o processo de formulação de políticas na Europa (Bocconi *et al*, 2016). O documento apresenta dois pontos básicos:

1. A motivação dos países em introduzir o PC em seus currículos e as implicações políticas; e
2. Práticas na introdução do PC na Educação Básica.

Isso não significa que todos esses países já tenham optado por introduzir o PC.

Em relação ao primeiro item 1 (Motivação para adotar o PC...), os países possuem diferentes visões para adotar ou não o PC no seu currículo. Mas, no geral, os países concordam com algumas vantagens que o PC pode promover, como as habilidades de pensamento lógico e de resolução de problemas. Uma quantidade menor tenta atrair mais estudantes para a área da Computação ou simplesmente tem a intenção que seus estudantes saibam programar e codificar. Poucos países têm uma visão mais abrangente do PC, para além do ensino de programação, e de

que essa formação pode vir a proporcionar empregos qualificados aos seus cidadãos.

As características do PC que estão sendo adotadas e implementadas na UE são:

- Habilidades de pensamento lógico;
- Habilidades de resolução de problemas;
- Habilidades de programação e codificação;
- Atrair mais estudantes para Ciência da Computação;
- Empregabilidade no setor de TICs.

O país que mais avançou, mesmo que pelo esforço de um grupo de professores e não pelo esforço do Governo, é a Finlândia que já busca atender a todos esses objetivos. Também se pode destacar: o Reino Unido, em particular a Escócia, mais a Hungria, a Itália e a Lituânia. Outros países da UE que também vêm avançando rumo à adoção do PC, mas mais lentamente, são a França, a Dinamarca e a Áustria.

Os demais países da Europa ou continuam em estudos ou começaram a aplicar o PC recentemente. Ainda, alguns países como, por exemplo, a Itália, concedem aos professores a liberdade cultural no uso dos métodos didáticos, cabendo ao Estado assegurar uma rede de instituições escolares obrigatórias e gratuitas para todos.

Em relação às práticas no emprego do PC na Educação Básica, o relatório da UE apresenta sugestões que podem servir de inspiração a outros países no processo de inserção do PC no currículo. A proposta é dividida em quatro dimensões, sendo elas: *a abrangência da integração, a sistemática de sua implantação, a consolidação do entendimento e as políticas de apoio*. A seguir, cada uma das dimensões é melhor detalhada:

**Integração:** definir de maneira clara a integração do PC no currículo é crucial. Uma estratégia recomendada é iniciar a implantação através de objetivos específicos, incluir as partes interessadas no processo e incluir o PC nas escolas gradativamente. Para que a adoção seja adequada, é necessário um planejamento cuidadoso e selecionar práticas pedagógicas compatíveis, incluindo ferramentas de ensino e estratégias de avaliação em todos os níveis.

**Implantação Sistemica:** a literatura é clara no tocante à importância dos métodos de avaliação do PC através da produção de artefatos pelos estudantes (isto é, jogos ou modelos), testes com múltiplas alternativas, critérios avaliativos ou alteração de códigos para atingir um objetivo específico. Como visto neste texto, não há ainda um consenso quanto à melhor alternativa para avaliar os estudantes, sendo necessário o desenvolvimento/adequação de métodos para tal. Para que isso ocorra, também é necessária uma formação continuada e constante de professores.

**Consolidação:** a pluralidade de significados do PC pode causar confusão, porém esse problema é solucionado com a inclusão de especialistas e parcerias com

outros países que já estejam utilizando a metodologia de uma forma similar à pretendida. No momento em que o entendimento do PC se encontrar consolidado, torna-se mais fácil assinalar onde é a fronteira entre o PC e o ensino das TICs, assim como identificar conteúdos em comum.

*Apoio:* a troca de experiência, com especialistas de outros países, com iniciativas prévias e as partes interessadas (por exemplo, a indústria, ONGs e fundações que atuam na área), pode fornecer informações relevantes, especialmente porque as razões para integrar o PC no currículo são semelhantes na maioria dos países. É imprescindível que, no momento de implantação do novo currículo, as partes interessadas, principalmente os professores, sejam informadas de maneira clara e objetiva quanto às respostas às seguintes perguntas: “O que é o PC?”, “O que requer/demanda?”, “Quais são os benefícios educacionais?”, “Quais os limites das diferentes abordagens (*plugged*, *unplugged*, concorrente, não-determinístico)?”, “Quais as críticas que se faz a essa abordagem?” É importante, principalmente que os professores apontem o quanto as suas escolas estão ou não equipadas de forma adequada para a adoção de uma ou de outra abordagem do PC. No caso do Brasil a participação ativa dos professores é fundamental, pois, o PC muito provavelmente será introduzido de forma interdisciplinar. Esses aspectos requerem que os resultados das ações implementadas venham a ser monitorados e analisados constantemente, para que sejam medidos os avanços ou retrocessos, de acordo com as metas políticas e estratégicas propostas para as diferentes realidades.

### 11.2.3 Code.Org (EUA/Global)

Lançado em 2013, nos EUA, o Code.Org (<https://code.org> – acesso em 10/12/2018) é uma organização sem fins lucrativos, que se dedica à expansão do acesso à Computação, incluindo minorias. Tem como visão possibilitar aos estudantes o acesso à Computação, além de acreditar que a referida disciplina deveria fazer parte do currículo escolar, juntamente com outras como, por exemplo, Biologia, Química e Álgebra. Tem como principais objetivos: aumentar a diversidade na Computação; inspirar estudantes; criar cursos voltados para a Computação; levar a Computação para as salas de aula; formar professores; atualizar o currículo nos distritos escolares; auxiliar na adequação de leis estaduais, para inclusão da Computação no currículo; e compartilhar o acesso ao material produzido pela ONG. Por exemplo, a code.org apresenta em sua página um livro digital, composto de várias lições, para diferentes idades. O conteúdo versa sobre os temas abordados nos pilares do PC. <https://code.org/curriculum/unplugged> (acesso em 10/12/2018) e existe versão apenas para a língua inglesa. O livro é chamado de: *Course A-F Curriculum Book*.

A ONG Code.Org é patrocinada por empresas como: *Amazon; American Airlines; Apple; Association for Computer Machinery (ACM); Computer Science Teachers Association (CSTA); Disney; Dropbox; Facebook; GitHub; Google; Khan Academy; Microsoft; Rovio; Salesforce; Zendesk*.

Uma das iniciativas dessa ONG, que ganhou visibilidade é a *Hour of Code* (Hora do Código). A Hora do Código é um movimento global que atinge dezenas de milhões de estudantes em mais de 180 países. Qualquer pessoa, independentemente do lugar, pode organizar um evento da Hora do Código e anunciar no sítio da organização. Tutoriais de uma hora estão disponíveis em mais de 40 (quarenta) idiomas. Não é exigida qualquer experiência dos participantes do evento. A idade mínima para a participação é de 4 (quatro) anos. O crescimento do evento pode ser constatado através dos dados: em 2013, 20 milhões e, em 2017, 25 milhões de estudantes em todo o mundo<sup>52</sup>

O principal objetivo da campanha é que dezenas de milhões de estudantes participem da Hora do Código no período pré-estabelecido pela ONG. No ano de 2016, a semana selecionada foi de 5 a 11 de dezembro, quando também é celebrada a Semana da Educação em Ciência da Computação. A semana é determinada como foco na promoção do evento, porém as atividades encontram-se disponíveis permanentemente no sítio. No Brasil, o principal parceiro da Hora do Código é o [Programae.org.br](http://Programae.org.br).

## 11.2.4 Instituto *Horizon*

Em uma linha diferente, que envolve a prospecção de tecnologias e necessidades futuras do mercado de trabalho, em 2017, o Instituto *Horizon* (<https://www.thehorizoninstitute.org/> - acesso em 10/12/2018), em um relatório de abrangência geral, apontou as seguintes tecnologias como sendo as que irão marcar presença em termos de desenvolvimento mundial na educação: ensino *on-line* (comum em muitos países, mas inexistente em tantos outros), para o prazo de um ano; Realidade Virtual e Tecnologias Vestíveis, para o prazo de dois anos; baterias com longa duração, Inteligência Artificial (em geral) e *displays* holográficos, para os próximos cinco anos.

Embora a proposta desse instituto seja diferente das propostas do PC, optou-se por sua inclusão pois o mesmo é o responsável pelo direcionamento das políticas de Ciência e Tecnologia no âmbito da União Europeia.

A prospecção realizada no presente trabalho não corrobora as propostas dos relatórios *Horizon* e de outras referências bibliográficas no que se refere à disseminação, em curto ou médio prazos, das tecnologias vestíveis/usáveis em sistemas educacionais. A exceção vai para fones de ouvido e para óculos inteligentes, com *design* e funcionalidades simplificadas. De acordo com os levantamentos de dados realizados neste estudo, tecnologias vestíveis serão ainda temas de pesquisa no curto e médio prazos, bem como a criatividade computacional e a abordagem ética para os sistemas de IA<sup>53</sup>. O Instituto *Horizon*

---

<sup>52</sup> <https://code.org/about/2013>

<https://code.org/about/2018> - acesso em 9/3/2018

<sup>53</sup> Disponível em: [https://papers.ssrn.com/sol3/papers.cfm?abstract\\_id=2586](https://papers.ssrn.com/sol3/papers.cfm?abstract_id=2586) - acesso em 13/3/2018

que atua na União Europeia, possui uma visão ampla do tema. Isto é, não apenas focada na Computação, mas sim nas TICs de forma geral.

### Iniciativas Brasileiras

No Brasil também existem algumas iniciativas dentre as quais foram selecionadas duas instituições, por utilizarem uma abordagem abrangente. É do conhecimento dos autores que existem outras tantas iniciativas, no entanto essas são de caráter localizado e, portanto, não estão sendo aqui contempladas.

#### 11.2.5 Programa.org.br (Brasil)

Essa iniciativa foi fundada em 2002, pelo empresário Jorge Paulo Lemann. A Fundação Lemann (<https://www.fundacaolemann.org.br> – acesso em 10/12/2018) é uma organização familiar, sem fins lucrativos, que atua em vários setores ligados à educação. A Fundação Lemann desenvolve e apoia projetos inovadores em educação, atua na realização de pesquisas para embasar políticas públicas no setor, oferecendo formação para profissionais da educação e aprimorando lideranças em diversas áreas. Segundo a própria fundação, as bases estão fundamentadas nos resultados obtidos em seus experimentos de uso de tecnologias na educação, em particular nos materiais de apoio pedagógicos e nos cursos gratuitos, utilizada pela fundação. As iniciativas e ações buscam contribuir para que o Brasil apresente, até 2018, soluções inovadoras, de qualidade, no cotidiano da educação de 30 (trinta) milhões de pessoas e na capacitação de 200 (duzentos) mil professores, a fim de garantir o aprendizado dos alunos.

O Programa.org.br é um movimento que tem como prioridade aproximar a programação do cotidiano de jovens de todo o Brasil e foi criado por a tecnologia ter um alto poder de atração de jovens. De acordo com a ONG, usá-la para a Educação pode fazer a diferença para muitas pessoas. A interface da organização com a população se dá através de um portal agregador de ideias, soluções e dicas. É parceira da Code.Org, no Brasil.

#### 11.2.6 Supergeeks.com.br (Brasil)

A SuperGeeks é uma das primeiras escolas de programação e robótica para crianças (a partir de sete anos) e adolescentes, no Brasil. As aulas baseiam-se na criação de *games*, aplicativos, robôs, sistemas, empreendedorismo e língua inglesa. A ideia de abertura da escola surgiu com um casal brasileiro que morava nos Estados Unidos, mais especificamente no Vale do Silício, que notaram que empresas e políticos estadunidenses estavam mobilizando-se para ensinar Ciência da Computação para crianças e adolescentes. Hoje, a SuperGeeks está se expandindo por todo o Brasil, através de franquias e unidades próprias. No final do ano de 2016, a rede já possuía 31 escolas em todo o Brasil. A SuperGeeks (<http://supergeeks.com.br> – acesso em 10/12/2018) utiliza quatro metodologias de ensino, sendo elas:



- *Game Learning*: utilização de *games* para promover o aprendizado de conceitos de programação e Ciência da Computação;
- *Gamification*: utilização de mecanismos de jogos, como pontos, recompensas e desafios de forma lúdica, para engajar os alunos no aprendizado. Os alunos recebem ou perdem pontuações, como se estivessem em um jogo;
- *Empreendedorismo*: os alunos aprendem a respeito do mercado de *games*, como projetá-los da melhor forma, quais os melhores canais de distribuição e divulgação. A partir da fase 3, os alunos serão incentivados a criar suas próprias *startups*, sozinhos ou em grupo, e lançar seus produtos no mercado;
- *Storytelling*: histórias inseridas em animações, livros e quadrinhos, para que os alunos entrem em contato com os conteúdos de forma mais prazerosa. Os conceitos estão inseridos nas histórias de aventuras.

Após o pioneirismo da SuperGeeks e com a demanda por esse tipo de iniciativa (escolas de programação e robótica para crianças), surgiram diversas outras iniciativas similares, tais como: Buddys, Ctrl+Play, DragonByte, Escolapixel, Happy Code, Konfidegeek e Madcode.

Além dessas iniciativas privadas, existem também iniciativas públicas, em particular do MEC, relacionadas com o ensino de Computação. No entanto, considerou-se que os leitores deste texto já as conheçam. Alguns dos programas do MEC para a área estão listados no item 15 - Anexos.

Cabe ainda lembrar que existem vários trabalhos acadêmicos que testam a introdução do PC (tanto *plugged* com *unplugged*), propriamente dito, como forma de ensino da Computação, em determinadas escolas ou turmas. Esses estudos acadêmicos estão distribuídos por diferentes cursos de diferentes níveis acadêmicos em vários estados do Brasil.

## 12. Análise Crítica

Na pesquisa bibliográfica, foram encontrados alguns autores que realizam uma análise crítica ao uso do PC. Por exemplo, Pappano (2017) relata que pesquisadores se preocupam com a tendência cultural atual de nos encontrarmos embriagados com a Computação, em detrimento dos conhecimentos obtidos de outras áreas.

Farag (2016), em seu blog, denominado *Please don't learn to code*, argumenta para que não seja ensinado programação para crianças. De acordo com sua visão, vender a codificação como um bilhete para a salvação econômica para as massas é desonestidade, pois o PC é apenas mais uma “moda” do Vale do Silício e que passará em breve, pois surgirão novas propostas baseadas em outros princípios. Ele coloca ainda, que a proliferação dos *Bootcamps*<sup>54</sup> e a divulgação massiva da Computação nos EUA produz uma quantidade de pessoas querendo abrir sua própria *Startup*, ou desejando ser Engenheiro de *Software*, sem uma formação adequada. O autor ainda afirma que as escolas não deveriam ensinar a codificar e se concentrar apenas na ideia da solução de problemas.

Um experimento interessante é apresentado por Grover e Pea (2013), e repetido de certa forma por Aggarwal et al (2017), em que os autores realizaram uma comparação entre o emprego do PC, com e sem o uso de máquinas, para a criação de um jogo. As atividades *unplugged*, a partir de um certo ponto, não foram tão eficazes quanto esperado e os estudantes sentiram-se frustrados. Isso poderá, por exemplo, distanciá-los da Computação, tendo em vista que programar usando máquinas é uma experiência fundamental e praticamente única dentro da abordagem do PC.

O trabalho de Fernandes (2001) e Fernandes & Menezes (2002), afirma que:

*...não adianta tornar disponível tecnologias de Informática, se rapidamente são abandonadas e substituídas. Por isto, tão importante quanto o ensino das tecnologias correntes é o ensino dos conceitos fundamentais da Ciência da Computação. Esses conceitos, além de proporcionarem um dos embasamentos teóricos para entendimento da ciência envolvida na Ciência da Computação, também propiciam o desenvolvimento de um raciocínio lógico e formal, assim como de habilidades que são exigidas no mundo atual.*

O mesmo trabalho, complementado em Bavaresco et al (2018), quando compara atividades *unplugged versus plugged*, especialmente quando da introdução de Algoritmos e Lógica do experimento realizado no Colégio de Aplicação da UFRGS, destaca que:

---

<sup>54</sup> Programa de ensino imersivo que foca nas habilidades consideradas relevantes em determinada área, para atuar no mercado do século XXI.

*Como o retorno da primeira parte da oficina, pela maneira (unplugged) como os conceitos foram trabalhados, acabou sendo não muito satisfatório, foi necessário começar a trabalhar os conceitos utilizando o computador.*

*O problema, pelo menos inicialmente, era como os assuntos fundamentalmente teóricos poderiam ser dados aos alunos. A saída encontrada para essa questão foi a utilização de simuladores e programas que permitissem que o foco do trabalho permanecesse o mesmo e os alunos não tivessem sua atenção desviada pela tecnologia utilizada.*

Outros países, como se pode constatar neste texto, consideram o ensino da Ciência da Computação de fundamental importância e relevância. No entanto, seu ensino, no Brasil, acontece, em geral, apenas nos âmbitos da graduação e pós-graduação. Antes disso, na maioria dos casos, existe uma formação, no máximo em Informática. O *ACM Model Curriculum for K-12. Computer Science* (CSTA, 2011) defende que é importante o desenvolvimento de habilidades computacionais na Educação Básica, no sentido da Ciência da Computação, visando abrir múltiplos caminhos profissionais, desenvolver a capacidade de resolver problemas e relacionar-se com outras ciências.

Bezerra e Silveira (2011) sugerem a formação em Computação de estudantes da educação básica, de acordo com uma abordagem tri-axial, que contempla conceitos, habilidades e competências. Nesse contexto, ensinar Computação não é o mesmo que ensinar Informática (o que é mais comum). O ensino de aplicativos como Word, Excel e navegadores não cabe na educação básica, pois equivaleria a ensinar a usar calculadoras e não a calcular, no ensino da Matemática (Nunes, 2008). O ensino de conceitos básicos de Computação nas escolas pode contribuir para desenvolver o raciocínio computacional das crianças, pelo seu caráter transversal às demais ciências, para formar cidadãos que vivem num mundo cada vez mais globalizado e para tornar o País mais rico e mais competitivo na área de Tecnologia da Informação (Nunes, 2011). Nesse sentido, seu ensino passa por noções de modelos computacionais, algoritmos, complexidade, autômatos, entre outros conteúdos (Nunes, 2008). Essa visão do PC é adotada em poucos países. Na sua maioria, ele se resume ao ensino de programação, Word e Excel.

## 13. Considerações Finais

As considerações finais deste trabalho visam resumir o que foi apresentado no estudo sobre o que é e como pode ser implementado o Pensamento Computacional, em particular na sua forma *unplugged*, mas, sobretudo, como se dá o ensino de Computação, no ensino básico em diferentes países, isto é, voltado para o ensino de programação.

Quanto à revisão bibliográfica, foi possível constatar que grande parte dos artigos se repetem na apresentação de experimentos empíricos e sobre o emprego do PC *plugged* ou *unplugged*. Esses experimentos são restritos a um pequeno número de alunos. Outro tema recorrente é o da avaliação ou da falta de avaliação da formação de professores. Essa formação envolve um determinado livro ou jogo, em particular, e normalmente é feita através de um MOOC. Ainda, os artigos, em sua maioria abordam o ensino de programação, ou seja, com foco nos conceitos de programação. Esses conceitos são apresentados através de jogos, no PC *unplugged*. No PC *plugged* são utilizadas plataformas de programação que implementam diferentes linguagens. Em alguns *blogs* ou sítios, foram encontradas sugestões do emprego dessa abordagem, vinculada ao currículo do ensino fundamental e médio.

Alguns países possuem em suas escolas um computador por aluno (Austrália, Nova Zelândia e Uruguai, por exemplo). Esse fato não significou grande desenvolvimento para os mesmos, nem no que se refere aos índices de seus alunos nos testes de Pisa, nem em um grande desenvolvimento econômico. De fato, A Organização para a Cooperação e Desenvolvimento Económico – OECD (2015) mostrou que o uso massivo de dispositivos tecnológicos, em sala de aula, não garante uma melhor educação. Por exemplo, Portugal destaca-se pela qualidade no ensino, mesmo compartilhando recursos entre os estudantes. É importante também destacar o caso de Israel, que não dissocia o PC da Computação, na forma como o PC é implementado.

Com base nesse princípio e considerando-se a situação socioeconômica brasileira, a solução *unplugged* pode ser útil, principalmente para as séries iniciais, não necessariamente de forma geral, mas muito provavelmente em determinadas regiões do País, com maior carência de recursos. Isso não significa que o conteúdo da BNCC não seja o mesmo para todos, apenas que a implementação da forma de ensinar possa variar. É importante salientar que atividades *unplugged* não devem ser entendidas como uma solução completa para o ensino de Computação. A abordagem *unplugged* não atende todos os fundamentos da Computação ou não proporciona uma prática plena. Por esse motivo, recomenda-se que seja feito uso dela, se for o caso, apenas para a introdução do PC, para as séries iniciais do ensino fundamental, pelo fato de “o trabalho com tecnologias digitais ter algo especial que não pode ser explorado por outros tipos de atividades” (*National US Research Council e Committee for the Workshop on Computational Thinking*, 2016). No entanto, essa não precisa ser uma política única para todo o País e sim ser utilizada como alternativa, para onde políticas *plugged* não sejam viáveis. Como visto neste

texto, vários países utilizam políticas diferenciadas para diferentes realidades de seu contexto.

O acesso às máquinas pelas crianças possibilita pôr em prática os conceitos abordados no ensino *unplugged*, ampliando os seus horizontes, cria intimidade com uma linguagem de programação e possibilita solucionar problemas ainda mais complexos. Ou seja, o ideal é utilizar as duas metodologias, onde os recursos tecnológicos podem ser compartilhados com um número maior de estudantes, visto que parte deles poderá, em um determinado momento, estar realizando atividades *unplugged* e em outros momentos as atividades *plugged*. Ainda, a geração atual aprende a operar um tablete ou um celular, mesmo antes de saber falar ou escrever. A escola privar essas crianças da utilização de tecnologia é uma decisão difícil de ser tomada. Sabe-se que, mesmo em regiões carentes do Brasil, as pessoas possuem um celular (basta ver que o número de aparelhos ativos corresponde ao dobro do número de habitantes).

A capacitação de professores também é um tema que precisa ser equacionado. Países, como os Estados Unidos da América e a Alemanha, encontraram, nas aulas de Matemática, a solução quanto à falta de professores para lecionar a disciplina de Computação.

Várias pesquisas, que visam a integração do PC no currículo e aperfeiçoar sua metodologia de ensino, focam na formação de professores e na forma como a disciplina de Computação/Informática, em geral pode vir a ser integrada no currículo. Muitos desses estudos e experiências de outros países apontam para a oferta de disciplinas voltadas para a Computação/Informática, de forma opcional, possibilitando o acesso dos estudantes aos conceitos do PC, para que, futuramente, essas disciplinas façam parte do currículo obrigatório. Em outros casos, o uso do PC se dá de maneira totalmente inter e transdisciplinar, ou seja, embutido dentro de atividades de outras áreas do conhecimento, que já fazem parte do currículo dos alunos.

Por outro lado, se olhar-se o PC propriamente dito, como uma opção para a introdução do ensino da Informática (e não de Computação), não faltarão argumentos, vindos do mercado de trabalho de alguns países, para justificá-la. Segundo várias empresas de tecnologia, há uma demanda de mão de obra qualificada com conhecimentos em programação. Em uma carta aberta, assinada pelos funcionários do alto escalão da Microsoft, Google e Facebook, consta a seguinte frase: “Nós precisamos de talentos, precisamos disso agora, e nós simplesmente não conseguimos achar o suficiente” (Cityofny, 2015). Ainda, de acordo com Guzdial (2016), o Escritório de Dados Estatísticos Trabalhista dos EUA estimou que o desenvolvimento de *software* é uma das categorias com maior crescimento, e um alto percentual, aproximadamente 50%, de profissões, relacionadas à Ciência (principalmente Genética, Biologia), Tecnologia, Engenharia e Matemática, no futuro necessitarão de pessoas com conhecimento em Computação. Essa carência poderá ser sanada se a programação for introduzida nas escolas. Ou seja, essa é a visão predominante nos países que optaram pela adoção do PC.

As pesquisas realizadas por Guzdial (Guzdial *et al*, 2012) apontam para o aumento da motivação para seguir carreira na área da Computação/Informática, em estudantes que tiveram aulas de Computação/Informática no ensino médio. Outro levantamento do Ministério do Trabalho dos EUA (Bureau of Labor Statistics, 2015) prevê um crescimento de 12%, de 2014 a 2024, na área da TICs. Um percentual maior do que a média para todas as demais ocupações. A agência também projeta que serão abertas em torno de 488.500 vagas de trabalho, passando de 3.9 milhões para 4.4 milhões de empregos, entre os anos de 2014 e 2024.

O caso do Brasil em particular, tudo indica, não manifesta essa demanda reprimida por programadores no presente e no futuro próximo, apontando para um adequado dimensionamento dos Cursos de Graduação de Computação (nas suas diferentes modalidades) no País, o que permite definir uma política nacional centrada na qualidade da solução. Também, deve-se destacar que uma das modalidades dos cursos de graduação em Computação é o de Licenciatura em Computação, formando professores com domínio adequado de PC para o Ensino Básico os quais, no momento, não são adequadamente aproveitados. Assim poder dar um tratamento Computacional (e não baseado em Informática) ao PC, bem como o fato de existirem professores adequadamente formados são variáveis que devem ser consideradas para a introdução do PC na BNCC, via disciplinas específicas de Computação, como sugerido por Nunes (Nunes 2017) e não apenas via abordagem transversal em outras áreas.

Quando abordado de forma transversal a outras áreas, o PC pode ser incluído com vantagens no ensino da Matemática, das Ciências, das Línguas e das Artes, por exemplo. A utilização de conceitos computacionais pode facilitar o trabalho de outras áreas, onde princípios da solução de problemas, utilizados na Computação, podem ser utilizados para a solução de problemas típicos dessas áreas. Wing (2010) apresenta essa ideia de outra forma, ou seja, que a utilização de métodos da Computação também pode ajudar na melhoria de processos da nossa vida diária (saber identificar informações importantes em meio a grande quantidade de dados, saber encontrar a informação necessária, etc.). Além disso, com um modelo estruturado de pensamento, é possível auxiliar o processo de aprender a aprender, da mesma forma que ocorre com o aprendizado da leitura: aprendemos a ler para que possamos ler para aprender.

Ainda existem outras correntes para o *software* educacional, como, por exemplo, à aprendizagem ativa (*active learning*) e suas implementações, como acontece na proposta “sala de aula invertida” e nos *Fab labs*. Essa mudança de paradigma congrega tecnologia e metodologias de ensino-aprendizagem.

Por fim, vive-se neste século no mundo da tecnologia e da informação. Esse contexto vem impactando cada vez mais o dia a dia dos estudantes, da vida e do trabalho de todos nós. Para que as futuras gerações estejam cada vez mais preparadas para essa realidade, uma das principais iniciativas está na readequação do currículo das escolas, como já vem ocorrendo em diversos países e continua sendo uma tendência mundial de ensino (competências do século XXI).



## 14. Referências Bibliográficas

- ABELSON, H. & SUSSMAN, G. J. *Structure and Interpretation of Computer Programs*. 2nd ed. Cambridge: MIT Press, 1996. Disponível em: <https://web.mit.edu/alexmv/6.037/sicp.pdf> (acesso em: 13/12/2018).
- ACARA (Australian Curriculum Assessment and Reporting Authority). *Australian Curriculum*. 2015. Disponível em: <http://www.australiancurriculum.edu.au/technologies/digital-technologies/curriculum/f-10?layout=1> (acesso em: 5/12/2018).
- ACM (Association for Computing Machinery). *A Model Curriculum for K-12 Computer Science: Final Report of the ACM K-12 Task Force Curriculum Committee*. New York, NY, USA: ACM Press, 2003. Disponível em: <https://dl.acm.org/citation.cfm?id=2593247> (acesso em: 12/12/2018).
- AfDB (African Development Bank); ADB (Asian Development Bank); EBRD (European Bank for Reconstruction and Development); IDB (Inter-American Development Bank). *The Future of Work: Regional Perspectives*, 2018. Disponível em: [https://www.afdb.org/fileadmin/uploads/afdb/Documents/Publications/The-Future-of-Work-regional\\_perspectives.pdf](https://www.afdb.org/fileadmin/uploads/afdb/Documents/Publications/The-Future-of-Work-regional_perspectives.pdf) (acesso em: 15/12/2018).
- AGGARWAL, A.; GARDNER-MCCUNE, C.; TOURETZKY, D. S. *Evaluating the Effect of Using Physical Manipulatives to Foster Computational Thinking in Elementary School*. In: ACM SIGCSE Technical Symposium on Computer Science Education (SIGCSE '17), 48., 2017, Seattle, WA, USA. *Proceedings...* New York, NY, USA: ACM Press, 2017. p.9 – 14. Disponível em: <http://dl.acm.org/citation.cfm?doid=3017680.3017791> (acesso em: 5/12/2018).
- ANDRADE, D.; CARVALHO, T.; SILVEIRA, J.; et al. *Proposta de Atividades para o Desenvolvimento do Pensamento Computacional no Ensino Fundamental*. In: Congresso Brasileiro de Informática na Educação (CBIE 2013), 2., 2013, Campinas, SP/Workshop de Informática na Escola (WIE 2013), 19., 2013, Campinas, SP. *Anais...* p.169-178, 2013. Disponível em: [https://www.researchgate.net/publication/299666357\\_Proposta\\_de\\_Atividades\\_para\\_o\\_Developolvimento\\_do\\_Pensamento\\_Computacional\\_no\\_Ensino\\_Fundamental](https://www.researchgate.net/publication/299666357_Proposta_de_Atividades_para_o_Developolvimento_do_Pensamento_Computacional_no_Ensino_Fundamental) (acesso em: 12/12/2018).
- ARGENTINA. CFE. *Resolución CFE No 263/15*. 2015. Disponível em: [http://www.unterseccionalroca.org.ar/imagenes/documentos/leg/Resolucion%20263-15%20%28Ley%20Nac%29\\_0.pdf](http://www.unterseccionalroca.org.ar/imagenes/documentos/leg/Resolucion%20263-15%20%28Ley%20Nac%29_0.pdf) (acesso em: 16/12/2018).
- BALANSKAT, A.; ENGELHARDT, K. *Computing our future: Computer programming and coding - Priorities, school curricula and initiatives across Europe*. Out. 2015. European Schoolnet. Disponível em: [https://www.researchgate.net/publication/284139559\\_Computing\\_our\\_future\\_Computer\\_programming\\_and\\_coding\\_-](https://www.researchgate.net/publication/284139559_Computing_our_future_Computer_programming_and_coding_-)

- [Priorities school curricula and initiatives across Europe](#) (acesso em: 13/12/2018).
- BARCELOS, T. S.; SILVEIRA, I. F. *Teaching Computational Thinking in initial series - An analysis of the confluence among mathematics and Computer Sciences in elementary education and its implications for higher education*. In: Conferencia Latinoamericana em Informatica (CLEI 2012), 38., 2012, Medellin, Colombia. p.1–8. Anais... Disponível em: <http://ieeexplore.ieee.org/document/6427135/> (acesso em: 05/12/2018).
- BARR, V.; STEPHENSON, C. *Bringing computational thinking to K-12: what is involved and what is the role of the computer science education community?* ACM Inroads, v. 2, n. 1, p. 48, 2011. Disponível em: <https://c.ymcdn.com/sites/www.csteachers.org/resource/resmgr/BarrStephensonInroadsArticle.pdf> (acesso em: 14/12/2018).
- BAVARESCO, S.; SZÜCS, J.; FERNANDES, C.; MENEZES, P. B. *Aplicação de uma Metodologia para o Ensino de Ciência da Computação para Crianças e para Estudantes do Curso de Bacharelado em Ciência da Computação*. 2018. 18 f. Relatório técnico. Programa de Pós-Graduação em Ciência da Computação, Instituto de Informática, Universidade Federal do Rio Grande do Sul. Porto Alegre, RS, Brasil, novembro 2018. Disponível em: [https://sabi.ufrgs.br/F/S1JNUFRTSU8QMQL2R5BJ3RUIADBDV5FD5SCAIKEVI6FXHJ6PLM-05354?func=full-set-set&set\\_number=012331&set\\_entry=000003&format=999](https://sabi.ufrgs.br/F/S1JNUFRTSU8QMQL2R5BJ3RUIADBDV5FD5SCAIKEVI6FXHJ6PLM-05354?func=full-set-set&set_number=012331&set_entry=000003&format=999) (acesso em: 19/12/2018).
- BBC LEARNING, B. *What is computational thinking?* 2015. Disponível em: <http://www.bbc.co.uk/education/guides/zp92mp3/revision> (acesso em: 5/12/2018).
- BELL, T.; WITTEN, I. H.; FELLOWS, M. *Computer Science Unplugged. An enrichment and extension programme for primary - aged students*. 2015. Disponível em: [https://ir.canterbury.ac.nz/bitstream/handle/10092/247/12584508\\_Main.pdf?sequence=1&isAllowed=y](https://ir.canterbury.ac.nz/bitstream/handle/10092/247/12584508_Main.pdf?sequence=1&isAllowed=y) (acesso em: 19/12/2018).
- BELL, T.; WITTEN, I. H.; FELLOWS, M. *Computer Science Unplugged... - Off-line activities and games for all ages (draft)*. 1998. Disponível em: <https://classic.csunplugged.org/wp-content/uploads/2015/01/unplugged-book-v1.pdf> (acesso em: 14/12/2018).
- BENSE, M. *Pequena Estética*. São Paulo: Perspectiva, 2017.
- BERRY, M. *Computing in the national curriculum - A guide for primary teachers*. 2013. Disponível em: <https://www.computingatschool.org.uk/data/uploads/CASPrimaryComputing.pdf> (acesso em: 14/12/2018).
- BEZERRA, L. N. M.; SILVEIRA, I. F. *Licenciatura em Computação no Estado de São Paulo: uma Análise Contextualizada e um Estudo de Caso*. In: Workshop sobre Computação em Educação, 19., 2011, Natal, RN. Anais do XXXI CSBC, 2011.

- Disponível em:  
<https://www.dimap.ufrn.br/csbc2011/anais/eventos/contents/CADERNO.pdf> (acesso em: 12/12/2018).
- BIRD, J.; CALDWELL, H.; MAYNE, P. *Lessons in teaching computing in primary schools. 2nd edition, revised and updated*. London: Learning Matters, 2017.
- BITA, N. *Computers in class “a scandalous waste”*: Sydney Grammar head. Disponível em: <http://www.theaustralian.com.au/national-affairs/education/computers-in-class-a-scandalous-waste-sydney-grammar-head/news-story/b6de07e63157c98db9974cedd6daa503> (acesso em: 5/12/2018).
- BOCCONI, S.; CHIOCCARIELLO, A.; DETTORI, G.; FERRARI, A.; ENGELHARDT, K. *Developing Computational Thinking in Compulsory Education - Implications for policy and practice*. Luxembourg: European Union, 2016. Disponível em: <https://ec.europa.eu/jrc/en/publication/eur-scientific-and-technical-research-reports/developing-computational-thinking-compulsory-education-implications-policy-and-practice> (acesso em: 14/12/2018).
- BOGLIOLO, A. *Coding in your classroom, now!* 1 ed. Firenze: Giunti, 2016.
- BOMBASAR, J.R.; SANTIAGO, R.; MIRANDA, E.M.; RAAB, A.L. *Ferramentas para o Ensino-Aprendizagem do Pensamento Computacional: onde está Alan Turing?* In: Simpósio Brasileiro de Informática na Educação (SBIE 2015), 26., 2015. CBIE-LACLO, 2015. Anais... Disponível em: [https://www.researchgate.net/publication/284174306\\_Ferramentas\\_para\\_o\\_Ensino-Aprendizagem\\_do\\_Pensamento\\_Computacional\\_onde\\_esta\\_Alan\\_Turing](https://www.researchgate.net/publication/284174306_Ferramentas_para_o_Ensino-Aprendizagem_do_Pensamento_Computacional_onde_esta_Alan_Turing) (acesso em: 5/12/2018).
- BOUCINHA, R.; BRACKMANN, C.; BARONE, D. A. C.; CASALI, A. *Construção do Pensamento Computacional através do Desenvolvimento de Games*. Revista Novas Tecnologias na Educação, v. 15, n. 1, 2017. Disponível em: <http://seer.ufrgs.br/index.php/renote/article/view/75146> (acesso em: 14/12/2018).
- BOULAY, B. DU; O'SHEA, T.; MONK, J. *The black box inside the glass box: presenting computing concepts to novices*. *International Journal of Man-Machine Studies*, v. 14, n. 3, p. 237–249, 1981.
- BRACKMANN, C. P. *Desenvolvimento do Pensamento Computacional Através de Atividades Desplugadas na Educação Básica*. 2017. 224 f. Tese (Doutorado em Informática na Educação) - Programa de Pós-Graduação em Informática na Educação, Centro de Estudos Interdisciplinares em Novas Tecnologias da Educação, Universidade Federal do Rio Grande do Sul. Porto Alegre, RS, Brasil, 2017. Disponível em: <https://www.lume.ufrgs.br/handle/10183/172208> (acesso em: 16/12/2018).
- BRACKMANN, C.; BARONE, D.; CASALI, A.; BOUCINHA, R.; MUNOZ-HERNANDEZ, S. *Computational thinking: Panorama of the Americas*. p.1–6, 2016. In: [2016 International Symposium on Computers in Education \(SIIE\)](http://ieeexplore.ieee.org/document/7751839/). Anais... Disponível em: <http://ieeexplore.ieee.org/document/7751839/> (acesso em: 5/12/2018).

- BRACKMANN, C.; ROMÁN-GONZÁLES, M.; ROBLES, G.; et al. *Development of Computational Thinking Skills through Unplugged Activities in Primary School*. In: *Workshop in Primary and Secondary Computing Education (WiPSCE 2017)*, 12., 2017, Nijmegen, The Netherlands. *Proceedings...* Nijmegen, The Netherlands: ACM Press, 2017. p.7. Disponível em: [https://www.researchgate.net/publication/320884563\\_Development\\_of\\_Computational\\_Thinking\\_Skills\\_through\\_Unplugged\\_Activities\\_in\\_Primary\\_School](https://www.researchgate.net/publication/320884563_Development_of_Computational_Thinking_Skills_through_Unplugged_Activities_in_Primary_School) acesso em: 12/12/2018).
- BRASIL. MEC. *Base Nacional Curricular Comum (BNCC)*. Disponível em: <http://basenacionalcomum.mec.gov.br/a-base> (acesso em: 16/12/2018).
- BRENNAN, K. *Creative computing: A design-based introduction to computational*. 2011. Disponível em: <http://scratched.gse.harvard.edu/sites/default/files/curriculumguide-v20110923.pdf> (acesso em: 14/12/2018).
- BRENNAN, K. *Scratch Curriculum Guide Draft*. ScratchEd, nov. 2011. Disponível em: <http://scratched.gse.harvard.edu/resources/scratch-curriculum-guide-draft> (acesso em: 5/12/2018).
- BRINDA, T.; PUHLMANN, H.; SCHULTE, C. *Bridging ICT and CS – Educational Standards for Computer Science in Lower Secondary Education*. New York, NY: ACM, 2009.
- BROOKS, S. *H.R.2536 - 113th Congress (2013-2014): Computer Science Education Act of 2013*. Disponível em: <https://www.congress.gov/bill/113th-congress/house-bill/2536> (acesso em: 14/12/2018).
- BUNCE, S. *100 ideas for primary teachers: computing*. Bloomsbury Educacion, 2015.
- BUNDY, A. *Computational Thinking is Pervasive*. *Journal of Scientific and Practical Computing*, v. 1, p. 67–69, 2007.
- BUREAU OF LABOR STATISTICS. *Computer and Information Technology Occupations*. Disponível em: <https://www.bls.gov/ooh/computer-and-information-technology/home.htm> (acesso em: 19/12/2018).
- BURRETT, M. *Teaching primary computing*. London: Bloomsbury, 2016.
- CALDWELL, H.; SMITH, N. *Teaching computing unplugged in primary schools: exploring primary computing through practical activities away from the computer*. [S. l.]: Learning Matters, 2017. 216 p.
- CASEY, R. *Computer Science Education and Jobs Act of 2013 (2013 - S. 1407)*. Disponível em: <https://www.govtrack.us/congress/bills/113/s1407> (acesso em: 5/12/2018).
- CASEY, R. *S.671 - 114th Congress (2015-2016): Computer Science Education and Jobs Act of 2015*. Disponível em: <https://www.congress.gov/bill/114th-congress/senate-bill/671> (acesso em: 5/12/2018).

- CASPERSEN, M. E.; FISLER, K.; VAHRENHOLD, J.; HERBSTTRITT, M. *Assessing Learning In Introductory Computer Science*. In: Dagstuhl Seminar 16072. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik GmbH, Wadern/Saarbruecken, Germany, 2016. Disponível em: [http://drops.dagstuhl.de/opus/volltexte/2016/5889/pdf/dagrep\\_v006\\_i002\\_p078\\_s16072.pdf](http://drops.dagstuhl.de/opus/volltexte/2016/5889/pdf/dagrep_v006_i002_p078_s16072.pdf) (acesso em: 14/12/2018).
- CHOI, J.; AN, S.; LEE, Y. *Computing Education in Korea - Current Issues and Endeavors*. *ACM Transactions on Computing Education*, v. 15, n. 2, p. 1–22, 2015. Disponível em: <https://dl.acm.org/citation.cfm?id=2716311> (acesso em: 16/12/2018).
- CHRISTENSEN, M. H. *Structural Synthesis using a Context Free Design Grammar*. *Meeting of the American Educational Research Association*, Vancouver, Canada, 2009.
- CITYOFNY. *An Open Letter from the Nation's Tech and Business Leaders*. 2015. Disponível em: <https://medium.com/cityofny/an-open-letter-from-the-nation-s-tech-and-business-leaders-ca9894a4d125> (acesso em: 5/12/2018).
- CLEMENTS, D. H. *Computers in Early Childhood Mathematics*. *Contemporary issues in early childhood*, v. 3, n. 2, 2002. Disponível em: [https://www.researchgate.net/publication/228788529\\_Computers\\_in\\_Early\\_Childhood\\_Mathematics](https://www.researchgate.net/publication/228788529_Computers_in_Early_Childhood_Mathematics) (acesso em: 16/12/2018).
- CODE.ORG. *Instructor Handbook - Code Studio Lesson Plans for Courses One, Two, and Three*. CODE.ORG, 2015.
- CODE.ORG. *Where computer science counts*. Disponível em: <https://code.org/action> (acesso em: 19/12/2018).
- COLLINS, N. *Live Coding Practice*. In: *Conference on New Interfaces for Musical Expression (NIME07)*, 2007, New York, New York. *Proceedings...* Disponível em: <https://composerprogrammer.com/research/livecodingpractice.pdf> (acesso em: 17/12/2018).
- CONSTANTINO, V.; IOANNOU, A. *Development of Computational Thinking Skills through Educational Robotics*. Disponível em: [ceur-ws.org/Vol-2193/paper9.pdf](http://ceur-ws.org/Vol-2193/paper9.pdf) (acesso em: 17/12/2018).
- COSTA, C. *"Ninguém sabe ainda como formar professores inovadores", diz educadora que trabalhou na Apple e com Obama*. Disponível em: <http://www.bbc.com/portuguese/brasil-38536327> (acesso em: 5/12/2018).
- CSIZMADIA, A.; CURZON, P.; DORLING, M.; et al. *Computational thinking - A guide for teachers*. 2015. *Computing At School (CAS)*. Disponível em: <http://community.computingatschool.org.uk/files/6695/original.pdf> (acesso em: 14/12/2018).
- CSIZMADIA, A. *Computings in the curriculum: challenges and strategies from a teacher's perspective*. *Education and Information Technologies*, V.22, 2. Pp 469-



- 495, 2017. DOI 10.1007/s10639-016-9482-0. Disponível em: <https://link.springer.com/article/10.1007/s10639-016-9482-0> (acesso em: 16/12/2018).
- CSTA (Computer Science Teachers Association). *Bugs in the System: Computer Science Teacher Certification in the U.S.* 2013. Disponível em: [www.csteachers.org/resource/resmgr/CSTA\\_BugsInTheSystem.pdf](http://www.csteachers.org/resource/resmgr/CSTA_BugsInTheSystem.pdf) (acesso em: 19/12/2018).
- CSTA (Computer Science Teachers Association). *CSTA K-12 Computer Science Standards. CSTA Standards Task Force.* ACM - Association for Computing Machinery, 2017. Disponível em: <https://www.doe.k12.de.us/cms/lib/DE01922744/Centricity/Domain/176/CSTA%20Computer%20Science%20Standards%20Revised%202017.pdf> (acesso em: 17/12/2018).
- CSTA (Computer Science Teachers Association). *K-12 Computer Science Standards - Revised 2011 - The CSTA Standards Task Force.* ACM - Association for Computing Machinery, 2011. Disponível em: [https://c.ymcdn.com/sites/www.csteachers.org/resource/resmgr/Docs/Standards/CSTA\\_K-12\\_CSS.pdf](https://c.ymcdn.com/sites/www.csteachers.org/resource/resmgr/Docs/Standards/CSTA_K-12_CSS.pdf) (acesso em: 14/12/2018).
- CSTA/ISTE. *Computational Thinking Across the Curriculum.* 2009. Disponível em: <https://c.ymcdn.com/sites/www.csteachers.org/resource/resmgr/CTExamplesTable.pdf> (acesso em: 14/12/2018).
- CSTA/ISTE. *Computational Thinking: leadership toolkit.* 2011. Disponível em: <https://c.ymcdn.com/sites/www.csteachers.org/resource/resmgr/471.11CTLeadershipToolkit-S.pdf> (acesso em: 14/12/2018).
- CSTA/ISTE. *Operational Definition of Computational Thinking for K–12 Education.* Disponível em: <http://www.iste.org/docs/ct-documents/computational-thinking-operational-definition-flyer.pdf> (acesso em: 13/12/2018).
- CUNY, J. *Computer Science Education for Everyone: A Groundswell of Support.* 2017. Disponível em: <http://www.infosys.org/infosys-foundation-usa/media/blog/Pages/groundswell-support.aspx> (acesso em: 12/12/2018).
- CURZON, P. *Cs4fn and Computational Thinking Unplugged.* In: *Workshop in Primary and Secondary Computing Education (WIPSCE 2013)*, 8., 2013, Aarhus, Denmark. p.47–50. *Proceedings...* New York, NY, USA: ACM Press, 2013. Disponível em: <http://dl.acm.org/citation.cfm?doid=2532748.2611263> (acesso em: 14/12/2018).
- CURZON, P.; MCOWAN, P. W.; PLANT, N.; MEAGHER, L. R. *Introducing teachers to computational thinking using unplugged storytelling.* In: *Workshop in Primary and Secondary Computing Education*, 9., 2014, Berlin, Germany. *Proceedings...* New York, NY, USA: ACM Press, 2014. p.89–92. Disponível em: <http://dl.acm.org/citation.cfm?doid=2670757.2670767> (acesso em: 12/2/2018).



- DARLING-HAMMOND, L.; BRANSFORD, J., Eds. *Preparing Teachers for a Changing World: What Teachers Should Learn and Be Able to Do*. San Francisco, CA: Jossey-Bass, 2005.
- DAVIS, J. *Australia Forgets that Code is Cultural: Replaces History and Geography with Computer Science*. 8. out. 2015. Disponível em: <https://thesocietypages.org/cyborgology/2015/10/08/australia-forgets-that-code-is-cultural-replaces-history-and-geography-with-computer-science/> (acesso em: 12/12/2018).
- DE SMEDT, T.; LECHAT, L.; DAELEMANS, W. *Generative Art Inspired by Nature, Using NodeBox*. In: *Lecture Notes in Computer Science*, 2011, Volume 6625, p.264-272. Berlin/Heidelberg: Springer. Disponível em: [https://www.researchgate.net/publication/220866838\\_Generative\\_Art\\_Inspired\\_by\\_Nature\\_Using\\_NodeBox](https://www.researchgate.net/publication/220866838_Generative_Art_Inspired_by_Nature_Using_NodeBox) (acesso em: 13/12/2018).
- DENNING, P. J. *The profession of IT Beyond computational thinking*. *Communications of the ACM*, v. 52, n. 6, p. 28, 2009.
- DERUY, E. *In Finland, Kids Learn Computer Science Without Computers*. *The Atlantic*. 24 fev. 2017. Disponível em: <https://www.theatlantic.com/education/archive/2017/02/teaching-computer-science-without-computers/517548/> (acesso em: 12/12/2018).
- DESJARDINS, J. *The Largest Companies by Market Cap Over 15 Years*. Disponível em: <http://www.visualcapitalist.com/chart-largest-companies-market-cap-15-years/> (acesso em: 12/12/2018).
- DICKINS, R.; NIELSEN, S. *Computers and coding: with over 100 flaps to lift*. London: Usborne Publishing, 2015.
- DORAN, K.; BOYCE, A.; FINKELSTEIN, S.; BARNES, T. *Outreach for improved student performance: a game design and development curriculum*. In: *ACM annual conference on Innovation and technology in computer science education*, 17., 2012. *Proceedings...* New York, NY, USA: ACM Press, 2012. p.209-214. Disponível em: <http://dl.acm.org/citation.cfm?doid=2325296.2325348> (acesso em: 12/12/2018).
- DOWNEY, A. *Programming as a Way of Thinking*. *Scientific American*, 26. abr. 2017. Disponível em: [https://blogs.scientificamerican.com/guest-blog/programming-as-a-way-of-thinking/?WT.mc\\_id=SA\\_WR\\_20170503](https://blogs.scientificamerican.com/guest-blog/programming-as-a-way-of-thinking/?WT.mc_id=SA_WR_20170503) (acesso em: 12/12/2018).
- EDUCATION SCOTLAND (Foghlam Alba). *Technologies: Experiences and outcomes*. Disponível em: [https://education.gov.scot/scottish-education-system/policy-for-scottish-education/policy-drivers/cfe-\(building-from-the-statement-appendix-incl-btc1-5\)/Experiences%20and%20outcomes](https://education.gov.scot/scottish-education-system/policy-for-scottish-education/policy-drivers/cfe-(building-from-the-statement-appendix-incl-btc1-5)/Experiences%20and%20outcomes) (acesso em: 16/12/2018).
- EL-HANI, C.; MORTIMER, E. *Multicultural education, pragmatism, and the goals of Science teaching*, *Cultural Studies of Science Education* 2 (3), 657-702, July 2007.

- ELLIS, J.; CHARLOTTE, M. *Giggle Chips*. 2017. Disponível em: <http://gigglechips.bigcartel.com/> (acesso em: 12/12/2018).
- EMEC. *e-MEC de Instituições e Cursos de Educação Superior*. Disponível em: <http://emec.mec.gov.br/> (acesso em: 12/12/2018).
- ENGELBERG, M.; THINKFUN. *Chocolate Fix*. 2012. Thinkfun. Disponível em: <http://www.thinkfun.com/products/chocolate-fix/> (acesso em: 12/12/2018).
- ENGELBERG, M.; THINKFUN. *Code Master Programming Logic Game*. 2015. Thinkfun. Disponível em: <http://www.thinkfun.com/products/code-master/> (acesso em: 12/12/2018).
- EPA. *Einheitliche Prüfungsanforderungen – Informatik*. 2004. Disponível em: [http://www.kmk.org/fileadmin/veroeffentlichungen\\_beschluesse/1989/1989\\_12\\_01-EPA-Informatik.pdf](http://www.kmk.org/fileadmin/veroeffentlichungen_beschluesse/1989/1989_12_01-EPA-Informatik.pdf) (acesso em: 12/12/2018).
- FABER, H.; WIERDSMA, M.; DOORNBOS, R.; VEN, J. S. VAN DER; VETTE, K. DE. *Teaching Computational Thinking to Primary School Students via Unplugged Programming Lessons*. *Journal of the European Teacher Education Network*, v. 12, 2017.
- FARAG, B. *Please don't learn to code*. 2016. Disponível em: <https://techcrunch.com/2016/05/10/please-dont-learn-to-code/> (acesso em: 12/12/2018).
- FARIAS, A.; ANDRADE, W.; ALENCAR, R. *Pensamento Computacional em Sala de Aula: Desafios, Possibilidades e a Formação Docente*. 2015. p.1226. Disponível em: <http://br-ie.org/pub/index.php/wcbie/article/view/6262> (acesso em: 12/12/2018).
- FEASTER, Y.; SEGARS, L.; WAHBA, S. K.; HALLSTROM, J. O. *Teaching CS unplugged in the high school (with limited success)*. 2011. p. 248. ACM Press. Disponível em: <http://portal.acm.org/citation.cfm?doid=1999747.1999817> (acesso em: 12/12/2018).
- FERNANDES, C. S. *Ciência da computação para crianças*. 2002. 156 p., il. Dissertação (mestrado) - Universidade Federal do Rio Grande do Sul. Instituto de Informática. Programa de Pós-Graduação em Computação, Porto Alegre, RS-BR, 2002. Orient.: Menezes, Paulo Fernando Blauth. Disponível em: <https://lume.ufrgs.br/handle/10183/6843> (acesso em: 16/12/2018).
- FERNANDES, C. S.; MENEZES, P. B. *Computer science for children*. In: *Society for Information Technology & Teacher Education International Conference, 12.*, 2001 Orlando, Florida. *Proceedings...* Norfolk, Virginia: AACE, 2001. p. 3163-3164.
- FERNANDES, C. S.; MENEZES, P. B. *Metodologia do ensino de ciência da computação : uma proposta para crianças*. In: *Congresso da Sociedade Brasileira de Computação, 21.*, 2001, Fortaleza, CE. *Anais...* Fortaleza: SBC, 2001. p. 89.

- FLEURY, A.; NEVEUX, C. Hamon : *Le code informatique à l'école dès septembre*. 2014. Disponível em: <http://www.lejdd.fr/Societe/Hamon-Le-code-informatiqu-a-l-ecole-des-septembre-675912> (acesso em: 12/12/2018).
- FRANÇA, R. S. DE; SILVA, W. C. DA; AMARAL, H. J. C. DO. *Despertando o interesse pela ciência da computação: Práticas na educação básica*. In: *International Conference on Engineering and Computer Education*, 8., 2013, Luanda, Angola. Anais... p.282–286.
- FRANKLIN, D.; SKIFSTAD, G.; ROLOCK, R. et al. *Using Upper-Elementary Student Performance to Understand Conceptual Sequencing in a Blocks-based Curriculum*. In: *ACM SIGCSE Technical Symposium on Computer Science Education*, 48., 2017, Seattle, Washington, USA. *Proceedings...* New York, NY, USA: ACM Press, 2017. p.231– 236. Disponível em: <http://dl.acm.org/citation.cfm?doid=3017680.3017760> (acesso em: 12/12/2018).
- FURBER, S. *Shut down or restart? - The way forward for computing in UK schools*. London, England: The Royal Society, 2012.
- GAL-EZER. *How to establish a computer science teacher preparation program at your university? the ECSTPP workshop*. *Magazine ACM Inroads*, v. 1, n. 1, p. 35-39, March 2010. New York, NY, USA: ACM.
- GAL-EZER, J.; BEERI, C.; HAREL, D.; YEHUDAI, A. *A high-school program in Computer Science*. *Computer*, 28(10), oct 1995. pp. 73-80.
- GAL-EZER, J.; HAREL, D. *Curriculum for a high school computer science curriculum*. *Computer Science Education* 9(2), pp. 114-147. 1999.
- GALLUP-GOOGLE. *Searching for Computer Science: Access and Barriers in U.S. K-12 Education*. 2015. Disponível em: [https://services.google.com/fh/files/misc/searching-for-computer-science\\_report.pdf](https://services.google.com/fh/files/misc/searching-for-computer-science_report.pdf) (acesso em: 14/12/2018).
- GAMES4LEARNING. *Bits & Bites*. 2014. Imago. Disponível em: <http://bitsandbytes.cards/> (acesso em: 12/12/2018).
- GARFIELD, R.; HASBRO GAMING. *Robot Rally*. 2015. Hasbro. Disponível em: <https://www.hasbro.com/en-us/product/richard-garfield-s-robo-rally-avalon-hill-game:6A791573-5056-9047-F529-BF676C121371> (acesso em: 12/12/2018).
- GOOGLE FOR EDUCATION. *What is Computational Thinking? Computational Thinking for Educators*, 2015. Disponível em: <https://computationalthinkingcourse.withgoogle.com/unit?lesson=8&unit=1> (acesso em: 13/12/2018).
- GRAY, J.; ABELSON, H.; WOLBER D.; FRIEND, M. *Teaching cs principles with app inventor*. In: *Annual Southeast Regional Conference (ACM-SE '12)*, 50., 2012, Tuscaloosa, Alabama, USA. *Proceedings...* New York, NY, USA: ACM Press, 2012. p. 405–406.

- GREENHILL, A.; SLEE, N. *Littlecodr*. 2015. *Littlecodr Games Inc*. Disponível em: <http://littlecodr.com/> (acesso em: 13/12/2018).
- GROVER, S.; BASU, S. *Measuring Student Learning in Introductory Block-Based Programming: Examining Misconceptions of Loops, Variables, and Boolean Logic*. In: *ACM SIGCSE Technical Symposium on Computer Science Education*, 48., Seattle, Washington, US. p.267– 272. 2017. *Proceedings...* New York, NY, USA: ACM Press, 2017. Disponível em: <http://dl.acm.org/citation.cfm?doid=3017680.3017723> (cesso em: 13/12/2018).
- GROVER, S.; PEA, R. *Computational Thinking in K-12: A Review of the State of the Field*. *Educational Researcher*, v. 42, n. 1, p. 38–43, 2013.
- GUZDIAL, M. *Learner-Centered Design of Computing Education - Research on Computing for Everyone*. Morgan & Claypool, 2016.
- GUZDIAL, M. *Why the U.S. is not ready for mandatory CS education*. *Communications of the ACM*, v. 57, n. 8, p. 8–9, 2014.
- GUZDIAL, M.; ERICSON, B. J.; MCKLIN, T.; ENGELMAN, S. *A statewide survey on computing education pathways and influences: factors in broadening participation in computing*. In: *Annual International Conference on International Computing Education Research*, 9., 2012, Auckland, New Zealand. p.143-150. *Proceedings...* New York, NY, USA: ACM Press, 2012. Disponível em: <http://dl.acm.org/citation.cfm?doid=2361276.2361304> (acesso em: 13/12/2018).
- HAZZAN, O.; GAL-EZER, J.; BLUM L. *A model for high school computer science education: the four key elements that make it!* In: *Technical Symposium on Computer Science Education (SIGCSE '08)*, 39., Portland, OR, USA, 2008. *Proceedings...* New York, NY, USA: ACM Press, 2008. p. 281-285.
- HIGGINS, J. PT.; GREEN, S. (Editores). *Cochrane Handbook for Systematic Reviews of Interventions*. Version 5.1.0 (atualizada em março de 2011). *The Cochrane Collaboration*, 2011. Disponível em: <https://training.cochrane.org/handbook> (acesso em: 23/12/2018).
- HOOVER, L.; THINKFUN. *Laser Maze*. 2012. Thinkfun. Disponível em: <http://www.thinkfun.com/products/laser-maze/> (acesso em: 13/12/2018).
- ISAÍAS, M. *Las maestras de primaria invitadas a formarse en programación*. *Jornal "La Capital"*, 24. jun. 2017, Argentina. Disponível em: <http://www.lacapital.com.ar/educacion/las-maestras-primaria-invitas-formarse-programacion-n1421812.html> (acesso em: 13/12/2018).
- ISTE. *ISTE Standards for Students*. Disponível em: <http://www.iste.org/standards/standards/for-students#startstandards> (acesso em: 13/12/2018).
- JOHNSON, P. *France to offer programming in elementary school*. Disponível em: <http://www.itworld.com/article/2696639/application-management/france-to-offer-programming-in-elementary-school.html> (acesso em: 13/12/2018).

- JONES, S. P. *International comparisons. Version 5. Computing at School*, November 2011. Disponível em:  
<https://www.google.com/url?sa=t&rct=j&q=&esrc=s&source=web&cd=2&ved=2ahUK EwiZjsue76TfAhWDEpAKHS3wAPYQFjABegQICBAC&url=https%3A%2F%2Fcommunity.computingatschool.org.uk%2Ffiles%2F6710%2Foriginal.pdf&usg=AOvVaw1EjpeYWHkQGUisnSI3xlgU> (acesso em: 16/12/2018).
- KALELIOGLU, F.; GÜLBAHAR, Y.; KUKUL, V. *A Framework for Computational Thinking Based on a Systematic Research Review*. *Baltic Journal of Modern Computing*, v. 4, n. 3, p. 583–596, 2016.
- KITCHENHAM, B.; BRERETON, O. PEARL; BUDGEN, D. *The value of mapping studies – A participant-observer case study*. In: *Evaluation and Assessment in Software Engineering* (EASE 2010), 14., Keele University, UK, 2010. *Proceedings...* Keele, UK, 2010.
- KNOTT, R. *Fibonacci Numbers and Nature*. Disponível em:  
<http://www.maths.surrey.ac.uk/hosted-sites/R.Knott/Fibonacci/fibnat.html> (acesso em: 13/12/2018).
- KOLOGESKI, A.; SILVA, C.; BARBOSA, D.; MATTOS, R.; MIORELLI, S. *Desenvolvendo o Raciocínio Lógico e o Pensamento Computacional: Experiências no Contexto do Projeto Logicando*. *Revista Novas Tecnologias na Educação (RENTE)*, v. 14, n. 2, 2016.
- KOODIAAPINEN. *Teacher's Guide to encode the school*. Disponível em:  
<http://koodiaapinen.fi/en/> (acesso em: 04/03/2018).
- KURSHAN, B. *Thawing from a Long Winter in Computer Science Education*. *Forbes*, p. 2, fev. 2016.
- LAMBERT, L.; GUIFFRE, H. *Computer science outreach in an elementary school*. *Journal of Computing Sciences in Colleges*, v. 24, n. 3, 2009.
- LIUKAS, L. *Hello Ruby: adventures in coding*. Feiwei & Friends, 2015.
- LOEWUS, L. H. *President Obama Announces 'Computer Science for All' Initiative*. *Education Week*, January 30, 2016. Disponível em:  
[http://blogs.edweek.org/edweek/curriculum/2016/01/white\\_house\\_computer\\_science\\_for\\_all\\_initiative.html](http://blogs.edweek.org/edweek/curriculum/2016/01/white_house_computer_science_for_all_initiative.html) (acesso em: 16/12/2018).
- 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.
- HEINTZ, F.; MANNILA, L.; FÄRNQVIST, T. *A Review of Models for Introducing Computational Thinking, Computer Science and Computing in K–12 Education*. In: *IEEE Frontiers in Education Conference (FIE)*, 46., 2016, Erie, PA, USA. *Proceedings...* DOI:10.1109/FIE.2016.7757410 Disponível em:  
<https://pdfs.semanticscholar.org/7baa/f05a405844af17446eea4159d0dcc055c75f.pdf> (acesso em: 14/12/2018).



- MANSEL, T. *La receta de Estonia para convertirse en una potencia tecnológica*. 15 maio 2013. Disponível em: [http://www.bbc.com/mundo/noticias/2013/05/130515\\_tecnologia\\_cuna\\_ninos\\_informatica\\_estonia\\_aa](http://www.bbc.com/mundo/noticias/2013/05/130515_tecnologia_cuna_ninos_informatica_estonia_aa) (acesso em: 13/12/2018).
- MATHANDCODING. *Coding Farmers*. 2015. MathAndCoding. Disponível em: <http://codingfarmers.net/> (acesso em: 13/12/2108).
- MATHANDCODING. *CodingIsGood*. 2017. MathAndCoding. Disponível em: <http://www.mathandcoding.org/> (acesso em: 13/12/2018).
- MCLUHAN, M. *Understanding media: the extensions of man*. 1st MIT Press ed. Cambridge, Mass: MIT Press, 1994.
- MCMMASTER, K.; RAGUE, B.; ANDERSON, N. *Integrating Mathematical Thinking, Abstract Thinking, and Computational Thinking*. In: ASEE/IEEE Frontiers in Education Conference, 40., out. 2010, Washington, DC, USA. *Proceedings...* Disponível em: [https://www.researchgate.net/publication/224206968\\_Integrating\\_Mathematical\\_Thinking\\_Abstract\\_Thinking\\_and\\_Computational\\_Thinking](https://www.researchgate.net/publication/224206968_Integrating_Mathematical_Thinking_Abstract_Thinking_and_Computational_Thinking) (acesso em: 05/02/2018).
- MEC. *Base Nacional Comum Curricular - Estudo Comparativo entre a Versão 2 e a Versão Final*. 12. abr. 2017. Disponível em: [http://cnebncc.mec.gov.br/docs/BNCC\\_Estudo\\_Comparativo.pdf](http://cnebncc.mec.gov.br/docs/BNCC_Estudo_Comparativo.pdf) (acesso em: 19/12/2018).
- MELLO, P. M.; MELLO, D. A. *Jogos Boole*. Disponível em: <http://www.jogosboole.com.br/> (acesso em: 13/12/2018).
- MENEZES, P. B.; SERNADAS, A.; COSTA, J. *Nonsequential automata semantics for a concurrent, object-based language*. *Electronic notes in theoretical computer science*, Amsterdam, v. 14, n. 1, p. 245-273, (1998), il.
- MORENO-LEÓN, J.; ROBLES, G.; ROMÁN-GONZÁLES, M. *Programar para aprender en Educación Primaria y Secundaria: ¿qué indica la evidencia empírica sobre este enfoque?* *Revista de Investigación en Docencia Universitaria de la Informática*, v. 10, n. 2, p. 45– 51, 2017.
- MUFFET, T. *Computer coding taught in Estonian primary schools*. 8. jan. 2014. Estonia. Disponível em: <http://www.bbc.com/news/av/education-25648769/computer-coding-taught-in-estonian-primary-schools> (acesso em: 13/12/2018).
- MYKKÄNEN, J.; LIUKAS, L. *Koodi 2016*. 1o ed. Finlândia: Lönnberg Print, 2014. Disponível em: [https://s3-eu-west-1.amazonaws.com/koodi2016/Koodi2016\\_LR.pdf](https://s3-eu-west-1.amazonaws.com/koodi2016/Koodi2016_LR.pdf) (acesso em: 14/12/2018).
- NATIONAL RESEARCH COUNCIL (U.S.) (ORG.). *Report of a workshop of pedagogical aspects of computational thinking*. Washington D.C: National Academies Press, 2011.



- NATIONAL RESEARCH COUNCIL (U.S.); COMMITTEE FOR THE WORKSHOPS ON COMPUTATIONAL THINKING. *Report of a workshop on the scope and nature of computational thinking*. Washington D.C.: National Academies Press, 2010.
- NAVAUX, P. O. A.; CÁCERES, E. N.; ZORZO, A. F. *Documento de Área: Ciência da Computação*. CAPES, 2016.
- NISHIDA, T.; KANEMUNE, S.; IDOSAKA, Y.; et al. A CS Unplugged Design Pattern. In: *ACM technical symposium on Computer science education*, 40., 2009, Chattanooga, TN, USA. *Proceedings...* New York, NY: ACM, 2009. p.231–235. Disponível em: <https://dl.acm.org/citation.cfm?id=1508951> (acesso em: 13/12/2018).
- NOSCHANG, L. F.; PELZ, F.; JESUS, E.; RAABE, A. *Portugol Studio: Uma IDE para Iniciantes em Programação*. 2014. Brasília, DF.
- NUNES, D. J. *Ciência da Computação na Educação Básica*. ADUFRGS - Sindical, 6. jun. 2011. Disponível em: <http://www.adufrgs.org.br/artigos/ciencia-da-computacao-na-educacao-basica/> (acesso em: 13/12/2018).
- NUNES, D. J. *Ciência da Computação na Educação Básica*. Jornal da Ciência, 09 de setembro, 2011. Disponível em: <http://jcnoticias.jornaldaciencia.org.br/12-ciencia-da-computacao-na-educacao-basica/#> (acesso em 24/12/2018).
- NUNES, D. J. *Licenciatura em Computação*. Jornal da Ciência, 09 de maio, 2008. Disponível em: <http://jcnoticias.jornaldaciencia.org.br/22-licenciatura-em-computacao-artigo-de-daltro-jose-nunes/#> (acesso em 24/12/2018).
- OECD. *PISA (Programme for International Student Assessment)*. Disponível em: <http://www.oecd.org/pisa/> (acesso em: 13/12/2018).
- OECD. *Students Computers and Learning*. OECD, 2015.
- OECD. *The Future of Education and Skills – The future we want*. OECD, 2018.
- PAPERT, S. *Mindstorms: Children, Computers, And Powerful Ideas*. Basic Books, 1980.
- PAPERT, S.; HAREL, I. *Constructionism: research reports and essays*. 1985-1990. Norwood, N.J: Ablex Pub. Corp, 1991.
- PAPERT, S.; SOLOMON, C. *Twenty things to do with a Computer*. *Educational Technology Magazine*, 1972. Disponível em: <http://www.stager.org/articles/twentythings.pdf> (acesso em: 14/12/2018).
- PAPERT, S. *Mindstorms: Children, computers, and powerful ideas*, Basic Books, Inc. New York, USA, 1980.
- PAPPANO, L. *Learning to Think Like a Computer*. *New York Times*, 4. abr. 2017. Disponível em: <https://www.nytimes.com/2017/04/04/education/edlife/teaching-students-computer-code.html> (acesso em: 13/12/2018).

- PATTERSON, S. *Programming in the primary grades: beyond the hour of code*. Lanham, Maryland: Rowman & Littlefield, 2016.
- PELGRUM, W. J. *Obstacles to the integration of ICT in education: results from a worldwide educational assessment*. *Computers & Education*, v. 37, 2001. p.163–178.
- PERRENOUD, P. *10 Novas Competências para Ensinar*. Unicef, 2018.
- PIAGET, J. *Morphisms and Categories: Comparing and transforming*. Laurence Erlbaum Associates, Inc., New Jersey, 1992.
- PIAGET, Jean. *O Raciocínio na Criança*. Trad. Valerie Rumjanek Chaves. Rio de Janeiro: Record, 1967. 241p.
- PISA. *Pisa 2015 Collaborative Problem-Solving Framework*. Abr. 2017. Disponível em: <https://www.oecd.org/pisa/pisaproducts/Draft%20PISA%202015%20Collaborative%20Problem%20Solving%20Framework%20.pdf> (acesso em: 13/12/2018).
- RAMOS, J. L.; ESPADEIRO, R. G. *Os Futuros Professores e os professores do futuro. Os desafios da introdução ao Pensamento Computacional na escola, no currículo e na aprendizagem*. Revista Educação, Formação & Tecnologias, v. 7, p. 4–25, 2014.
- RIBEIRO, L.; NUNES, D. J.; CRUZ, M. K. DA; MATOS, E. DE S. *Computational Thinking: Possibilities and Challenges*. In: *Workshop-School on Theoretical Computer Science*, 2.2013, Rio Grande, RS. Anais... 2013.
- RICHARD, L. *A Cognitive Model of the Antecedents and Consequences of Satisfaction Decisions*. *Journal of Marketing Research*, v. 17, n. 4 (Nov., 1980), pp. 460-469. American Marketing Association. Disponível em: <http://www.jstor.org/stable/3150499> (acesso em: 13/12/2018).
- RILEY, D. D.; HUNT, K. A. *Computational thinking for the modern problem solver*. Boca Raton, FL: CRC Press, 2014.
- RODRIGUES, F.; BRACKMANN, C. P.; BARONE, D. A. C. *Estudo da Evasão no Curso de Ciência da Computação da UFRGS*. Revista Brasileira de Informática na Educação, v. 23, n. 01, p. 97, 2015.
- RODRIGUEZ, B.; RADER, C.; CAMP, T. *Using Student Performance to Assess CS Unplugged Activities in a Classroom Environment*. In: *ACM Conference on Innovation and Technology in Computer Science Education*, 21., 2016, Arequipa, Peru. *Proceedings...* ACM New York, NY, USA: ACM Press, 2016. p.95–100. Disponível em: <http://dl.acm.org/citation.cfm?doid=2899415.2899465> (acesso em: 13/12/2018).
- RODRIGUEZ, B.; STEPHEN, K.; RADER, C.; CAMP, T. *Assessing Computational Thinking in CS Unplugged Activities*. *ACM SIGCSE Technical Symposium on*

- Computer Science Education*, 48., 2017, Seattle, Washington, USA. *Proceedings...* New York, NY, USA: ACM Press, 2017. p. 501–50.
- ROMÁN-GONZÁLES, M.; MORENO-LEÓN, J.; ROBLES, G. *Complementary Tools for Computational Thinking Assessment*. In: *International Conference on Computational Thinking Education*, 2017, Hong Kong. Disponível em: [https://www.researchgate.net/publication/318469859\\_Complementary\\_Tools\\_for\\_Computational\\_Thinking\\_Assessment](https://www.researchgate.net/publication/318469859_Complementary_Tools_for_Computational_Thinking_Assessment) (acesso em: 15/12/2018).
- ROMÁN-GONZÁLES, M.; PÉREZ, J. C.; ROBLES, G.; MORENO-LEÓN, J. *Full validation of the Computational Thinking Test (CTt)*. 2017. Disponível em: <https://www.researchgate.net/project/Full-validation-of-the-Computational-Thinking-Test-CTt> (acesso em: 15/12/2018).
- ROMÁN-GONZÁLES, M.; PÉREZ-GONZÁLEZ, J.-C.; JIMÉNEZ-FERNÁNDEZ, C. *Which cognitive abilities underlie computational thinking? Criterion validity of the Computational Thinking Test*. *Computers in Human Behavior*, v. 72, p. 678–691, 2017.
- ROSSI, B. Python overtakes French as the most popular language taught in primary schools. *Information Age*, 27 August 2015. Disponível em: <http://www.information-age.com/it-management/skills-training-and-leadership/123460073/python-overtakes-french-most-popular-language-taught-primary-schools> (acesso em: 13/12/2018).
- Royal Society. *Shut down or restart: The way forward for computing in UK schools*. 2012. Disponível: <https://royalsociety.org/~media/education/computing-in-schools/2012-01-12-computing-in-schools.pdf> (acesso em: 15/12/2018).
- RYAN, P. *A Esmeralda do Rio Negro*. New York, EUA: Ediouro, 1994.
- RYAN, P. *The Black River emerald*. New York: Berkley Books, 1987.
- SADOSKY, F. *Una propuesta para refundar la enseñanza de la computación en las escuelas Argentinas*. Jan. 2013. Disponível em: <http://www.fundacionsadosky.org.ar/wp-content/uploads/2014/06/cc-2016.pdf> (acesso em: 13/12/2018).
- SANKARAN, R. *Burning questions*. US Department of Energy. Abr. 2007. Disponível em: <http://ascr-discovery.science.doe.gov/2007/04/burning-questions> (acesso em: 13/12/2018).
- SBC. *Nota Técnica da Sociedade Brasileira de Computação sobre a BNCC-EF e a BNCC-EM*. 2018b. Disponível em: <http://www.sbc.org.br/institucional-3/cartas-abertas/summary/93-cartas-abertas/1197-nota-tecnica-sobre-a-bncc-ensino-medio-e-fundamental> (acesso em: 13/12/2018).
- SBC. *Diretrizes para ensino de Computação na Educação Básica Ensino de Computação na Educação Básica, documento da Sociedade Brasileira de Computação*. 2018a. Disponível em: <http://www.sbc.org.br/documentos-da-sbc/send/131-curriculos-de-referencia/1177-diretrizes-para-ensino-de-computacao-na-educacao-basica> (acesso em: 13/12/2018).

- SCHIFF, R.; VAKIL, E. *Age differences in cognitive skill learning, retention and transfer: The case of the Tower of Hanoi Puzzle*. *Learning and Individual Differences*, v. 39, p. 164– 171, 2015.
- SCHMUNDT, H. *Should IT Classes Be Required?* Spiegel, 16 maio 2013. Disponível em: <http://www.spiegel.de/international/germany/experts-in-germany-divided-on-computer-science-in-school-curriculum-a-899979.html> (acesso em: 13/12/2018).
- SEITER, L.; FOREMAN, B. *Modeling the learning progressions of computational thinking of primary grade students*. In: *Annual International ACM Conference on International Computing Education Research*, 9., 2013, San Diego, San California, USA. *Proceedings...* New York, NY, USA: ACM Press, 2013. p.59. Disponível em: <http://dl.acm.org/citation.cfm?doid=2493394.2493403> (acesso em: 13/12/2018).
- SENTANCE, S.; CSIZMADIA, A. Teachers' perspectives on successful strategies for teaching Computing in school. In: IFIP TCS 2015. **Anais...** 2015. Disponível em: <http://community.computingschool.org.uk/files/6769/original.pdf>. Acesso em: 13/12/2018.
- SHAPIRO, D.; THINKFUN. *Robot Turtle*. 2014. Disponível em: <http://www.robotturtles.com/> (acesso em: 19/12/2018).
- SIDHU, R. *Code Monkey Island*. 2014. Code Monkey Games, LLC. Disponível em: <https://www.kickstarter.com/projects/rajsidhu/code-monkey-island-making-programming-childs-play> (acesso em: 13/12/2018).
- SOUZA, P. N. P. DE; SILVA, E. B. DA. *Como entender e aplicar a nova LDB: LEI no. 9.394/96*. São Paulo, Brasil: Pioneira, 1997.
- SUNG, W.; AHN, J-H.; BLACK, J. 2017. *Introducing Computational Thinking to Young Learners: Practicing Computational Perspectives Through Embodiment in Mathematics Education*. *Technology, Knowledge and Learning*. 10.1007/s10758-017-9328-x. Disponível em: [https://www.researchgate.net/publication/318754784\\_Introducing\\_Computational\\_Thinking\\_to\\_Young\\_Learners\\_Practicing\\_Computational\\_Perspectives\\_Through\\_Embodiment\\_in\\_Mathematics\\_Education](https://www.researchgate.net/publication/318754784_Introducing_Computational_Thinking_to_Young_Learners_Practicing_Computational_Perspectives_Through_Embodiment_in_Mathematics_Education) (acesso em 04/12/2018).
- TAKAHASHI, M.; AZUMA, S.; TREND-PRO CO., LTD. *Guia Mangá de Banco de Dados*. 1. ed. São Paulo, Brasil: Novatec, 2009.
- TAUB, R.; ARMONI, M.; BEN-ARI, M. *CS Unplugged and Middle-School Students? Views, Attitudes, and Intentions Regarding CS*. *ACM Transactions on Computing Education*, v. 12, n. 2, p. 1–29, 2012.
- TAUB, R.; BEN-ARI, M.; ARMONI, M. *The effect of CS unplugged on middle-school students' views of CS*. p.99, 2009. ACM Press. Disponível em: <http://portal.acm.org/citation.cfm?doid=1562877.1562912> (acesso em: 13/12/2018).
- TAYLOR, R.P. *Introduction*. In: *The Computer in School: Tutor, Tool, Tutee*. Ed. Teachers College Press, New York, 1980, 1–10.

- TEACHING LONDON COMPUTING. *Developing Computational Thinking. Teaching London Computing*, 2014. Disponível em: <http://teachinglondoncomputing.org/resources/developing-computational-thinking> (acesso em: 5/12/2018).
- TEW, A. E.; DORN, B.; LEAHY, W. D.; GUZDIAL, M. *Context as Support for Learning Computer Organization. Journal on Educational Resources in Computing*, v. 8, n. 3, p. 1– 18, 2008.
- THE WHITE HOUSE. *President Obama Announces Computer Science For All Initiative*. Disponível em: <https://obamawhitehouse.archives.gov/the-press-office/2016/01/30/fact-sheet-president-obama-announces-computer-science-all-initiative-0> (acesso em: 7/11/2017).
- TURING, A. M. *On Computable Numbers, with an Application to the Entscheidungsproblem*. 1936. Disponível em: [https://www.cs.virginia.edu/~robins/Turing\\_Paper\\_1936.pdf](https://www.cs.virginia.edu/~robins/Turing_Paper_1936.pdf) (acesso em: 13/12/2018).
- UNNIKRISHNAN, R.; AMRITA, N.; MUIR, A.; RAO, B. *Of Elephants and Nested Loops: How to Introduce Computing to Youth in Rural India*. p.137–146, 2016. ACM Press. Disponível em: [http://www.academia.edu/27465463/Of\\_Elephants\\_and\\_Nested\\_Loops\\_How\\_to\\_Introduce\\_Computing\\_to\\_Youth\\_in\\_Rural\\_India](http://www.academia.edu/27465463/Of_Elephants_and_Nested_Loops_How_to_Introduce_Computing_to_Youth_in_Rural_India) (acesso em: 13/12/2018).
- VALENTE, J. A. *Integração do Pensamento Computacional no Currículo da Educação Básica: Diferentes Estratégias Usadas e Questões de Formação de Professores e Avaliação do Aluno*. Revista e-Curriculum, v. 14, n. 3, 2016.
- VICARI, R.M. *Tendências em Inteligência Artificial na Educação no Período de 2017 a 2030*. UNESCO, 2017.
- VIEIRA, A.; PASSOS, O.; BARRETO, R. *Um Relato de Experiência do Uso da Técnica Computação Desplugada*. In: Workshop sobre Educação em Computação (Wei), 21., 2013, Maceió, AL, Brasil. Anais...SBC, 2013. Disponível em: <http://www.lbd.dcc.ufmg.br/colecoes/wei/2013/0031.pdf>
- WANG, P. S. *From computing to computational thinking*. Boca Raton, FL: CRC Press, Taylor & Francis Group, 2016.
- WING, J. *Computational Thinking with Jeannette Wing*. Columbia Journalism School, 2014.
- WING, J. *Computational Thinking*. 2007. Carnegie Mellon University. Disponível em: [http://www.cs.cmu.edu/afs/cs/usr/wing/www/Computational\\_Thinking.pdf](http://www.cs.cmu.edu/afs/cs/usr/wing/www/Computational_Thinking.pdf) (acesso em: 13/12/2018).
- WING, J. M. *Computational thinking and thinking about computing. Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, v. 366, n. 1881, p. 3717–3725, 2008.

- WING, J. M. *Computational Thinking Benefits Society. Social Issues in Computing*. 2014. Disponível em: <http://socialissues.cs.toronto.edu/2014/01/computational-thinking> (acesso em: 13/12/2018).
- WING, J. M. *Computational Thinking for Everyone*. Apresentação. Disponível em: <http://exploringcs.org/wp-content/uploads/2010/09/Wing08.ppt> (acesso em: 19/12/2018).
- WING, J. M. *Computational thinking. Communications of the ACM*, v. 49, n. 3, p. 33, 2006.
- WING, J. M. *Computational Thinking: What and Why?* 17. out. 2010. Disponível em: <http://www.cs.cmu.edu/~CompThink/resources/TheLinkWing.pdf> (acesso em: 13/12/2018).
- WBG (World Bank Group). *World Development Report 2018 (WDR 2018)*. Disponível em <http://www.worldbank.org/en/publication/wdr2018> (acesso em 23/12/2018).
- YADAV, A. ; KORB, J.T. *Learning to teach computer science. Commun. ACM*, 55, 11 (Nov. 2012), 31.
- YADAV, A. et al. *Computational thinking in elementary and secondary teacher education. ACM Transactions on Computing Education*, 14, 1 (Mar. 2014), 1–16.
- YADAV, A. et al. *Introducing computational thinking in education courses*. In: *ACM Technical Symposium on Computer Science Education*, 42., 2011, Dallas, TX, USA. *Proceedings...* New York, NY, USA: ACM Press, 2011. P. 465–470. Disponível em: [http://cs4edu.cs.purdue.edu/\\_media/sigcse11-final.pdf](http://cs4edu.cs.purdue.edu/_media/sigcse11-final.pdf) (acesso em: 15/12/2018).
- YAKOS, D.; THINKFUN. *Circuit Maze*. 2015. Thinkfun. Disponível em: <http://www.thinkfun.com/products/circuit-maze> (acesso em: 13/12/2018).



## 15. Anexos

### Anexo I - Resultados do Teste de Pisa com Utilização de Tecnologia

O que os dados nos dizem sobre os testes de Pisa com o uso de tecnologia podem ser considerados, ainda que de forma limitada, como os primeiros resultados globais que avaliam o desempenho de alunos, em tarefas similares, com e sem o uso de tecnologia. Esses resultados falam sobre o desempenho de alguns países como, por exemplo, Cingapura, seguida pela Coreia, Hong Kong-China, Japão, Canadá e Xangai-China foram os países/economias com melhor desempenho em leitura digital em 2012; Cingapura e Xangai-China, seguidos pela Coreia, Hong Kong-China, Macau-China, Japão e China Taipei foram os que tiveram melhor desempenho na avaliação de Matemática baseada em computador de 2012. Na Coreia e em Cingapura, os alunos pontuam mais de 20 pontos na escala de leitura digital, em média, do que os estudantes de outros países com habilidades similares em leitura impressa. Estudantes na Austrália, Áustria, Canadá, Japão, Eslovênia e Estados Unidos, bem como estudantes em países/economias parceiros Macao-China e Emirados Árabes Unidos, apresentam melhor desempenho em tarefas matemáticas que exigem o uso de computadores para resolver problemas comparados com o seu sucesso em tarefas tradicionais. Por outro lado, os estudantes na Bélgica, Chile, França, Irlanda, Polônia e Espanha têm um desempenho pior que o esperado em tais tarefas, dado seu desempenho em tarefas matemáticas tradicionais.

O Estudo Internacional sobre Informática e Informação (2013) e sua relação com a leitura digital no PISA

Em 2013, 21 sistemas educacionais em todo o mundo participaram do primeiro Estudo Internacional sobre Informática e Informação (ICILS), organizado pela Associação Internacional para Avaliação de Realização Educacional (IEA). Alfabetização em Informática e Computação é definida como “a capacidade de um indivíduo de usar computadores para investigar, criar e comunicar para participar efetivamente em casa, na escola, no local de trabalho e na sociedade”. A estrutura destaca duas vertentes da competência digital: “coletar e gerenciar informações”, que também envolve a localização e avaliação de informações, e “produzir e trocar informações”, das quais uma compreensão das questões de segurança e proteção *on-line* faz parte. Embora alguns aspectos destacados pela estrutura de leitura digital do PISA sejam cobertos, em particular, pela primeira vertente da estrutura do ICILS, o conceito de letramento computacional e informacional é claramente distinto da leitura digital. O teste foi administrado a alunos da oitava série. Entre os 12 países que atenderam aos requisitos de amostragem para ICILS, a República Tcheca obteve a maior pontuação média, seguida por um grupo de quatro países (Austrália, Coreia, Noruega, da 9ª série, e Polônia) com pontuações médias semelhantes. Embora a população-alvo seja diferente, é notável que o desempenho

médio da Polônia tenha sido claramente superior ao de países, como a Federação Russa, a Eslováquia e a Eslovênia, cujas pontuações médias na avaliação de leitura digital do PISA foram semelhantes (Fraillon *et al.*, 2014).

## **Anexo II - Políticas do MEC mais Relacionadas com o Foco deste Trabalho**

Linux Educacional

Guia de Tecnologias Educacionais

Portal do Professor

Repositórios de Conteúdos Educacionais

Banco Internacional de Objetos Educacionais

Plano Nacional de Formação de Professores da Rede Pública

Universidade Aberta do Brasil

Plataforma Paulo Freire (Plano Nacional de Formação de Professores)

EducaCenso

Plano de Desenvolvimento da Educação

IDEB – Índice de Desenvolvimento da Educação Básica

Computadores na Escola (Laboratórios, UCA e Tablet educacional)

Olimpíadas da Língua Portuguesa

Olimpíadas de Matemática

Programa Nacional de Apoio às Feiras de Ciências da Educação Básica

Programa Educação Inclusiva: Direito à Diversidade

Programa de Formação Continuada de Professores na Educação Especial

Prolind

Programa de Formação em Direitos Humanos

Procampo

Planetários, Observatórios e Portais de Astronomia

Museus do Brasil

## Anexo III – Exemplos de Conteúdos

Para melhor exemplificar o tipo de conteúdos encontrados para a formação de professores (estudantes ou já em atividade) será usado parte do material disponível nos cursos da Google. A seleção e tradução deste conteúdo é livre, realizada pelos autores. Também é apresentado um exemplo de atividade vinculada a um livro em particular (*CT Unplugged*). Esse segundo tipo de sugestão de atividades de apoio aos professores não pode ser caracterizado como um conteúdo de formação, mas pode ser útil no dia a dia da sala de aula.

### Exemplo I - Conteúdos do curso PC/Google

#### *Sobre o Curso*

O objetivo deste curso é ajudar os educadores a aprender sobre o PC, como ele difere da ciência da computação (sic) e como ele pode ser integrado a uma variedade de assuntos. Como participante do curso, você aumentará sua conscientização sobre PC, explorará exemplos de PC integrados em suas áreas de estudo, experimentará exemplos de atividades integradas em PC para suas áreas/disciplinas e criará um plano para integrar a PC em seus próprios currículos.

O curso é dividido em cinco unidades, cada uma com foco nos seguintes temas:

- Introduzindo o pensamento computacional: o que é PC? onde ocorre, por que você deveria se importar e como está sendo aplicado?
- Algoritmos de Exploração - Percorra exemplos de algoritmos usados em sua área de conhecimento. Reconheça por que os algoritmos são ferramentas poderosas para aumentar o que você pode fazer e que a tecnologia pode ser útil para implementar e automatizar algoritmos.
- Encontrar Padrões - Explore exemplos de padrões em vários assuntos e desenvolva seus próprios processos para abordar um problema através do reconhecimento de padrões.
- Algoritmos de desenvolvimento - Aumente sua confiança na aplicação do processo computacional a um determinado problema e reconheça como os algoritmos podem articular um processo ou uma regra.
- Projeto Final: Aplicando o PC - Crie um texto de como o PC se aplica à sua área de conhecimento/disciplina e um plano para integrá-lo ao seu trabalho e sala de aula.

**NOTA:** este curso está organizado de acordo com as políticas para o PC da Austrália e do UK. Essas referências poderão aparecer nos exemplos aqui utilizados.

#### *Estrutura do Curso*

O curso *Computational Thinking for Educators* inclui o seguinte:

- Curso: consiste em cinco unidades, incluindo um projeto final de duas partes.

- Unidades: inclui um grupo de lições; cada unidade contém uma mistura de lições e atividades para quatro grupos diferentes:
  - Professores de Humanidades
  - Professores de Matemática
  - Professores de Ciências
  - Professores de Ciência da Computação
- Atividades das lições: inclui exemplos de simulações, programas e exercícios que aumentam a conscientização sobre a PC, mostram a integração do PC e permitem que você interaja e desenvolva o PC na sua disciplina. As atividades da aula também fornecem instruções passo a passo para realizar as tarefas incluídas em cada atividade, *links* para aprender mais, atividades para praticar as habilidades e obter *feedback*, e uma comunidade de discussão para compartilhar ideias e obter ajuda.
- Projeto Final: oferece uma chance para você aplicar as habilidades aprendidas na aula.

## Certificação

Para você receber o certificado de participação no curso, você precisa completar as duas etapas do projeto final de curso.

## Recursos Adicionais

Este curso é apenas um início. Você pode encontrar mais lições, recursos, artigos científicos e mais em: [Exploring Computational Thinking](#).

## Comunidade

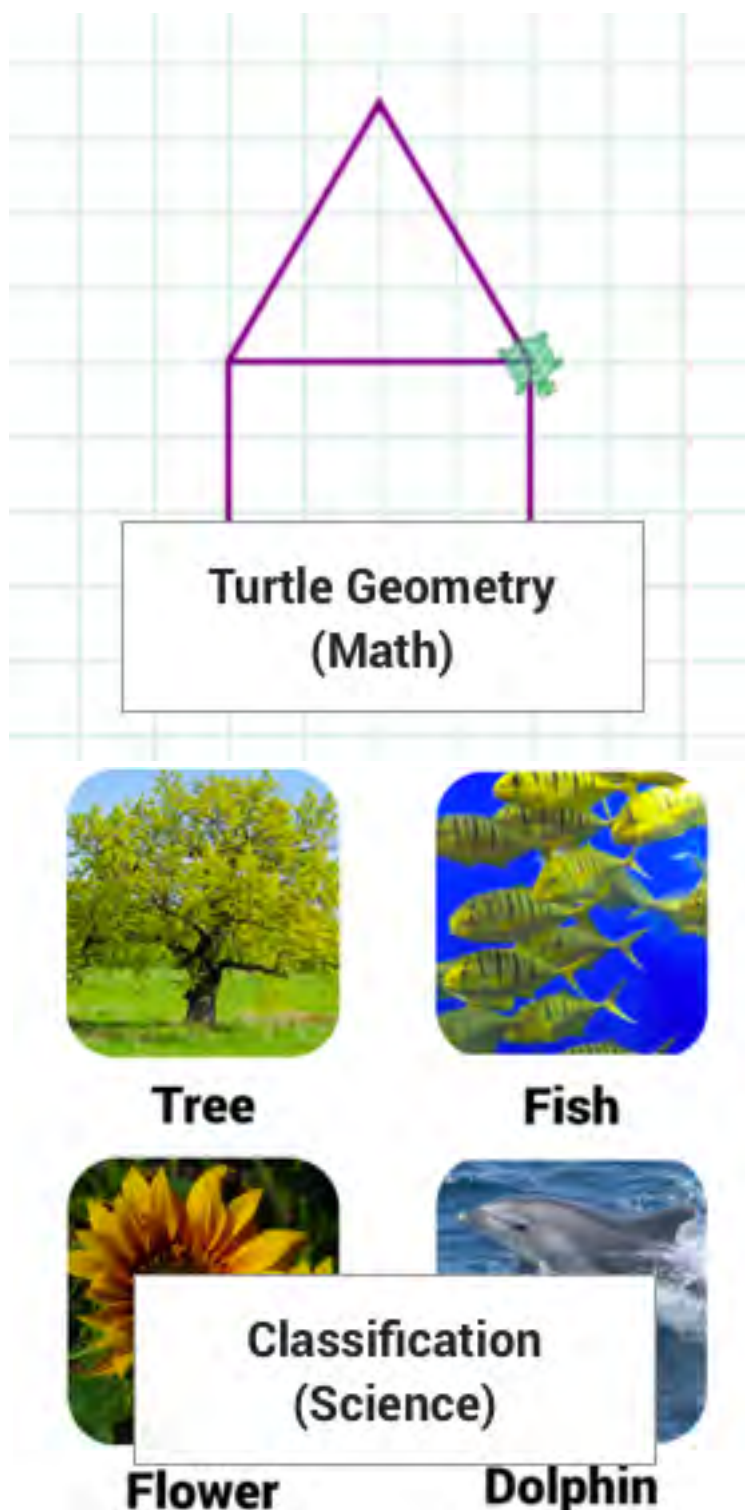
Este é o seu lugar para fazer perguntas, compartilhar ideias e fornecer *feedback* à equipe e aos colegas do curso. Os assistentes de ensino estarão monitorando a comunidade durante todo o curso. Se você ainda não aderiu, clique na Comunidade de Pensamento Computacional abaixo e diga oi aos seus colegas e compartilhe suas metas para o curso.

### Exemplo de Aula (Escolha aleatória) Reconhecimento de Padrões (visão geral)

Nesta seção, você pode experimentar algumas atividades que você também pode modificar para uso em sua sala de aula. Essas atividades envolvem a edição de código existente, mas nenhuma experiência anterior com programação é necessária. Esta é uma oportunidade para experimentar o processo de PC em diferentes assuntos, então sinta-se à vontade para experimentar as atividades fora de sua área de assunto. Clique em uma atividade abaixo para testá-la. Explore, experimente, jogue e divirta-se (exemplos de vídeos e jogos disponíveis).







*Exemplo de um jogo*

Você já ouviu falar do jogo 20 perguntas? Este é o jogo em que uma

pessoa pensa em algo, por exemplo, como um avião, e a outra pessoa só pode fazer 20 perguntas, sim ou não, para tentar adivinhar a resposta correta. Uma estimativa recente do número de diferentes espécies de vida na Terra é entre 8 e 9 milhões.

Quantas perguntas você acha que seria necessário para adivinhar com confiança qualquer espécie na Terra em que eu estava pensando?



Espaço para resposta

---

### Encontrando Padrões (lição)

Aqui está um resumo de onde o processo do PC pode ser encontrado nesta seção. Sinta-se à vontade para observar as atividades em outros assuntos, já que o PC é altamente interdisciplinar.

#### Compactação de Dados

- Decomposição - Tratar uma imagem não apenas como um todo, mas também como seus pixels individuais.
- Reconhecimento de Padrões - Aplicando diferentes *bitmasks* para ver como os valores impactaram o tamanho da memória da imagem, bem como sua qualidade.
- Abstração / Generalização - Usando os padrões para decidir sobre um *bitmask* que reduziria o tamanho da imagem sem sacrificar a qualidade.
- *Design* de Algoritmo - Embora você não tenha desenvolvido o algoritmo nesta seção, você deve ter notado vários atributos da máscara de bits que podem ajudar a refinar o algoritmo usado (que já está programado).

#### Música

- Decomposição - Olhando para as várias partes de uma música você deve dividi-la em seus componentes, como batidas, notas e harmonia, para que você possa examinar cada uma por si.
- Reconhecimento de Padrões - Ajustar cada uma das variáveis, como a duração de uma nota ou a frequência de uma nota em uma escala e descobrir quais configurações você prefere.

- Abstração / Generalização - Adicionando um terceiro piano para servir de harmonia.
- *Design* de algoritmo - Embora você não tenha criado o algoritmo para esta seção, ele foi desenvolvido para facilitar a modificação das configurações, usar outras escalas e adicionar pianos adicionais para que você possa explorar. Você pode ter pensado em outros atributos da música e como eles poderiam ser incorporados para melhorar ainda mais a música.

### Geometria da Tartaruga

- Decomposição - Traduzindo um objetivo, como desenhar um quadrado em uma série de etapas e curvas.
- Reconhecimento de Padrões - Ver a correlação entre a repetição de uma etapa e os lados da forma desenhada.
- Abstração / Generalização - Usando o padrão do ângulo do giro e o número de etapas para desenvolver uma compreensão da relação entre o número de etapas, os ângulos e os lados de um polígono.
- *Design* de Algoritmo - Desenvolvendo um algoritmo para criar um polígono de qualquer formato. Existem muitas maneiras de estender esse algoritmo. Por exemplo, considere o número de lados ao determinar por quanto tempo a tartaruga deve dar um passo.

### Organização

- Decomposição - Observando as diferentes formas pelas quais os organismos podem ser classificados. Por exemplo, por serem aquáticos ou terem asas para voar no ar.
- Reconhecimento de padrões - Desenvolver um método para classificar de forma eficiente alguns organismos com base nas características compartilhadas.
- Generalização de abstração ou padrão - Determinar quanto tempo levaria para classificar alguns organismos com base nesse método pode ajudá-lo a prever quantas perguntas ou classificações são necessárias para classificar todos os organismos conhecidos.
- Projeto de algoritmo - Nesta seção você não desenvolveu um algoritmo, mas você pode ter pensado em outras maneiras mais eficientes de classificar os organismos além do método usado na atividade.


Agora que você já teve uma visão geral do processo do PC que outros tópicos do seu currículo podem beneficiar os alunos aplicando o processo de PC? Compartilhe suas ideias na comunidade do curso.

### *Exemplo de atividade proposta nesta etapa do curso*

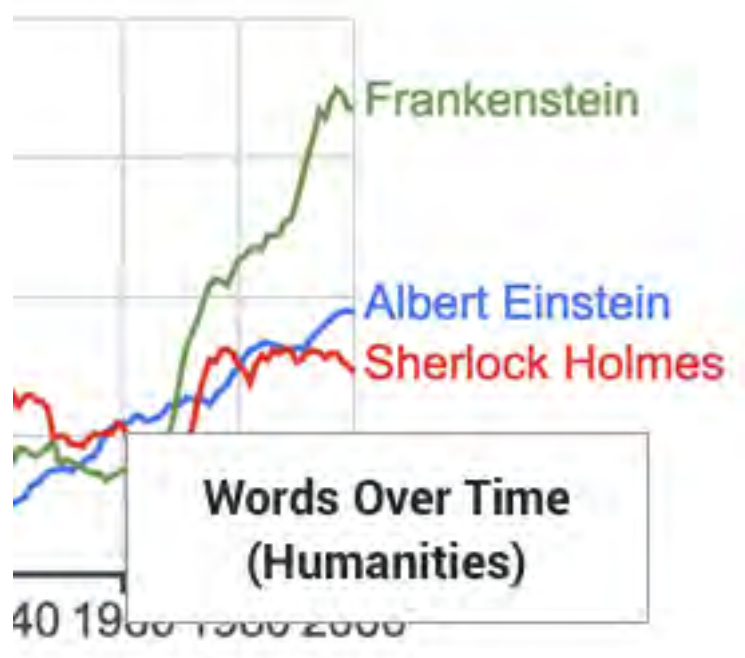
Existe uma lenda de uma sala com 3 postes (torres). Na torre mais à esquerda, 64 discos de ouro são empilhados do maior disco até o menor (uma versão simplificada é mostrada à direita). Aqueles encarregados da tarefa de mover todos os discos de um posto para outro foram informados de que o mundo acabaria quando essa tarefa estivesse concluída.

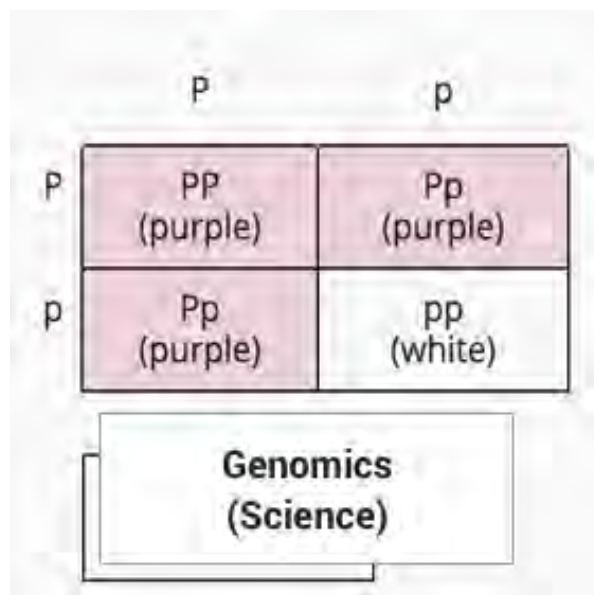
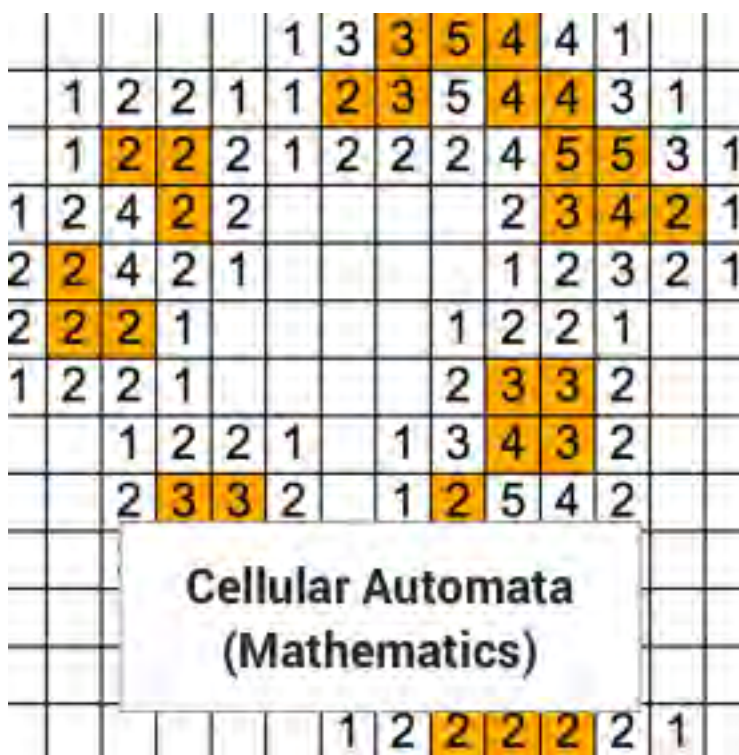
As regras a seguir ao resolver o quebra-cabeça da torre são:

1. Os discos devem ser removidos um por vez a partir do topo de uma torre e colocados no topo de outra torre.
2. Nenhum disco pode ser maior do que qualquer disco abaixo dele (ou seja, os discos em cada torre fazem uma forma de pirâmide). Se existe alguém secretamente completando esta tarefa movendo um disco por segundo, quanto tempo você acha que isso levaria para resolvê-lo?

☐

## Outros exemplos de atividades





## Seleccionamos o Automato celular (Matemática)

Você recebe uma grade de quadrados onde cada quadrado pode ser clicado para trazer esse quadrado para "vida". Além do quadrado selecionado, alterando as cores, os números mostram quantos quadrados adjacentes estão vivos. Por exemplo, se apenas um quadrado for selecionado, todos os quadrados ao redor mostrarão um 1. A seguir estão alguns exemplos:



	1	1	1
	1		1
	1	1	1

				1	1	1			
				2	1	2			
		1	2	5	4	5	2	1	
		1	1	4	4	4	1	1	
		1	2	5	4	5	2	1	
				2	1	2			
				1	1	1			

	1	2	3	2	1
	2	2	4	2	2
	3	4	8	4	3
	2	2	4	2	2
	1	2	3	2	1

Tente clicar em alguns dos quadrados na grade e, em seguida, pressione [Avançar](#) algumas vezes para ver como a grade muda. Você também pode clicar em [Iniciar](#) / [Parar](#) para ver se continua sendo executado. O que faz com que os quadrados apareçam e desapareçam?

O que torna este jogo interessante são as regras que o governam. As regras são definidas nas linhas 2 e 3 do código e estão definidas para o seguinte:

- Linha 2 Se o quadrado está vivo (destacado) e há 2 ou 3 quadrados vivos adjacentes, então o quadrado permanece vivo ou sobrevive.
- Linha 3 Se o quadrado está morto (não alto), mas é cercado por 3 ou 6 quadrados vivos adjacentes, então o quadrado ganha vida.

Caso contrário, o quadrado morre (muda de laranja para branco). Lembre-se que isto é apenas uma simulação, nenhum quadrado real foi danificado! Aqui estão alguns exemplos:

	1	1	1		
1	3	3	3	1	
1	3	4	3	1	
1	3	3	3	1	
	1	1	1		

→

1	2	3	2	1	
2	2	4	2	2	
3	4	8	4	3	
2	2	4	2	2	
1	2	3	2	1	

1	1	1	1	1	1
1	1	3	3	1	1
1	3	4	4	3	1
1	3	4	4	3	1
1	1	3	3	1	1
1	1	1	1	1	1

→

	1	2	2	1	
1	2	2	2	2	1
2	2	4	4	2	2
2	2	4	4	2	2
1	2	2	2	2	1
	1	2	2	1	

1	2	3	2	1	
1	1	2	1	1	
1	2	3	2	1	

→

1	1	1			
2	1	2			
3	2	3			
2	1	2			
1	1	1			

1	1	1			
1	1	1			
1	1	1			

→


Clique na grade para criar qualquer padrão desejado e aplique as regras em um ciclo de cada vez, pressionando [Avançar](#) repetidas vezes, ou deixe-o passar por vários ciclos, clicando em Iniciar. O que você está testemunhando é um exemplo de

complexidade, onde o comportamento interessante surge como resultado de múltiplos elementos interagindo uns com os outros. Tais interações interessantes vêm de regras arbitrárias!

Esses padrões foram estudados durante anos e certos grupos foram descobertos da mesma forma que se pode descobrir um tipo de animal na natureza. Tente recriar esses tipos por conta própria, criando padrões na grade e pressionando Iniciar:

- Estável - Padrões que não mudam com o tempo.
- Osciladores - Padrões que mudam com o tempo, mas se repetem a cada  $n$  ciclos.
- *Flyers* - Padrões que essencialmente permanecem os mesmos ao longo do tempo. Mas se movem pelo tabuleiro como um pássaro ou uma nave espacial.
- Descubra seu próprio tipo de padrão.

Outra maneira de modificar o sistema é alterar as próprias regras. As regras usadas nessa simulação são apenas um exemplo do que é mais amplamente conhecido como autômatos celulares e um conjunto inteiramente novo de propriedades surge para cada tipo de regra. A simulação usa a *string* de regra "S/B", onde S se refere à sobrevivência e B se refere a nascimentos. Nesta simulação o Jogo da Vida S/B é 23/36 que corresponde a uma célula viva com 2 ou 3 células vivas adjacentes sobrevivendo e uma célula morta com 3 ou 6 células vivas adjacentes se tornando vivas. Você pode ver muitas regras diferentes que foram estudadas ao longo dos anos e experimentá-las mudando as linhas 2 e 3 na simulação. Se você alterar as regras, precisará pressionar:



Pressione o botão para executar o código novamente. A Matemática é um tema interdisciplinar e o estudo de sistemas e comportamentos emergentes tem influenciado e iniciado novas áreas de pesquisa em ciências e humanidades. Novos *insights* e novos campos de estudo são frequentemente o resultado quando os algoritmos são projetados e aplicados.

Se você estiver interessado em aprender mais sobre esse assunto, procure na Internet por autômatos celulares e *Game of Life*. Stephen Wolfram<sup>55</sup> desenvolveu um sistema para criar regras para autômatos celulares e estudou extensivamente suas propriedades. Você pode ler mais sobre isso no *Wolfram's Mathworld* ou em *A New Kind of Science*, que é gratuito para o público *online*. Padrões potenciais com os quais essa atividade poderia se alinhar, se usada com os alunos.

---

<sup>55</sup> Nota dos autores: <http://mathworld.wolfram.com/GameofLife.html>

## Projeto Final

Meta: Use as habilidades deste curso para aumentar a eficiência ou a eficácia da integração do PC no ensino em sala de aula.

Parte I: Na Parte I deste projeto, você criará um texto de como o PC se aplica ao seu domínio ou assunto (disciplina ministrada pelo aluno).

Parte II: Na Parte II deste projeto, você documentará seu plano para integrar pelo menos um conceito do PC em uma lição, atividade, unidade, projeto, módulo ou currículo.

Exemplos de projetos que são planos de aula podem ser vistos em:

- [Exploring Computational Thinking Lesson Plans](#)

## Criptografando uma sentença

Veja o plano: [Lesson Plan](#)

- Este plano de aula permite que o aluno desenvolva um código de criptografia, codifique uma sentença e depois desenvolva um algoritmo para codificação e decodificação.
- Área: Ciência da Computação
- Assunto: Algoritmos e Complexidade
- Idade Sugerida: 11-18
- Tipo: Lição

## Pensamento Algorítmico

Veja o plano: [Lesson Plan](#)

Este plano de aula demonstra que um algoritmo é um conjunto preciso de instruções passo a passo. Os alunos serão solicitados a criar algoritmos orais para resolver problemas que outros alunos possam usar de maneira eficaz.

---

### ECT Lesson Plan: Algorithmic Thinking

---

#### *Lesson plan at a glance...*

<b>Núcleo temas(s)</b>	Ciência da Computação, Matemática
<b>Assunto area(s)</b>	Algoritmos e Complexidade
<b>Idade</b>	11 a 18 anos
<b>Prerequisitos</b>	Não
<b>Tempo</b>	<b>Preparaç:</b> 10 a 20 minutos <b>Aula:</b> 35 minutos
<b>Padrões</b>	<b>Núcleo do assunto:</b> <a href="#">CCSS Math</a> <b>CS:</b> <a href="#">CSTA</a> , <a href="#">UK</a> , <a href="#">Australia</a>

#### *Neste plano de aula...*

- [Introdução da aula](#)
- [Materiais e Equipamentos](#)
- [Preparação das Tarefas](#)
- [A Aula](#)
- [Objetivos Educacionais e padrões](#)
- [Informações Adicionais e Recursos](#)

## Visão geral da aula

Os alunos irão explorar o *design* do algoritmo, criando algoritmos orais, dando instruções para outros estudantes seguirem para duplicar um modelo fornecido pelo professor. A interação aluno-aluno promoverá a comunidade e os ajudará a analisar a eficácia de seus algoritmos.

## Materiais e Equipamentos

- Para o Professor:
  - tijolos plásticos intertravados (ex: LEGO®)
  - cartões ou *tangrams*

## Preparação das Atividades

Antes de os alunos virem às aulas: Usando tijolos, cartões, <i>tangrams</i> ou qualquer coisa que possa ser arranjada, crie modelos simples para copiar usando os itens; um para cada quatro estudantes.	10 a 20 minutos
--	-----------------

## A Aula

Atividade de aquecimento: registro em diário sobre o pensamento algorítmico	10 minutos
---	------------

Atividade de aquecimento: registro em diário para o pensamento algorítmico (10 minutos)

Visão geral da atividade: Nesta atividade, os alunos identificarão os principais conceitos utilizados no *design* de algoritmo.

### Atividade:

Discussão: Os alunos respondem ao seguinte aviso em pequenos ou grandes grupos:

Pense em escovar os dentes. Quais etapas você percorre toda vez que escova? Como você daria

Como você daria instruções passo a passo para alguém sobre como escovar os dentes?

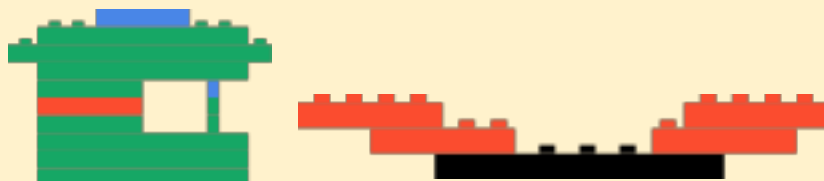
### Atividade 1: Pensamento algorítmico (20 minutos)

Visão Geral da Atividade: Nesta atividade, os alunos usarão o *design* de algoritmo para criar uma série ordenada de instruções para resolver um problema, e outros alunos seguirão o algoritmo. As interações aluno-aluno ajudam-nos a construir

redes de apoio entre pares e fomentam uma comunidade de aprendizagem centrada no aluno.

Notes to the Teacher:

Refer to the the previously built models for this activity.



Atividade:

1. Divida os alunos em grupos de quatro e dê a cada grupo um saco de tijolos.

- A pessoa 1 pode olhar para tudo, mas só pode dar instruções.
- As pessoas 2 e 3 podem não olhar para o modelo, mas serão construídas com base nas instruções da pessoa 1.
- A pessoa 4 contará quantas instruções são dadas até que o modelo esteja completo e anotará as instruções que foram úteis ou não úteis para o grupo.

2. Entregue o modelo à pessoa 1 de cada grupo e ajuste o temporizador por não mais que 5 minutos.

Instrua os alunos a copiar o modelo dado com a maior precisão possível.

3. Depois que o tempo acabar, peça aos alunos que mostrem o modelo completo e compartilhem com a turma inteira.

4. Peça à pessoa 4 para indicar o número de instruções dadas (mesmo que o modelo não tenha terminado).

5. Se você tiver tempo e outro modelo, peça aos alunos que passem por um segundo turno.

Q1: que tipo de instruções funcionou melhor?

Q2: Por que este tipo de atividade é representante dos seres humanos que trabalham com computadores?



**Avaliação:**

A1: instruções específicas simples, passo a passo.

A2: Os computadores só podem entender as instruções exatas que receberam, mesmo que as instruções sejam falhas.

Atividade de encerramento: Discutindo o pensamento algorítmico (5 minutos)

Visão geral da atividade: Nesta atividade, os alunos discutirão os resultados das atividades anteriores.

**Atividade:**

Comece introduzindo o termo “algoritmo” e dando a definição (veja o vocabulário da lição abaixo).

Discussão: Incentive seus alunos a terem uma discussão sobre as atividades anteriores.

Como o termo "algoritmo" se encaixa na equação?

### Objetivos Educacionais e Padrões

Objetivos Educacionais	Padrões
L01: Os alunos poderão escrever uma série de instruções para criar um modelo.	<p><i>Núcleo Comum</i></p> <p><a href="#">CCSS.MATH.PRACTICE.MP4</a>: Modelos Matemáticos.</p> <p><i>Ciência da Computação</i></p> <p><a href="#">AUSTRALIA 8.11 (Collaborating and managing)</a>: Planejar e gerenciar projetos, incluindo tarefas, tempo e outros recursos necessários, considerando segurança e sustentabilidade.</p> <p><a href="#">CSTA L1:6.CPP.5</a>: Construa um programa como um conjunto de instruções passo a passo a ser encenado (por exemplo, fazer a atividade de sanduíche de manteiga de amendoim e geléia).</p> <p><a href="#">CSTA L2.CL.3</a>: Colabore com colegas, especialistas e outras pessoas usando práticas colaborativas, como</p>

	<p>programação em pares, trabalhando em equipes de projeto e participando de atividades de aprendizado ativo em grupo.</p> <p><a href="#">UK 2.1</a>: Design CC, escrever e depurar programas que atinjam objetivos específicos, incluindo o controle ou a simulação de sistemas físicos; resolver problemas decompondo-os em partes menores.</p>
<p>L02: Os alunos serão capazes de articular algumas estratégias importantes e possíveis armadilhas na escrita de algoritmos.</p>	<p><i>Núcleo Comum</i></p> <p><a href="#">CCSS.MATH.PRACTICE.MP3</a>: Construa argumentos viáveis e critique o raciocínio dos outros</p> <p><a href="#">CCSS.MATH.PRACTICE.MP2</a>: Raciocínio abstrato e quantitativo.</p> <p><i>Ciência da Computação</i></p> <p><a href="#">CSTA L2.CT.7</a>: Represente os dados de várias formas, incluindo texto, sons, imagens e números.</p> <p><a href="#">UK 3.6</a>: Entenda como as instruções são armazenadas e executadas dentro de um sistema de computador; entender como dados de vários tipos (incluindo texto, sons e imagens) podem ser representados e manipulados digitalmente, na forma de dígitos binários.</p>

### Informações Adicionais

Termos utilizados no plano de aula

Termo	Definição	Informações Adicionais
Algoritmo	Uma série de instruções que podem ser repetidas várias vezes com o mesmo resultado para uma determinada entrada (por exemplo, receita, software de computador, notas de música)	<a href="http://en.wikipedia.org/wiki/Algorithm">http://en.wikipedia.org/wiki/Algorithm</a>

### Conceitos de PC

Conceito	Definição
----------	-----------

Desenho de Algoritmos	Criando uma série ordenada de instruções para resolver problemas semelhantes
-----------------------	--

### Detalhes Administrativos

Contado	For more info about Exploring Computational Thinking (ECT), visit the ECT website ( <a href="http://g.co/exploringCT">g.co/exploringCT</a> )
Créditos	Developed by the Exploring Computational Thinking team at Google and reviewed by K-12 educators from around the world.
Última atualização	07/02/2015
Copyright	Except as otherwise <a href="#">noted</a> , the content of this document is licensed under the <a href="#">Creative Commons Attribution 4.0 International License</a> , and code samples are licensed under the <a href="#">Apache 2.0 License</a> .

- Exemplo de Comunicação do Plano

Escrever uma carta de comunicação para integração do PC para os colegas e o diretor da escola. Nesta carta você pode mencionar que realizou uma formação sobre o tema e que se sente confiante em iniciar um experimento em sua disciplina.

- [Exemplo de planejamento estratégico](#)

### Plano Administrativo para o Sr X

Este documento descreve como planejo integrar o pensamento computacional em algumas das minhas aulas durante o ano letivo.

Nome	Porque?	Como?	Para Quem?	O que?	Quando?
	<i>Qual é o propósito?</i>	<i>Que conceito de PC será usado?</i>	<i>Para que série e idade de estudantes se destina??</i>	<i>O que será incluído?</i>	<i>Com que frequência isso acontece?</i>
Aula sobre Probabilidade	Automatize eventos estatísticos para simular grande quantidade de números e	Desenho de algoritmos para o reconhecimento de padrões e para a generalização de padrões	Estudantes de matemática do 7º ano, estudantes de estatística do 11º ano	<ul style="list-style-type: none"> <li>• Escreva um programa para simular o lançamento de moedas</li> <li>• Repetir muitas vezes</li> </ul>	Use na lição sobre probabilidade e depois sobre a unidade na probabilidade; pode ser

	ver tendências			<ul style="list-style-type: none"> <li>• Reconhecer padrões</li> <li>• Prever o número de resultados de cabeças em uma determinada situação</li> </ul>	reutilizada nos próximos anos
Atribuição para determinar o melhor dedilhado para a próxima peça musical	Para encontrar o dedilhado mais fácil de usar para uma seção de uma peça musical para encurtar o tempo de aprendizado	Decomposição de reconhecimentos de padrões e da generalização de padrões	Estudantes de piano dos níveis intermediário e avançado	<ul style="list-style-type: none"> <li>• Divide a música em seções da mão e movimentos dos dedos</li> <li>• Procure padrões de dedo e mudanças na localização da mão no piano</li> <li>• Encontre o método mais fácil de transição</li> <li>• Descubra quais padrões surgem para os dedilhados mais fáceis</li> <li>• Faça regras sobre as melhores maneiras de encontrar os dedilhados mais fáceis</li> </ul>	Após o primeiro uso, use para cada peça aumentando o tamanho da música em que os alunos devem trabalhar
Leitura	Melhor determinar o nível de leitura e concentrar os alunos nos livros que os ajudam a evoluir para o próximo	Decomposição do reconhecimento de padrões	Grau K-5, Estudantes do ensino fundamental	<ul style="list-style-type: none"> <li>• Os alunos escolhem livros e registram o número de páginas lidas, palavras por página, quanto</li> </ul>	Reavaliação semanal

	nível, em vez de ler livros que são fáceis demais ou inadequados ao nível deles			<p>tempo demorou e o nível de leitura dos livros que leu a cada semana</p> <ul style="list-style-type: none"> <li>• Análise os registros para descobrir o que os livros de nível de leitura que fazem com que eles leiam muito lentamente e quais deles eles passam direto.</li> <li>• Os alunos, em seguida, usam a folha para determinar (prever) o seu nível de leitura com base em palavras / páginas dadas por minuto</li> </ul>	
--	---	--	--	---	--

### *Proposta do projeto final de Curso*

Após a série de 5 aulas, exercícios, atividades, etc., os alunos passam para a etapa final do curso, que é a da proposta de um projeto.

Nota: o projeto final não pode mais ser submetido, e só está presente aqui para você praticar as habilidades aprendidas na aula.

Parte I. Crie uma declaração sobre o PC em sua área/disciplina.

Instruções: Para a Parte I deste projeto, dizer o que você sabe sobre como o PC pode ser aplicado nos conteúdos de sua disciplina.

1. De que forma o PC pode ser aplicado em sua área de conhecimento/disciplina?

*Você pode escrever sobre o PC em geral ou pode pensar sobre os diferentes conceitos de pensamento computacional e escrever sobre quais se aplicam ao seu domínio ou área / disciplina.*

2. Eu certifico que esta declaração que estou submetendo é fruto do meu próprio trabalho.

Parte II: Desenvolva um plano que integre o PC em seu dia a dia da sala de aula

Instruções para a parte II: Para a Parte II deste projeto, documente seu plano para integrar pelo menos um conceito de PC em uma lição, atividade, unidade, projeto, módulo ou currículo. Esta documentação pode cair em qualquer uma das seguintes categorias:

- Plano curricular que integra pelo menos um conceito de pensamento computacional (decomposição, algoritmos, reconhecimento de padrões, abstração, etc.);
- Plano de aula (roteiro do que os alunos precisam aprender e como será efetivamente implementado durante a aula, palestra ou período de tempo; descreve o título, palavras-chave, objetivos, atividades e avaliações durante o horário de aula);
- Avaliação (uma atividade, teste ou projeto que avalia ou mede se os alunos atingiram os objetivos e em que grau);
- Atividade da lição (uma ação ou um grupo de ações feitas pelos alunos com o objetivo de apoiar os objetivos da aula);
- Projeto (um trabalho no qual os alunos ganham habilidades trabalhando por um período de tempo especificado para resolver um problema, investigar ou responder a um desafio, problema ou tarefa complexos);
- Plano de Unidade (ou Plano do Módulo; um grupo de lições, atividades, avaliações e projetos relacionados que compartilham um objetivo maior comum, geralmente concluído durante o período de vários dias a várias semanas).

Elabore um plano de comunicação para seus alunos, pais ou comunidade escolar explicando:

- o que é o pensamento computacional
- porquê o pensamento computacional é importante e
- o como e onde você começará a integrá-lo (por exemplo, assunto, termo, currículo, projeto, atividade, lição, projeto, currículo pós-escola, projeto de serviço, módulo, etc.)
- outros!

## Exemplo 2 - Conteúdos do curso PC/Google

Outro tipo de proposta de atividades ligada a um livro em particular (selecionamos o livro *CS Unplugged* pois, no sítio, as atividades estão disponíveis em diversas línguas, inclusive o Português, e também pelo fato desse livro ter sido



citado anteriormente neste texto). Esse tipo de sugestão de atividades não é exatamente de formação de professores, mas são comuns para vários dos livros apresentados neste texto. Embora muitos livros não possuam versão para o Português, os sítios apresentam a possibilidade de obter as atividades (não o texto do livro) em outras línguas.

## Atividade 1

### Contando os Pontos—Números Binários

#### Sumário

Os dados são armazenados em computadores e transmitidos como uma série de zeros e uns. Como podemos representar palavras e números usando apenas esses dois símbolos?

#### Correlações curriculares

- Matemática: Representação de números em outras bases. Representação de números na base dois.
- Matemática: Sequências e padrões sequenciais, descrição de uma regra para esses padrões. Padrões e relacionamentos com potências de dois.

#### Habilidades

- Contagem
- Comparação
- Sequenciamento

#### Idade

- A partir de 7 anos

#### Material

Será necessário confeccionar um conjunto de cinco cartões binários (ver página 4) para a demonstração.

- Cartões A4 com caras sorridentes são igualmente adequados.

#### Cada criança precisará de:

- Um conjunto de cinco cartões.
- Fotocopiar a Fotocópia-Mestre: Números Binários (página 6) sobre um cartão e recortá-la.
- Planilha de Atividade: Números Binários (página 5)

#### Existem atividades de extensão adicionais, para as quais cada criança precisará de:

- Planilha de Atividade: Trabalhando com Binários (página 7)
- Planilha de Atividade: Enviando Mensagens Secretas (página 8)
- Planilha de Atividade: Correio Eletrônico e *Modems* (página 9)
- Planilha de Atividade: Contando acima de 31 (página 10)
- Planilha de Atividade: Mais sobre números binários (página 11).

## **Anexo IV – Avaliação de Materiais Educacionais Voltados ao Desenvolvimento do Pensamento Computacional nas Escolas Brasileiras**

Segue no Volume 2 deste trabalho.