

**UMA INVESTIGAÇÃO SOBRE O  
PENSAMENTO  
COMPUTACIONAL  
COM O USO DA PLATAFORMA  
*PROGRAMAÊ!***

**GUARAPUAVA**

**2022**

**UNIVERSIDADE ESTADUAL DO CENTRO-OESTE, UNICENTRO-PR**

**UMA INVESTIGAÇÃO SOBRE O  
PENSAMENTO COMPUTACIONAL  
COM O USO DA PLATAFORMA *PROGRAMAÊ!***

**DISSERTAÇÃO DE MESTRADO**

**ANA PAULA GERMANO**

**GUARAPUAVA, PR**

**2022**

Catálogo na Publicação  
Rede de Bibliotecas da Unicentro

G373i Germano, Ana Paula  
Uma investigação sobre o pensamento computacional com o uso da plataforma Programaê! / Ana Paula Germano. -- Guarapuava, 2022.  
xii, 74 f. : il. ; 28 cm

Dissertação apresentada à Universidade Estadual do Centro Oeste, como parte das exigências do Programa de Pós-Graduação em Ensino de Ciências Naturais e Matemática, para obtenção do título de Mestre. Área de concentração em Educação e Aprendizagem de Ciências Naturais e Matemática, 2022.

Orientadora: Profa. Dra. Joyce Jaqueline Caetano, Profa. Dra. Ana Elisa Tozetto Pierarski

Banca examinadora: Profa. Dra. Marcell Behm Goulart, Prof. Dr. Clodogil Fabiano Ribeiro dos Santos, Profa. Dra. Joyce Jaqueline Caetano, Profa. Dra. Ana Elisa Tozetto Piekarski

Bibliografia

1. Curso Técnico. 2. Lógica Computacional. 3. Programação em Blocos. I. Título. II. Programa de Pós-Graduação em Ensino de Ciências Naturais e Matemática.

CDD 500

**ANA PAULA GERMANO**

**UMA INVESTIGAÇÃO SOBRE O PENSAMENTO  
COMPUTACIONAL COM O USO DA PLATAFORMA *PROGRAMAÊ!***

Dissertação apresentada à Universidade Estadual do Centro-Oeste, como parte das exigências do Programa de Pós-Graduação em Ensino de Ciências Naturais e Matemática, área de concentração em Ensino e Aprendizagem de Ciências Naturais e Matemática, para a obtenção do título de Mestre.

Prof<sup>a</sup> Dr<sup>a</sup> Joyce Jaquelline Caetano  
Orientadora

Prof<sup>a</sup> Dr<sup>a</sup> Ana Elisa Tozetto Piekarski  
Coorientadora

GUARAPUAVA, PR  
2022

**ANA PAULA GERMANO**

**UMA INVESTIGAÇÃO SOBRE O PENSAMENTO  
COMPUTACIONAL COM O USO DA PLATAFORMA *PROGRAMAÊ!***

Dissertação apresentada à Universidade Estadual do Centro-Oeste, como parte das exigências do Programa de Pós-Graduação em Ensino de Ciências Naturais e Matemática, área de concentração em Ensino e Aprendizagem de Ciências Naturais e Matemática, para a obtenção do título de Mestre.

Aprovada em 05 de agosto de 2022

Profa. Dra. Marceli Behm Goulart – UEPG

Prof. Dr. Clodogil Fabiano Ribeiro dos Santos – PPGEN/UNICENTRO

Profa. Dra. Joyce Jaqueline Caetano  
Orientadora

Profa. Dra. Ana Elisa Tozetto  
Piekarski Coorientadora

GUARAPUAVA, PR

2022

## SUMÁRIO

<b>RESUMO</b> .....	<b>08</b>
<b>ABSTRACT</b> .....	<b>09</b>
<b>CAPÍTULO I - INTRODUÇÃO</b> .....	<b>12</b>
<b>CAPÍTULO II - PENSAMENTO COMPUTACIONAL NO ENSINO MÉDIO</b> .....	<b>18</b>
<b>2.1 PC segundo a literatura</b> .....	<b>18</b>
<b>2.2 PC e Ciência da Computação</b> .....	<b>22</b>
<b>2.3 PC no contexto educacional</b> .....	<b>24</b>
<b>2.4 Gamificação</b> .....	<b>27</b>
<b>CAPÍTULO III - O OBJETO DE ESTUDO E O ARCABOUÇO METODOLÓGICO</b> .....	<b>30</b>
<b>3.1 Natureza da pesquisa</b> .....	<b>30</b>
<b>3.2 O Campo da Pesquisa</b> .....	<b>31</b>
<b>3.3 O local e os participantes da pesquisa</b> .....	<b>32</b>
<b>3.4 Programação em blocos</b> .....	<b>33</b>
<b>3.5 A plataforma <i>Programaê!</i></b> .....	<b>34</b>
<b>3.6 O Roteiro de atividades</b> .....	<b>36</b>
<b>3.7 Coleta e Análise dos dados</b> .....	<b>41</b>
<b>CAPÍTULO IV - A EXPERIÊNCIA DE ENSINO ENVOLVENDO PC</b> .....	<b>44</b>
<b>4.1 As atividades realizadas e os Pilares do PC</b> .....	<b>45</b>
<b>4.2 Síntese dos Resultados</b> .....	<b>61</b>
<b>CAPÍTULO V - CONSIDERAÇÕES FINAIS</b> .....	<b>66</b>
<b>REFERÊNCIAS</b> .....	<b>69</b>

Dedico este trabalho à  
Ana Paula do futuro

## **AGRADECIMENTOS**

Agradeço a Deus por me dar saúde e força.

Agradeço aos meus professores orientadores, Márcio, Joyce e principalmente Ana Elisa, por não desistirem de mim.

Agradeço aos meus pais e irmãos que me ajudaram na medida do possível.

Agradeço aos professores, amigos e colaboradores: Mônica, Samuel, Rita, Felipe, Maurício, Nerielly, João e Paulo que também ajudaram e acompanharam minha trajetória



## RESUMO

Ana Paula Germano Stadler. Uma investigação sobre o Pensamento Computacional com o uso da plataforma *Programaê!*

O Pensamento Computacional (PC) pode ser entendido como uma abordagem para resolução de problemas, projetos de sistemas e compreensão do comportamento humano que utiliza conceitos fundamentais da ciência da computação, uma habilidade fundamental para todos. Considerando a necessidade de desenvolver competências e habilidades que estimulem o PC, os profissionais da educação devem proporcionar aos alunos práticas adequadas e motivadoras para esse fim. Nesse contexto, o objetivo desta pesquisa é investigar o desenvolvimento do PC por meio da plataforma Programaê!, que utiliza a programação em blocos como ferramenta para esse fim. Para isso, um roteiro de atividades da plataforma Programaê! foi preparado, com o objetivo de estimular o PC em alunos do Ensino Técnico em Nível Médio. A experiência de ensino envolveu alunos do 1<sup>a</sup> ano do curso Técnico em Informática do Colégio Estadual Francisco Carneiro Martins, em Guarapuava-PR. Os dados coletados durante a experiência de ensino foram analisados seguindo a metodologia de análise de conteúdo. A análise consistiu em relacionar as práticas adotadas durante o desenvolvimento das atividades selecionadas às quatro dimensões do TC (também chamadas de pilares): abstração, decomposição, reconhecimento de padrões e algoritmos. Os resultados mostraram que os alunos desenvolveram habilidades de PC, de acordo com as dimensões, o que é evidenciado pelo uso adequado dos elementos de programação (eventos, repetições, estruturas de seleção e funções) nas atividades propostas. Considerando o PC como tema transversal, o produto didático resultante desta pesquisa constitui-se em um roteiro de atividades para apoiar professores de diferentes disciplinas, que buscam estimular o desenvolvimento das habilidades do PC.

**Palavras-Chave:** Curso Técnico; Lógica Computacional; Programação em Blocos.

## ABSTRACT

Computational Thinking (CT) can be understood as an approach to problem solving, systems design and understanding human behavior that uses fundamental concepts of computer science, a fundamental skill for everyone. Considering the need to develop skills and abilities that stimulate CT, education professionals should provide students with appropriate and motivating practices for this purpose. In this context, this research aim is investigate the development of the PC through the Programaê! platform, which uses block-based programming as a tool for this purpose. For this, a roadmap of activities for the Programaê! platform was prepared, aiming to stimulate CT in students of Secondary Technical Education. The teaching experience involved students in the 1st year of the Technical Course in Informatics at Colégio Estadual Francisco Carneiro Martins, in Guarapuava-PR. Data collected during the teaching experience were analyzed following the content analysis methodology. The analysis consisted of relating the practices adopted during the development of the selected activities to the four CT dimensions (also called pillars): abstraction, decomposition, pattern recognition and algorithms. The results showed that the students developed CT skills, according to the dimensions, which is evidenced by the adequate use of programming elements (events, loops, selection structures and functions) in the proposed activities. Considering CT as a transverse theme, the didactic product resulting from this research constitutes a roadmap of activities to support teachers from different disciplines, who seek to encourage the development of CT skills.

**Keywords:** Secondary Technical Education; Computational logic; Block-based programming.

## LISTA DE FIGURAS

<i>Figura 1</i> - Competências para o Século XXI.....	19
<i>Figura 2</i> - Semelhanças e diferenças entre pensamento matemático e PC.....	26
<i>Figura 3</i> - Página inicial da Plataforma Programaê!.....	36
<i>Figura 4</i> - Frozen: Trilha 11.....	39
<i>Figura 5</i> - Basquete NBA: Trilha 6 (Área de trabalho).....	40
<i>Figura 6</i> - Frozen: Trilha 4 (Bloco repita).....	40
<i>Figura 7</i> - Frozen: Trilha 5 (Bloco repita).....	40
<i>Figura 8</i> - Flappy Bird: Trilha 5.....	41
<i>Figura 9</i> - Infinity Play Lab: Trilha 10 - Tela inicial.....	46
<i>Figura 10</i> - Infinity Play Lab: Trilha 10 - Produção do aluno A.....	47
<i>Figura 11</i> - Infinity Play Lab: Trilha 10 - Blocos de eventos.....	48
<i>Figura 12</i> - Aventureiro Minecraft (Cenário).....	50
<i>Figura 13</i> - Aventureiro Minecraft: Trilha 3.....	51
<i>Figura 14</i> - Aventureiro Minecraft: Trilha 7 - Blocos.....	51
<i>Figura 15</i> - Aventureiro Minecraft: Trilha 7 - Cenário.....	52
<i>Figura 16</i> - Frozen: Trilha 1.....	53
<i>Figura 17</i> - Frozen: Trilha 10.....	53
<i>Figura 18</i> - Frozen: Trilha 10 - Resolução do aluno D.....	54
<i>Figura 19</i> - Frozen: Trilha 13 - Criação do círculo.....	54
<i>Figura 20</i> - Frozen: Trilha 14 - Uso da função ‘círculo’.....	55
<i>Figura 21</i> - O Artista: Trilha 8 - Definindo uma função.....	56
<i>Figura 22</i> - O Artista: Trilha 8 - Utilizando a função ‘flor’.....	56
<i>Figura 23</i> - O Artista: Trilha 8 - Padrão de flores.....	57
<i>Figura 24</i> - A Jornada do Herói Minecraft: Tela inicial.....	58
<i>Figura 25</i> - Jornada do Herói Minecraft: Trilha 10.....	59
<i>Figura 26</i> - Jornada do Herói Minecraft: Trilha 10 - Área de trabalho.....	60
<i>Figura 27</i> - Jornada do Herói Minecraft: Trilha 10 - Uso do bloco ‘repita’.....	60

## LISTA DE QUADROS

<i>Quadro 1</i> - Os quatro pilares do PC.....	21
<i>Quadro 2</i> - Roteiro de Atividades da plataforma Programaê!.....	38
<i>Quadro 3</i> - Estratégias do raciocínio matemático e Práticas do PC para análise dos dados....	42
<i>Quadro 4</i> - Elementos e práticas de programação nas atividades do Roteiro.....	62

# CAPÍTULO I

## INTRODUÇÃO

Atualmente, muitas pessoas, direta ou indiretamente, precisaram utilizar o computador e a Internet com mais intensidade, isso se tornou ainda mais evidente com a necessidade de medidas de afastamento social devido à pandemia da Covid-19. Tal acontecimento potencializou o uso de computadores e dispositivos móveis, e aproximou pais, alunos e vários segmentos da sociedade do uso pedagógicos das Tecnologias Digitais de Informação e Comunicação (TDIC). Esse movimento intenso em torno das aulas remotas mostrou a importância do aprendizado das TDICs, mas mostrou também que é necessário ir além da mera manipulação das ferramentas tecnológicas.

Com os desafios da recuperação pós-pandemia, novas demandas profissionais e oportunidades surgiram. Um estudo realizado pelo Senai, Universidade Federal do Rio Grande do Sul e Agência Alemã de Cooperação Internacional revelou profissões da era digital emergentes nos setores de software e Tecnologia da Informação (TI). Destacam-se as funções de programador, cientista de dados e analista de cibersegurança (SENAI, 2021).

Por isso, os jovens que decidem frequentar um curso técnico integrado ao Ensino Médio, enxergam em um curso profissionalizante uma oportunidade para construir conhecimento adicional. Além disso, ao escolherem percorrer esta etapa do ensino em um curso profissionalizante, almejam especializar-se em determinado conhecimento para ingressar no mercado de trabalho.

Entre algumas destas escolhas está o Curso Técnico em Informática. Na Rede Pública de Ensino do Estado do Paraná, a matriz curricular do 1º ano do Curso Técnico em Informática contempla as disciplinas de Lógica Computacional e Linguagem de Programação. Essas disciplinas reúnem um conjunto de habilidades e conceitos referenciadas como Pensamento Computacional (PC). Para Brackman:

é necessário tratar da tecnologia não apenas como ferramenta de aprendizagem, haja visto que, além de ser fascinante recurso didático pedagógico de elevado impacto, também pode ser utilizada como uma forma de estruturar problemas e encontrar soluções para os mesmos, utilizando fundamentos da Computação (Pensamento Computacional) (BRACKMAN, 2017, p.20).

Ainda de acordo com Brackmann:

O Pensamento Computacional é uma distinta capacidade criativa, crítica e estratégica humana de saber utilizar os fundamentos da Computação, nas mais diversas áreas do conhecimento, com a finalidade de identificar e resolver problemas, de maneira individual ou colaborativa, através de passos claros, de tal forma que uma pessoa ou uma máquina possam executá-los eficazmente (BRACKMANN, 2017, p.29).

A citação acima corrobora com o fato de que o PC não está necessariamente atrelado à programação de computadores. Assim como consta nos estudos de Jiannete Wing (2006), atualmente o nome mais citado na literatura sobre o assunto: “Pensamento computacional é uma habilidade fundamental para todos, não somente para cientistas da computação. À leitura, escrita e aritmética, deveríamos incluir PC na habilidade analítica de todas as crianças” (WING, 2006).

No curso Técnico em Informática Integrado, no Colégio Estadual Francisco Carneiro Martins, em Guarapuava-PR, existe a função de coordenador de curso. Essa função fica sob a responsabilidade de um professor, indicado pela direção do Colégio, cujas atribuições contemplam, entre outras, a busca por atividades extracurriculares que proporcionem aos educandos experiências diferenciadas e complementares às vivenciadas no cotidiano das aulas. Nesse sentido, são organizadas palestras proferidas por profissionais da área e/ou por professores das Universidades para terem conhecimento tanto de nichos no mercado de trabalho como para tratar de temas que vão além dos estudados nas disciplinas do curso, de interesse acadêmico e científico. Cabe ao coordenador também promover visitas técnicas e visitas aos *campi* universitários, participações em Semanas de Estudos e oficinas, indo além dos muros do colégio para ampliar as experiências e visão de mundo dos alunos, na expectativa de motivá-los a permanecer no curso.

Dentre as atividades extracurriculares que possuem boa aceitação pelos alunos e costumam repetir-se nos últimos anos estão as oficinas de programação utilizando diferentes abordagens, como por exemplo: a plataforma *Code.org*, a linguagem *Scratch*, o *Greenfoot* e a plataforma Arduino.

Ao atuar como professora, percebi que os alunos enfrentam logo no início do curso as dificuldades próprias da área, como por exemplo, os conceitos de raciocínio lógico estreitamente relacionados à Matemática e Teoria da Computação, empregados na construção de algoritmos. Também observei que as ferramentas utilizadas nas oficinas extracurriculares eram atrativas para os alunos.

O processo de aprendizagem dos conceitos iniciais de programação é complexo, muitas vezes necessitando de um nível de abstração ainda não desenvolvido na maioria dos alunos iniciantes, havendo a necessidade de se criar um ambiente mais diversificado e motivador para o aluno (ZANETTI; OLIVEIRA, 2015).

Embora para aqueles que ainda não tiveram alguma experiência com o desenvolvimento do PC e pareça ser algo extremamente difícil, a facilidade de acesso aos conteúdos relacionados na Internet tem permitido que um número cada vez maior de pessoas construa conhecimentos nessa área e possa ingressar no mundo da informática, robótica e das novas tecnologias digitais (ZANETTI; OLIVEIRA, 2015). Para os ingressantes nos cursos de Informática integrados ao Ensino Médio, trabalhar em um ambiente sem conhecimento prévio sobre o assunto, pode proporcionar um estímulo a diferentes formas de raciocínio, muitas vezes sem um precedente conhecido em seu mundo acadêmico, propiciando a formação de cidadãos preparados para a profissão que almejam.

Resnick (2007) ressalta que a utilização de ferramentas de programação visual e de encaixe auxiliam no processo criativo e sugere a adoção do PC, através do ensino de programação, na educação básica em disciplinas complementares. Considerarei, portanto, o uso das ferramentas utilizadas nas atividades extracurriculares como uma forma de amenizar as dificuldades de desenvolvimento do raciocínio lógico observadas em sala de aula, como forma de tentar diminuir a angústia desses estudantes e prováveis desistências futuras. A vontade de pôr em prática este anseio deu origem a essa pesquisa de mestrado.

Assim sendo, este trabalho investiga o uso da plataforma *Programaê!* (PROGRAMAÊ! 2018) como prática alternativa na disciplina de Lógica Computacional, para os alunos do 1º ano do Curso Técnico em Informática, destacando-a como um complemento aos conteúdos curriculares e inserindo-a nos planos de trabalho docente. Este trabalho considera o *Programaê!* como uma ferramenta para estimular o PC nos estudantes da disciplina de Lógica computacional.

A escolha da plataforma *Programaê!*, em vez de utilizar a linguagem *Scratch*, foi por conter narrativas digitais que, de acordo com De Almeida e Valente (2012) constituem uma prática pedagógica que se vale da seguinte característica: o uso de recursos digitais para a elaboração de textos e histórias próximos da sua realidade. Os recursos digitais usados pelos alunos auxiliam o processo para expandir e criar possibilidades do poder de imaginação, o que é fortemente incentivado em práticas educativas (DE ALMEIDA; VALENTE, 2012). Outro motivo dessa escolha é que a maioria das atividades propostas na plataforma funciona como

um sistema tutor, em que o aprendiz não precisa de conhecimentos prévios em lógica de programação nem em relação ao manuseio da ferramenta e também porque a interferência do professor é mínima, pois a interface é intuitiva.

É importante destacar que extrair a prática esporádica de participação em minicursos introdutórios de programação e concretizá-las no plano de trabalho docente do curso Técnico em Informática, é algo urgente. Além disso, seria pertinente para atualização curricular e para diminuir o impacto que os estudantes sentem quando se deparam com a resolução de problemas computacionais, acarretando dificuldades no processo de ensino aprendizagem e até mesmo um possível abandono dos estudos no curso profissionalizante.

Outro ponto a ser considerado é a falta da disciplina de Matemática no primeiro ano do curso, visto que está inserida apenas do segundo ano em diante. Essa lacuna gera preocupação, pois não parece correto interromper a construção das habilidades matemáticas dos alunos, as quais estimulam o PC, e são conceitos que estão relacionados entre si.

A Base Nacional Comum Curricular (BNCC) reforça essa ideia, quando incumbe ao professor de Matemática a tarefa de desenvolver o PC nas suas aulas (BRASIL, 2017). Barcelos (2012) aponta essa relação em seus estudos, e para o autor “a Matemática, com sua representação simbólica e sistema de dedução fundamentado em axiomas, é base para o estudo da complexidade de algoritmos e da análise numérica” (BARCELOS, 2012, p.4).

Como na matriz curricular do 1º ano do curso técnico em Informática não tem a disciplina de matemática, sugere-se então, que a disciplina de Lógica Computacional assuma esse papel uma vez que justifica-se, em seus fundamentos teóricos, da seguinte forma: “para a computação, a lógica é essencial por fortalecer noções de sintaxe (através de linguagem), semântica (através do estudo de tabelas verdade) e sistema dedutivo” (PPC, 2021). Assim, a disciplina de Lógica Computacional atuaria para a inserção do PC, na hipótese de motivar o interesse dos alunos pela área e estimular o raciocínio, com atividades diferenciadas.

Considerando que o uso da plataforma Programaê! fornece um contexto e um conjunto de oportunidades potenciais, queremos responder a seguinte pergunta de pesquisa: de que forma as atividades desenvolvidas estimulam o PC nos alunos, assim como contribuem para a superação de dificuldades na disciplina de Lógica Computacional?

Ao investigar o uso da plataforma Programaê! no desenvolvimento do PC com estudantes do 1º ano do Curso Técnico em Informática, este trabalho tem como objetivos específicos:



- Caracterizar PC no Ensino Médio;
- Apresentar a plataforma Programaê!, nas aulas de Lógica Computacional, como forma de inserção do PC;
- Desenvolver um material didático voltado ao trabalho com o PC no Ensino Médio.

A plataforma Programaê!, que é uma das várias iniciativas que incentivam o ensino de programação, foi explorada e será apresentada de forma organizada e direcionada como um material didático, para uso em sala de aula, facilitando sua utilização por parte do professor. Para tanto, este material será oferecido como um produto educacional em forma de *Ebook*, utilizando conceitos de PC como tema transversal de estudo. Esse produto educacional contém as atividades selecionadas, definições de PC e elementos de programação e respostas aos principais questionamentos dos estudantes mediante a experimentação vivenciada no ambiente escolar.

Sugere-se que esse produto educacional seja utilizado como material de apoio para professores do Ensino Médio e dos Cursos Técnicos em Informática, para contribuir na superação da lacuna existente entre as disciplinas de Lógica Educacional, Matemática e Linguagem de Programação.

Como as atividades disponíveis na plataforma Programaê! estão dispostas de maneira aleatória, existindo a necessidade de experimento e exploração por parte do usuário inicial foi construído um roteiro. Algumas atividades foram organizadas em níveis de dificuldade (da mais fácil para a mais difícil) e de acordo com os elementos de programação exigidos e conforme aparecem na plataforma do Code.org, página em que as atividades funcionam de forma online.

Para investigação proposta nesse trabalho foram realizados cinco encontros com os alunos do 1º ano do curso técnico em Informática, de forma remota, o que afetou a capacidade de coletar os dados. As poucas interações que os alunos realizaram durante os encontros propiciaram analisar algumas práticas e estratégias utilizadas por eles para atingir os objetivos das atividades e, através delas, desenvolver algumas habilidades de Pensamento Computacional. Portanto, relacionamos a prática do uso da plataforma com os pilares do PC, no intuito de proporcionar aos alunos o estímulo ao raciocínio lógico. Como o PC não constitui uma metodologia de ensino, buscou-se na gamificação o respaldo para o trabalho do professor durante a aplicação do roteiro de atividades proposto.

Conforme a investigação proposta, este trabalho está organizado na seguinte forma: no primeiro capítulo está apresentada uma revisão bibliográfica sobre os conceitos e abordagens do PC no ensino, bem como sobre gamificação. O capítulo seguinte contém a descrição da metodologia, incluindo detalhes sobre a natureza e o campo da pesquisa, a implementação da proposta, os instrumentos de coleta e análise dos dados. No terceiro e último capítulo são apresentados os resultados sobre a experiência de ensino, conforme os pilares do PC, seguidos das considerações finais.

## CAPÍTULO II

### PENSAMENTO COMPUTACIONAL

A fim de fornecer subsídios para a questão de pesquisa, esta seção apresenta uma revisão sobre PC, PC no âmbito educacional e *gamificação*.

#### 2.1 PC segundo a literatura

O PC é um conceito que vem sendo atualizado constantemente na literatura especializada vigente. Em 1980, Papert, em seu livro “*Mindstorms: Children, Computers, and Powerful Ideas*”, caracterizou o construcionismo como uma perspectiva teórica sobre o uso pedagógico da linguagem LOGO, destacando as ligações entre linguagem de programação e pensamento matemático (BORBA *et al.*, 2016). Wing (2006) destaca que “como outras habilidades que são transmitidas, o PC deveria também ser adicionado como habilidade analítica para as crianças, não limitada à profissionais da computação” (WING, 2006).

*The Partnership for 21st Century Learning* (P21<sup>1</sup>), organização sem fins lucrativos, mediada por líderes educacionais e por instituições como a Apple e a Microsoft, elaborou um modelo que descreve as habilidades, os conhecimentos e as experiências necessárias ao sucesso do indivíduo como cidadão, no trabalho e na vida (MARTINELLI, 2017, p.34). As novas aptidões, no contexto do PC, definidas por P21 são: Criatividade e Inovação; Pensamento Crítico; Solução de Problemas; Comunicação e Colaboração; Informação (procurar, avaliar, gerenciar); Uso de Mídias (construção, legalidade); e TICs (ferramentas, equipamentos) (BRACKMANN, 2017, p.21), as quais podem ser observadas na Figura 1.

Bundy (2007) define PC como uma habilidade para programação e uma metodologia para resolver problemas. Nunes (2011) complementa que a metodologia do PC está relacionada a resolver problemas nas mais diversas áreas. Vicari *et al.* (2018) destacam que esses autores foram os primeiros a descrever o PC como metodologia para resolver problemas.

---

1 <https://www.battelleforkids.org/networks/p21> acesso em 17 de março de 2022.

Figura 1: Competências para o Século XXI



Fonte: P21 traduzido por Martinelli (2017)

Blikstein (2008) afirma que PC corresponde ao uso do computador como ferramenta intelectual e funcional, a fim de aumentar nossa produtividade, inventividade e criatividade. Mestre *et al.* (2015) justificam que a asserção estimulada pelo PC envolve oferecer um método à resolução de problemas, pois envolve a capacidade de compreender as situações propostas em diversas áreas de conhecimento, para assim, criar soluções por meio de modelos matemáticos, científicos ou sociais.

Nesse mesmo viés, a CSTA (*Computer Science Teachers Association*)<sup>2</sup> define o Pensamento Computacional como um processo de resolução de problemas que inclui, não exclusivamente, as seguintes características: (i) formulação de problemas de forma que computadores e outras ferramentas possam ajudar a resolvê-los; (ii) organização lógica e análise de dados; (iii) representação de dados através de abstrações como modelos e simulações; (v) identificação, análise e implementação de soluções visando a combinação mais eficiente e eficaz de etapas e recursos; (vi) generalização e transferência de soluções para uma ampla gama de problemas (CSTA, 2020).

De acordo com o Centro de Inovação para a Educação Brasileira (CIEB), o PC refere-se à capacidade de resolver problemas a partir de conhecimentos e práticas da computação, englobando: sistematizar, representar, analisar e resolver problemas. Tem sido considerado como um dos pilares fundamentais do intelecto humano, acompanhando a leitura, a escrita e a aritmética, visto que ele também é aplicado para descrever, explicar e modelar o universo e

2 <https://www.csteachers.org/>

seus processos complexos (CIEB, 2020).

Shute *et al.* (2017), em uma revisão sobre o conceito de PC, concluíram que ele está evoluindo conforme as pesquisas estão sendo realizadas sobre o assunto. Estabeleceram uma definição e estrutura sobre o PC para apoiar o desenvolvimento pedagógico e avaliação do PC, particularmente para o Ensino Médio, que envolvem: conhecimento prévio e habilidades cognitivas gerais; habilidades de resolução de problemas; e medidas de estilo de aprendizagem. Com o objetivo de resolver problemas de forma eficiente e criativa, o PC envolve os seguintes processos cognitivos (WING, 2006 *apud* SHUTE *et al.*, 2017):

- a) Reformulação do problema: reenquadrar um problema na forma de uma solução familiar;
- b) Recursão: construir um sistema incrementalmente com base nas informações anteriores;
- c) Decomposição do problema: dividir o problema em unidades gerenciáveis;
- d) Abstração: modela os aspectos centrais de problemas ou sistemas complexos;
- e) Testes sistemáticos: tomar ações intencionais para chegar nas soluções.

Esses processos cognitivos, também chamados de pilares do PC, são agrupados e denominados de formas distintas entre os principais trabalhos sobre o tema. No entanto, os mais frequentes são: a decomposição, o reconhecimento de padrões, a abstração e o algoritmo (GROVER; PEA, 2013; LIUKAS, 2015; *Code.Org*, 2016; *BBC Learning*, 2019). Dessa forma, esses quatro pilares contribuem para atingir o objetivo de solucionar problemas, de maneira eficiente (MEDEIROS, 2019). Decompor é dividir em partes, técnica denominada “Dividir para conquistar”. O reconhecimento de padrões é o pilar que permite extrair características semelhantes de situações, tem como base a identificação de similaridades. A abstração está relacionada com o pensamento algébrico, no que diz respeito a não dar um significado único e estanque para determinado problema, deixando suas variáveis assumirem diferentes valores. Por fim, o algoritmo é quem integra todos os outros pilares; pois se refere ao conjunto de instruções e de regras que devem ser criadas e seguidas para a resolução de um problema (MEDEIROS, 2019). No Quadro 1 estão relacionados os quatro pilares do PC e suas características, aplicações e questões.

Quadro 1 - Os quatro pilares do PC

	Características	Aplicações	Questões
Decomposição	<ul style="list-style-type: none"> <li>• Quebrar problemas em partes menores</li> <li>• Analisar aquilo que compõe o todo de forma individualizada</li> <li>• Dividir tarefas para potencializar a construção final de um produto</li> </ul>	<ul style="list-style-type: none"> <li>• Decompor as partes de uma bicicleta, planta, misturas de alimentos, partes de corpo, ou conjunto de dados</li> </ul>	<ul style="list-style-type: none"> <li>• O problema pode ser dividido em partes menores?</li> </ul>
Reconhecimento de padrões	<ul style="list-style-type: none"> <li>• Pode ser entendida como generalização</li> <li>• Encontrar similaridades entre as partes decompostas</li> <li>• Realizar classificações de espécies e relevância de dados</li> </ul>	<ul style="list-style-type: none"> <li>• Distinguir as similaridades em um conjunto de dados ou espécies</li> <li>• Padronizar receitas, fórmulas matemáticas ou elementos da natureza</li> </ul>	<ul style="list-style-type: none"> <li>• Existe algum padrão entre as partes?</li> </ul>
Abstração	<ul style="list-style-type: none"> <li>• Filtrar das partes padronizadas: apenas aquilo que é relevante</li> <li>• Anular informações desnecessárias para a resolução de um problema</li> <li>• Facilitar o processo de solucionar uma questão levantada</li> </ul>	<ul style="list-style-type: none"> <li>• Elaborar calendário, perguntas, mapas, GPS, encontrar erros em conjunto de dados</li> </ul>	<ul style="list-style-type: none"> <li>• Há alguma informação desnecessária?</li> <li>• O que é mais importante para ser resolvido antes?</li> </ul>
Algoritmo	<ul style="list-style-type: none"> <li>• Tem diferentes níveis de complexidade</li> <li>• Usar métodos prontos para a solução de problemas</li> <li>• Desenvolver uma solução passo a passo, normalmente por meio de regras</li> </ul>	<ul style="list-style-type: none"> <li>• Executar operações aritméticas e lógicas</li> <li>• Elaborar sequencias de ações</li> </ul>	<ul style="list-style-type: none"> <li>• Existe um método ou regra já estabelecido que pode ser utilizado para resolver esse problema?</li> </ul>

Fonte: GUIMARÃES (2022)

Vicari *et al.* (2018) propõe:

“um conjunto de orientações para aplicar o PC em situações de ensino/aprendizagem do currículo escolar que sirva como um guia para que os professores busquem utilizar o PC nas suas disciplinas:

- Exploração de problemas, propondo situações em que os alunos desenvolvam algum tipo de estratégia para resolvê-las;
- Enunciado do problema, que exerce função importante, pois esse é fornecido ao aluno para que ele interprete e estruture a situação que lhe é apresentada para ser resolvida;
- Método de busca da solução, envolvendo a realização de aproximações sucessivas rumo a resolução do problema, utilizando o que o aluno já aprendeu para a resolução de outros problemas;
- Construção de um conceito para a definição de um problema pelo aluno, que também constrói conceitos articulados com outros que façam sentido num conjunto de problemas similares (que compartilham alguma etapa da sua solução)” (Vicari *et al.*, 2018, p.110-111).

Dentre os vários estudos sobre a integração do PC na Educação Básica, cabe apontar a pesquisa de Valente (2016), que com base em um levantamento com diferentes autores, define seis categorias de abordagens no ensino do PC no Ensino Fundamental, a saber:

- f) Atividades sem o uso das tecnologias: uso de atividades lúdicas “para mostrar às crianças o tipo de pensamento que é esperado por um cientista da computação” (VALENTE, 2016, p. 873);
- g) Programação *Scratch*: linguagem baseada em blocos visuais, com o intuito de facilitar o aprendizado de programação;
- h) Robótica pedagógica: aprendizado de programação baseado em dispositivos robóticos (objetos concretos);
- i) Produção de narrativas digitais: espirais de aprendizagem tendo como foco a produção de histórias, utilizando texto, imagens, animações, e outros recursos digitais;
- j) Criação de *games*: potencial de aprendizagem de programação integrada, explorando os vários elementos envolvidos no desenvolvimento de jogos digitais: estética, narrativa, mecânica e tecnologia;
- k) Uso de simulações: explorar conceitos e verificar o comportamento dos fenômenos mediante a alteração de variáveis em simuladores disponíveis em diversas áreas (tais como física, química e biologia).

## 2.2 PC e Ciência da Computação

A informática na educação é um tema que está consagrado na literatura, sendo tratado há décadas. Atualmente existe um uso intenso do computador e da informática ou de outros recursos tecnológicos no âmbito do ensino, visto que a escola é reflexo da sociedade. Entretanto foram surgindo demandas novas, que vão além do uso de editores de texto, *e-mails* e navegação na Internet, que se tornaram corriqueiros na realidade dos alunos. E, apesar de levantamentos bibliográficos apontarem bons resultados nos estudos realizados no âmbito do Ensino Fundamental, essa necessidade acontece também no Ensino Médio e no Ensino Superior. Nesse cenário, a Computação não é equivalente à Informática. O ensino de aplicativos de escritório é estante na educação básica, é como ensinar a usar calculadoras e não a calcular (FRANÇA *et al.*, 2014).

Enquanto a informática na educação envolve o ensino de Tecnologias da Informação, como aplicativos de escritório, o básico sobre sistemas operacionais e sistemas de Internet (navegadores e dispositivos de conexão e segurança) aos alunos da educação básica, a computação trata de conteúdos que ensinam os alunos a criar recursos tecnológicos através da programação e desenvolvimento de sistemas. Esses conteúdos envolvem um nível de complexi-

dade maior, pois trata de disciplinas relacionadas à Ciência da Computação (CC), que exigem conhecimentos aprofundados de lógica e matemática. O ensino de computação está relacionado ao processo de resolução de problemas, que pode ser associado ao processo de pensamento lógico e matemático (ARAÚJO *et al.*, 2015).

O interesse em ensinar computação na educação em geral é um tema que vem sendo tratado em vários locais no mundo. Em alguns países se utiliza um currículo mínimo para o ensino de CC, como acontece em Israel (GAL-EZER; HAREL, 1999), no Canadá (MEO, 2008 *apud* CUNHA *et al.*, 2019) e nos Estados Unidos, em que o modelo de currículo investe no ensino de conteúdos que explorem a chamada Educação Imperativa, na qual mais importante do que aprender temas ligados às tecnologias está a capacidade de desenvolver o PC (CSTA, 2020).

Entretanto, face às dificuldades de introduzir a CC como uma disciplina nos currículos escolares, muitas iniciativas têm inserido esse tema através de atividades extracurriculares. O trabalho de Futschek e Moschitz (2011) utiliza um cenário lúdico chamado de “Tim the Train” para trabalhar o conceito de abstração com estudantes em idade escolar que estão aprendendo programação, que proporciona a oportunidade de desenvolver capacidades úteis para resolver problemas mais avançados da CC.

No Brasil, o trabalho para inserir o PC e o ensino de programação e algoritmos nos currículos tem sido realizado pela Sociedade Brasileira de Computação (SBC). Esse trabalho é feito em forma de discussões para despertar o interesse dos alunos na disciplina, desenvolvendo assim as competências necessárias e, no futuro, contar com mais profissionais disponíveis para trabalhar nessa área (PEREIRA JUNIOR *et al.*, 2005).

As habilidades do PC não são iguais às de programação de computadores, mas ser capaz de programar é um benefício que ele desenvolve (SHUTE *et al.* 2017). As habilidades do PC e de programação, bem como resolução de problemas, estão intimamente relacionadas.

Conteúdos de computação ainda não fazem parte do currículo das escolas brasileiras, como apontado por Araújo *et al.* (2015). Porém já existem algumas iniciativas a respeito dessa prática, como as escolas de programação SuperGeek e MadCod, sediadas em São Paulo, e programas como Robótica na Escola, promovido em Recife. Outros exemplos são os trabalhos de Garcia (2008) e Barros (2009), que relatam ações de ensino de lógica de programação, visando a preparação dos estudantes para a Olimpíada Brasileira de Informática (OBI) nas escolas.

Segundo Araújo *et al.* (2015), é um equívoco dizer que o ensino da computação para-



os adolescentes deve ser o mesmo que ocorre nas aulas para os alunos universitários. Diferentemente da abordagem utilizada no Ensino Superior, o ensino da computação no Ensino Médio deve compreender técnicas para a resolução de problemas e o processo de raciocínio lógico-matemático. De acordo com Valente (2016):

a ênfase nos conceitos da Ciência da Computação tem sido justificada com vistas ao desenvolvimento de habilidades do pensamento crítico e computacional, permitindo um entendimento sobre como criar com as tecnologias digitais, e não simplesmente utilizá-las como ferramentas de expediente. Esses conhecimentos são considerados fundamentais para preparar as pessoas para o século XXI (VALENTE, 2016, p. 867).

### 2.3 PC no contexto educacional

O levantamento bibliográfico sobre PC no ensino básico resultou em vários estudos de caso, relatos de experiências, criações de jogos e aplicativos, análise de currículos escolares etc. Há diferentes enfoques sobre o tema, tais como:

- a) Uma Revisão Sistemática da Literatura Brasileira sobre o PC e o Ensino de Programação (ZANETTI *et al.*, 2016);
- b) Discussões sobre a importância em estimular o PC na Educação Básica com o uso de ferramentas lúdicas, como o Scratch (ZANETTI *et al.*, 2017);
- c) Criar um método de diagnóstico do nível de PC dos indivíduos, com a aplicação de vários testes (RAABE *et al.*, 2017);
- d) Relações entre o PC e a Matemática na Educação Básica (SILVA, 2019);
- e) Sondagem sobre a utilização de atividades durante as aulas, que estimulem o PC, por professores da Educação Básica (NUNES e KOLOGESKI, 2019);
- f) PC na formação de professores (REICHERT; BARONE & KIST, 2019);
- g) Uma estratégia para estimular o interesse em computação, em alunas do Ensino Médio, através de um projeto com robótica, devido ao baixo índice de mulheres na área (DAL BEN; SANTOS, 2020).

Outro exemplo de integração do PC no Ensino Fundamental, nesse caso com atividades desplugadas (sem a necessidade de usar qualquer equipamento eletrônico), é o trabalho de BRACKMANN (2017). Trata-se de um estudo piloto, aplicado em escolas brasileiras, resultado de sua tese de doutorado no PGIE/UFRGS (Pós-Graduação em Informática na Educação/ Universidade Federal do Rio Grande do Sul).

Um exemplo mais próximo da proposta deste trabalho, encontramos em Garlet *et al.* (2016, p. 1), que “buscou mostrar o quão importante é o ensino da Lógica de Programação desde o ensino básico, para que os alunos estejam preparados para o ensino superior.” A autora afirma que a busca por métodos que facilitem essa aprendizagem no Ensino Básico podem ser úteis e “propôs um método de ensino de lógica de programação que foi validado, por meio de um estudo de caso, realizado com alunos dos 7º, 8º e 9º anos em duas escolas de Ensino Fundamental da Região do Médio Alto Uruguai do Rio Grande do Sul” (GARLET *et al.*, 2016,p. 1). Os resultados obtidos indicam a eficácia do método por meio da utilização da ferramenta *VisuAlg*, utilizando-se desta para que fosse realizado o estudo da lógica de programação e conceitos da programação de computadores em três escolas da região do Alto Médio Uruguai, Rio Grande do Sul.

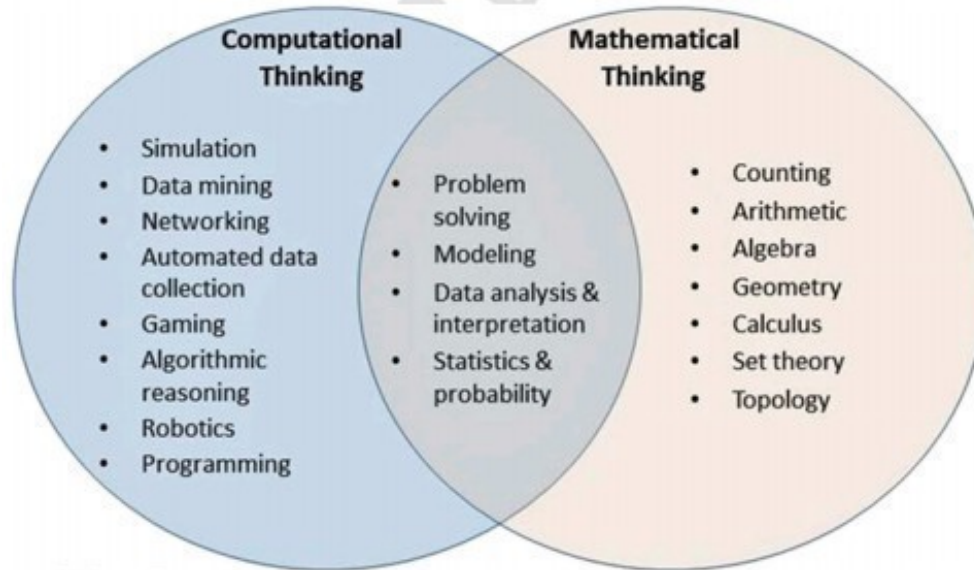
Sobre o desenvolvimento das habilidades relacionadas ao PC juntamente com a disciplina de Matemática, podemos citar o trabalho de Dalla Vecchia (2012), que propõe a abordagem de modelagem matemática com o uso de simuladores.

Os autores Shute, Sun e Asbell-Clarke (2017), em seu artigo “Desmistificando o Pensamento Computacional”, pesquisaram as diferenças e semelhanças entre o PC e outros tipos de pensamento e comparam o PC com matemática, engenharia, *design* e pensamento sistêmico. A Figura 2 mostra o conjunto completo de conceitos de PC e matemático: resolução de problemas, modelagem, análise e interpretação de dados, estatística e probabilidade.

A principal semelhança entre o PC e o pensamento matemático são os processos de resolução de problemas (Wing, 2008).

Estudos realizados por Barcelos e Silveira (2012) apontam que através da disciplina de Matemática existe uma estratégia para inserção do PC no ensino básico. Com o mapeamento das diretrizes presentes no Parâmetros Curriculares Nacionais (PCNs), que estavam relacionadas ao PC, abordaram o impacto da tecnologia no ensino da Matemática. Com isso, identificaram três competências que podem ser trabalhadas: articulação dos símbolos e códigos; estabelecer relações e identificação de regularidades; e modelos explicativos e representativos (BARCELOS; SILVEIRA, 2012).

Figura 2 - Semelhanças e diferenças entre pensamento matemático e PC



Fonte: SHUTE, SUN e ASBELL-CLARKE (2017)

Tratando-se das escolas públicas brasileiras, a inclusão do PC como conteúdo curricular parece ousada, porém um passo importante foi dado nessa direção com a inserção do termo PC no principal documento norteador dos currículos escolares em todo o país, a Base Nacional Comum Curricular (BNCC).

A BNCC é o documento que define o conjunto de aprendizagens essenciais que escolas públicas e privadas do país devem promover. Entre as dez competências gerais da Educação Básica citadas no documento, a quinta, sobre Cultura Digital, diz o seguinte:

compreender, utilizar e criar tecnologias digitais de informação e comunicação de forma crítica, significativa, reflexiva e ética nas diversas práticas sociais para se comunicar, acessar e disseminar informações, produzir conhecimentos, resolver problemas e exercer protagonismo e autoria na vida pessoal e coletiva (BRASIL, 2017, p. 9).

O PC é mencionado na BNCC como sendo uma habilidade específica na área de Matemática. Esse documento estabelece que, no Ensino Médio, o estudante deve ser capaz de “utilizar os conceitos básicos de uma linguagem de programação na implementação de algoritmos escritos em linguagem corrente e/ou matemática” (BRASIL, 2017).

Ainda nesse contexto, a atualização das Diretrizes Nacionais para o Ensino Médio, promulgada pela Lei Federal n.º 13.415/2017, chamada de Novo Ensino Médio, garante que as linguagens digitais e o PC estejam na proposta do Ensino Médio. Essas seriam abordadas em dois aspectos apontadas no art.12 da lei, sendo

(...) II - 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;

(...) V - formação técnica e profissional: desenvolvimento de programas educacionais inovadores e atualizados que promovam efetivamente a qualificação profissional dos estudantes para o mundo do trabalho, objetivando sua habilitação profissional tanto para o desenvolvimento de vida e carreira, quanto para adaptar-se às novas condições ocupacionais e às exigências do mundo do trabalho contemporâneo e suas contínuas transformações, em condições de competitividade, produtividade e inovação, considerando o contexto local e as possibilidades de oferta pelos sistemas de ensino (BRASIL, 2017, p.478).

Silva (2019) realizou “a sistematização da relação entre as habilidades das duas áreas com o objetivo de nortear educadores e escolas na elaboração de seus currículos, servindo de referencial, apontando em quais habilidades da BNCC é possível inserir as habilidades do PC” (SILVA, 2019, página 70). Com as mudanças sociais, culturais, e tecnológicas faz-se necessário que o educador também mude sua prática em sala, utilizando a maior gama possível de recursos.

No entanto, como já citado, PC consiste em uma metodologia de resolução de problemas; portanto, faz-se necessário valer-se de uma metodologia de ensino que apoie esse processo. Com base nas experiências de ensino envolvendo PC, bem como no formato das atividades das plataformas avaliadas, é possível identificar a gamificação como uma metodologia de ensino.

## 2.4 Gamificação

A gamificação, denominação sugerida por Werbach e Hunter (2012), consiste na utilização de elementos dos *games* (mecânicas, estratégias, pensamentos) com a finalidade de auxiliar na solução de problemas e promover aprendizagens (FARDO, 2013 *apud* KAPP, 2012). Teve origem como método aplicado em programas de *marketing* e aplicações para *web*, com a finalidade de motivar, engajar e fidelizar clientes e usuários (ZICHERMANN; CUNNINGHAM, 2012).

A gamificação emerge em meio às tradicionais abordagens do processo de ensino e de aprendizagem, às diversas possibilidades de se ampliar as ações pedagógicas em sala de aula, dado o contexto social e tecnológico, e os novos hábitos e práticas sociais. O conceito de gamificação na educação, existe desde os anos de 1970, quando era associado a programação e desenvolvimento de softwares.

A gamificação também surgiu devido a apropriação dos “elementos dos jogos aplicados em contextos, produtos e serviços necessariamente não focados em jogos, mas com a intenção de promover a motivação e o comportamento do indivíduo” (BUSARELLO; ULBRICHT; FADEL, 2014, p. 14). A gamificação pressupõe:

“a utilização de elementos tradicionalmente encontrados nos games, como narrativa, sistema de feedback, sistema de recompensas, conflito, cooperação, competição, objetivos e regras claras, níveis, tentativa e erro, diversão, interação, interatividade, entre outros, em outras atividades que não são diretamente associadas aos games, com a finalidade de tentar obter o mesmo grau de envolvimento e motivação que normalmente encontramos nos jogadores quando em interação com bons games” (FARDO, 2013, p. 2).

Atualmente, a gamificação encontra nas instituições de ensino um público familiarizado com a utilização dos *games*. Encontra também um contexto educacional, que anseia por novas estratégias para sensibilizar indivíduos que estão mais inseridos no contexto das tecnologias digitais e se mostram desinteressados pelos métodos tradicionais de ensino e aprendizagem utilizados na maioria das escolas (FARDO, 2013). Fardo (2013), em seus estudos, cita linhas gerais para a aplicação da gamificação em sala de aula, dentre as quais se destacam:

- a) *Feedback*: nos *games* os jogadores são sempre capazes de visualizar o efeito de suas ações em tempo real. Nas escolas normalmente os alunos só conseguem visualizar seus resultados depois de certo tempo, maior do que aquele que estão acostumados nos *games*. Acelerar esse processo de *feedback* estimula a procura por novos caminhos para atingir os objetivos, bem como o redirecionamento de uma estratégia, caso ela não esteja apresentando os resultados esperados;
- b) Aumentar a dificuldade das tarefas conforme a habilidade dos alunos: proporcionar diferentes níveis de dificuldade para os desafios propostos pode auxiliar na construção um senso de crescimento e avanço pessoal nos estudantes, e também faz com que cada um siga o seu próprio ritmo de aprendizagem;
- c) Decomposição: problemas são divididos em uma série de outros menores e mais fáceis de serem superados. Dessa forma, o estudante vai construindo seu conhecimento de forma gradual, observando as partes do problema e de que modo elas se relacionam com o todo, proporcionando maior motivação e preparo para superar o desafio maior inicial;
- d) Depuração: o erro faz parte dos *games* de forma natural. Nenhum jogador espera interagir com um *game* sem se deparar com a falha várias vezes. Incluir e aceitar o erro como parte do processo de aprendizagem e estimular a reflexão dos motivos desses erros faz parte de um processo semelhante ao que ocorre nos games;
- e) Narrativa: nos *games* normalmente temos uma história que justifica o porquê dos personagens estarem fazendo aquilo. Construir um contexto para a aprendizagem pode fornecer um bom motivo para os estudantes empenharem suas energias em aprender (FARDO, 2013).

Nesse contexto, o professor exerce um papel vital ao tornar-se o responsável por

acompanhar todas as atividades realizadas pelos seus alunos, observando e avaliando seus níveis de desenvolvimento e identificando as dificuldades individuais e coletivas, além de moderar o entretenimento oferecido com a aprendizagem necessária (PONTES, 2013).

Para Pontes (2013), o emprego das estratégias dos jogos serve como catalizador do processo de ensino pelo docente, pois propicia o desenvolvimento da autonomia e da confiança do aluno de forma espontânea, despertando habilidades como planejamento, seleção, organização, interpretação, decisão e socialização.

Paralelamente aos pilares do PC, uma abordagem de gamificação ajuda na formação de diversas habilidades cognitivas aplicáveis em diversas ciências do conhecimento além da Computação, como a decomposição de problemas, a interligação de elementos e a capacidade de abstração (PONTES, 2013). Desta forma, a gamificação pode ser utilizada para desenvolver estratégias para acelerar o pensamento dos alunos.

## **CAPÍTULO III**

### **O OBJETO DE ESTUDO E O ARCABOUÇO METODOLÓGICO**

Conforme descrição apresentada na Introdução, esse trabalho objetiva investigar uma prática alternativa visando ao desenvolvimento do Pensamento Computacional durante as aulas de Lógica Computacional do curso Técnico em Informática, considerando como ferramenta tecnológica a plataforma *Programaê!*. Assim, as seções seguintes apresentam os fundamentos metodológicos da pesquisa de campo, o perfil dos participantes, o ambiente e o formato da experiência de ensino vivenciada. Na sequência o texto apresenta a plataforma digital utilizada para a condução das atividades. A penúltima seção trata do roteiro das atividades, incluindo os critérios para a classificação das atividades desenvolvidas e, por último, são descritos os instrumentos adotados para a coleta e tratamento das informações.

#### **3.1 Natureza da pesquisa**

Quanto à natureza da investigação, a pesquisa descrita neste trabalho caracteriza-se como qualitativa e interpretativa. “Isso significa que os dados recolhidos serão ricos em pormenores descritivos relativamente a pessoas, locais e conversas e testam hipóteses com o objetivo de investigar os fenômenos em toda sua complexidade e em contexto natural” (BOGDAN; BIKLEN, 1994)., com a intenção de explorar uma metodologia ou ferramenta e adotar procedimentos que contribuem com a aprendizagem dos alunos

A pesquisa qualitativa envolve uma gama de técnicas e procedimentos interpretativos, que procuram descrever, decodificar, traduzir, construir e analisar o sentido e o significado para as pessoas; e não apenas a frequência de eventos ou fenômenos do mundo social (MERRIAM; TISDELL, 2015, 2016 *apud* GOMES, 2019). Trata-se de uma atividade situada de pesquisa, que localiza o observador no mundo e define a subjetividade deste como perspectiva epistemológica (GOMES, 2019, p. 6).

A pesquisa qualitativa apresenta características essenciais como: ter um ambiente natural como fonte direta de dados; o pesquisador como ferramenta fundamental de construção e análise dos dados; é descritiva; identifica o significado que as pessoas dão às

coisas e o pesquisador utiliza o enfoque indutivo na análise de seus dados (GOMES, 2019, p.6).

Para Gomes (2019), a pesquisa qualitativa envolve a obtenção de dados descritivos sobre pessoas, lugares e processos interativos pelo contato direto do pesquisador com a situação estudada, procurando compreender os fenômenos segundo a perspectiva dos sujeitos na situação de estudo. A profundidade das análises qualitativas pode ser muito variada, indo desde uma descrição simples do que é observado até sofisticadas relações dialéticas e críticas.

### **3.2 O Campo da Pesquisa**

Atualmente existem quatro documentos importantes que normatizam a Educação Profissional no Brasil, sendo o principal deles a Lei de Diretrizes e Bases da Educação Nacional (LDB), as Diretrizes Curriculares Nacionais (DCN), a Estadual e o documento base da Educação Profissional Técnica de Nível Médio Integrado ao Ensino Médio, elaborado pelo Ministério da Educação e Cultura (MEC).

Conforme as DCN para a Educação Profissional Técnica de Nível Médio, aprovada em 2012, em seu Capítulo II, dispõem sobre a Organização Curricular, no Art. 12, traz que os cursos de Educação Profissional Técnica de Nível Médio são organizados por eixos tecnológicos constantes do Catálogo Nacional de Cursos Técnicos, instituído e organizado pelo MEC ou em uma ou mais ocupações da Classificação Brasileira de Ocupações (CBO). No caso do curso Técnico em Informática o eixo é Informação e Comunicação e sua descrição está disponível no catálogo de cursos da Secretaria de Estado da Educação do Paraná (SEED) (PARANÁ, 2013).

No texto das Diretrizes Estaduais da Educação Profissional aparece a revogação do Decreto n. 2.208/97 e a promulgação do Decreto n. 5.154/04 que possibilitou conceber propostas curriculares considerando:

a necessária articulação entre as diferentes dimensões do trabalho de formação profissional do cidadão/aluno, na perspectiva da oferta pública da Educação Profissional técnica de nível médio, enfatizando o trabalho, a cultura, a ciência e a tecnologia, como princípios fundantes da organização curricular integrada ao Ensino Médio (PARANÁ, 2006, p.18)

Os cursos organizados na forma integrada, articulados com o Ensino Médio, são cursos de matrícula única, que conduzem os educandos à habilitação profissional técnica de nível médio ao mesmo tempo em que concluem a última etapa da Educação Básica.



Atualmente a SEED dispõe de duas formas de oferta: Integrada e Subsequente.

O Curso Integrado tem o objetivo de propiciar autonomia intelectual, de tal forma que, a cada mudança científica e tecnológica, o cidadão consiga por si próprio formar-se ou buscar a formação necessária para o seu desenvolvimento profissional.

Redefinir as finalidades e os projetos de educação dos trabalhadores de modo a contemplar novas prioridades e alternativas que impactem as suas condições de trabalho e de existência. Ao invés de profissionais rígidos, competentes nos fazeres que se repetem pela memorização, há que formar profissionais flexíveis, que acompanhem as mudanças tecnológicas decorrentes da dinamicidade da produção científico-tecnológica contemporânea (PARANÁ, 2006, p. 26).

Sobre o Ensino Médio Integrado define-se ser um espaço de processo de travessia, concebida como o momento de assegurar ao jovem o direito de conclusão da escolaridade média numa relação imediata com o trabalho produtivo “de tal forma que lhe seja garantida uma Educação Básica de qualidade como direito – que poderá vir a contribuir para sua inserção numa área técnica ou tecnológica específica e garantir a sua subsistência” (PARANÁ, 2006, p. 41).

### **3.3 O local e os participantes da pesquisa**

O curso Técnico em Informática, integrado ao nível médio é ofertado no Colégio Estadual Francisco Carneiro Martins, na cidade de Guarapuava, PR. A instituição está localizada na região central da cidade e possui uma estrutura física que conta com dois laboratórios de informática equipados com computadores de mesa e com acesso à Internet, com capacidade para 30 alunos.

O ambiente de investigação aconteceu com a turma da 1ª série do curso técnico em Informática, contando com a participação de estudantes com faixa etária entre 14 e 15 anos. Não há requisito para ingresso no curso, ou seja, os alunos não necessitam ter conhecimento prévio em informática ou programação de computadores, pois o Colégio não realiza meios formais de seleção de candidatos. É realizada uma entrevista com o objetivo de identificar interesses e afinidades compatíveis com o perfil de formação proposto no curso e é considerada a situação sócio-econômica do aluno.

A experiência de ensino foi realizada durante o período letivo no ano de 2021, no contraturno, por meio de uma proposição de complemento aos conteúdos curriculares a serem inseridos nos planos de trabalho docente. A execução das atividades ocorreu no período da manhã, entre os meses de setembro e outubro de 2021. Nesse período as aulas presenciais

havam retornado, mas a presença no colégio não era obrigatória, apenas com a autorização dos pais. Foram cinco encontros síncronos realizados no ambiente virtual *Google Meet*, totalizando uma carga horária de 30 horas.

A experiência assumiu um caráter prático, com a utilização do computador pessoal dos estudantes, e a realização de atividades de natureza exploratória, em um ambiente de cooperação e colaboração entre os discentes e a docente. Esta, por sua vez, autora deste texto, desempenhou seu papel em consonância com a disciplina Prática Docente Supervisionada do Programa de Pós- Graduação em Ensino de Ciências Naturais e Matemática, PPGEN, da UNICENTRO.

A presente proposta passou pela avaliação e aprovação dos Conselhos de Ética da Universidade, parecer nº 4.470.975 e da Secretaria de Estado da Educação, protocolo nº 16.772.220-2, através de trâmites legais, cuja realização foi autorizada pela Direção do Colégio. A identidade dos alunos foi preservada, incluindo imagens e diálogos. A participação foi voluntária, sendo que os alunos estavam cientes da natureza do estudo.

### **3.4 Programação em blocos**

Brennan e Resnick (2012) consideram a programação em blocos com mídia amigável e interativa um contexto poderoso para o desenvolvimento de habilidades do PC. Na pesquisa envolvendo a interação dos jovens com o Scratch, os autores notaram que os jovens entrevistados adotaram uma variedade de estratégias e práticas para resolver as atividades propostas. Com base nisso, consideram que o PC envolve três dimensões:

- Conceitos computacionais: os elementos de programação que desenvolvedores empregam enquanto programam;
- Práticas: as práticas que desenvolvedores utilizam à medida que programam;
- Perspectivas: as perspectivas que os desenvolvedores formam sobre o mundo ao seu redor e sobre eles mesmos.

Segundo Brennan e Resnick (2012), são sete os conceitos de programação: seqüências, loops, paralelismo, eventos, condicionais, operadores e dados.

Quanto as práticas, os autores observaram os seguintes conjuntos:

- Incremental e iterativo: processo adaptativo, não linear, composto por ciclos iterativos. A solução vai sendo refinada; trata-se do processo em que o plano pode mudar em resposta à abordagem de uma solução em pequenas etapas;
- Teste e depuração: busca da solução por tentativa e erro;
- Reutilização e remixagem: a prática consiste em empregar artefatos já conhecidos (não necessariamente código pronto) para resolver novos problemas. Ajuda programadores a encontrar ideias e código, permitindo criar soluções potencialmente muito mais complexas do que poderiam ter criado por conta própria;
- Abstração e modularização: construir algo grande, juntando coleções de partes menores. Todo o processo de projeto e resolução do problema, com o foco no enunciado.

As perspectivas tratam da capacidade de expressão, de conexão social e de análise crítica, analisadas nos estudantes. Ao usar essa plataforma os aprendizes seguem os desafios, da mesma forma que nos jogos; a prática de depuração fica evidente quando o aluno retorna a um estágio anterior para refazer uma atividade. Sendo assim, a gamificação torna-se uma característica própria da ferramenta. Ao ensinar programação da forma tradicional, a programação estruturada, o professor usa a forma escrita, mas por meio das diversas plataformas disponíveis atualmente para auxiliar no ensino de programação, os alunos utilizam o “Blockly”, que são blocos visuais que servem para arrastar e soltar comandos na tela do computador, para escrever programas. E por trás desses blocos, o aluno está criando códigos. Alguns exemplos dessas plataformas são o Code.org e o Scratch. A interface das atividades do Code.org é semelhante à interface do Scratch, que é outra plataforma onde pode-se aprender programação através de blocos. O Scratch é um projeto do *Lifelong Kindergarten Group* do MIT Media Lab também disponibilizado gratuitamente. Porém, é na plataforma Programaê! em que estão disponíveis diferentes iniciativas de incentivo ao ensino de programação em língua portuguesa.

### **3.5 A plataforma *Programaê!***

A plataforma *Programaê!* foi desenvolvida por meio de uma parceria entre a

Fundação Telefônica Vivo e a Fundação *Lemann*. É um projeto de disseminação do conhecimento de Ciência da Computação e programação para professores, jovens e crianças, que busca prepará-los para o desenvolvimento das competências essenciais para futuros profissionais do século XXI. Nesse sentido, criou-se uma plataforma que agrega cursos e conteúdos gratuitos sobre programação desenvolvidos por parceiros internacionais, como *Scratch*, *Khan Academy*, *Code* e *Codecademy*.

A proposta tem como propósito principal desenvolver estratégias para introduzir a linguagem de programação e o PC de forma mais intensa nas práticas pedagógicas, garantindo subsídios para que os professores e estudantes sejam protagonistas desse processo, também para auxiliar redes de ensino e escolas a desenvolverem um currículo que contemple o PC, em consonância com a Base Nacional Comum Curricular (BNCC) (PROGRAMAÊ!, 2018).

A plataforma *Programaê!* conta com duas formas para ensino, uma para discentes ‘Quero aprender’ e outra para docentes ‘Quero ensinar’. No módulo ‘Quero ensinar’, destinado aos professores, há exemplos de como elaborar atividades, códigos e desenvolvimento de aplicativos. No módulo ‘Quero aprender’, o aluno realiza as atividades em forma de trilhas. As trilhas são os caminhos a serem percorridos para aprender a programar, e assim desenvolver a lógica de programação (PROGRAMAÊ!, 2018). Além disso, a plataforma disponibiliza um Guia com mais de duzentas páginas, com conteúdo sobre a importância de estudos como este e, em eixos, discute elementos significativos para a implantação da cultura digital e do PC nas escolas brasileiras. Os seis eixos são: Políticas Públicas, Infraestrutura, Gestão Escolar, Currículo, Formação de Professores e Aluno (PROGRAMAÊ!, 2018). Cada um destes temas apresenta informações, reflexões, práticas e dicas que foram organizadas em pequenos blocos: linha do tempo, contexto, evolução, agora, ideia (sugere práticas) e casos (exemplos de usos no Brasil e no mundo).

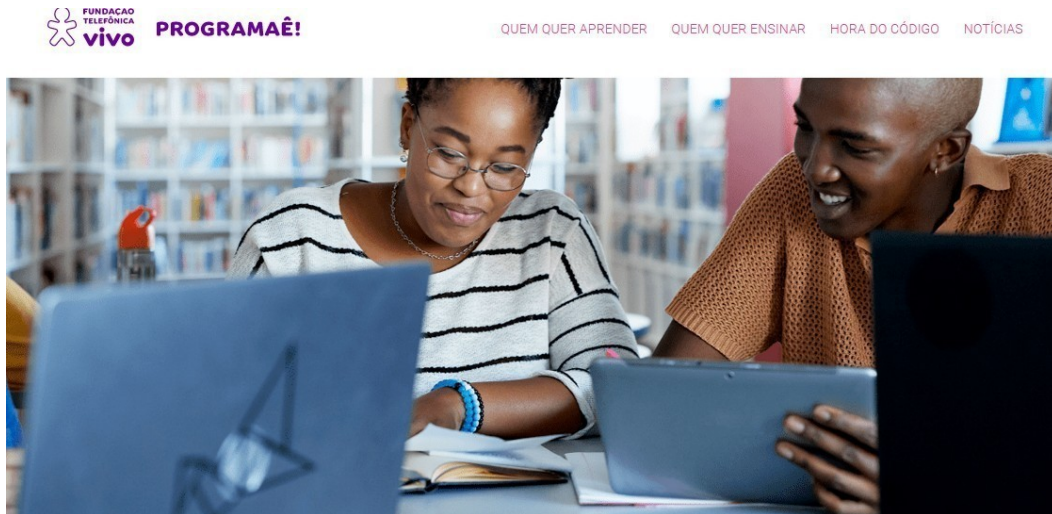
As atividades disponibilizadas pela plataforma, cuja página inicial consta na Figura 3, fazem parte de um evento denominado “A Hora do Código” criado pela Code.org, e são ideais para uso de forma isolada e que não necessitam de uma sequência preestabelecida. O movimento faz parte de uma ação global que busca mostrar que alguns conceitos da programação estão ao alcance de todos e que aprendê-los é mais fácil do que se imagina. Para começar, a ideia é que alunos e professores programem juntos, por ao menos uma hora, durante a semana do evento<sup>3</sup>.

---

3 [www.programae.org.br/horadocodigo](http://www.programae.org.br/horadocodigo), acessado em 05 de dezembro de 2021.

Valorizar e aplicar esse material nas aulas da disciplina de Lógica Computacional, com os alunos do curso Técnico em Informática, vêm ao encontro de um ensino de programação que tenha significado para o estudante, para a sua aprendizagem, ou seja, lança o olhar para estratégias de ensino e não apenas para um resultado.

*Figura 3: Página inicial do Programaê!*



Fonte: [programae.org.br](http://programae.org.br)

### 3.6 O Roteiro de atividades

Considerando que a plataforma Programaê! disponibiliza atividades sem um critério de níveis de dificuldade, ou trilhas como a plataforma Code.org, dez atividades disponíveis na plataforma Programaê! foram selecionadas, a fim de constituir um roteiro introdutório de apoio à disciplina de Lógica Computacional, com foco no desenvolvimento do PC.

A sequência das atividades foi definida conforme os conceitos computacionais envolvidos. Para a classificação das atividades do roteiro aqui proposto, foram considerados os seguintes conceitos (também chamados elementos) de programação:

- Programação sequencial: uma determinada atividade ou tarefa é expressa como uma série de etapas ou instruções individuais que devem ser realizadas em sequência. Como uma receita, uma sequência de instruções de programação especifica o comportamento ou ação que deve ser produzida (BRENNAN; RESNICK, 2012);
- Eventos: são gatilhos para que certas ações aconteçam para criar ambientes interativos como o uso de teclas, uso do mouse, uso de ambos os dispositivos

(SHUTE *et al.*, 2017);

- Estruturas de repetição: também conhecidas como laços (ou *loops*), efetuam o processamento de um bloco de instruções quantas vezes forem necessárias. Uma estrutura de repetição pode ser usada de forma isolada (uma única estrutura) ou combinada, sendo várias estruturas em sequência (estruturas alinhadas) ou encaixadas (estruturas aninhadas) (MANZANO; OLIVEIRA, 2012);
- Estruturas de seleção: esse tipo de instrução dentro de um programa tem por finalidade tomar uma decisão, por meio da avaliação de uma condição. Também chamadas de estruturas de controle, podem ser usadas de forma isolada ou em conjunto, da mesma formas que as estruturas de repetição (em sequência ou encaixadas) (MANZANO; OLIVEIRA, 2012);
- Funções: são conjuntos de instruções identificados por um nome. Este nome pode ser usado em qualquer outra parte do programa. É um bloco de instruções que pode ser reaproveitado e pode retornar um valor (numérico, lógico ou literal), entre outros tipos (MANZANO; OLIVEIRA, 2012).

Os elementos de programação são apresentados nessa ordem nas atividades da plataforma Code.org. Pode-se, então, considerar que eventos são elementos mais simples, com menor nível de dificuldade do que funções, que, quando são incluídas, estão nos desafios finais das atividades. O mesmo acontece com as estruturas de repetição e de seleção, em que aquelas aparecem antes que essas. Antes de introduzir esses elementos nas atividades, um vídeo explicativo (tutorial) anuncia o que significa e como determinado elemento deve ser utilizado, com exemplos contextualizados. No Quadro 2 está apresentado o roteiro de atividades. Cada atividade é formada por várias trilhas (ou desafios), variando de oito a vinte trilhas. Os números apresentados no quadro correspondem a quantidade de trilhas para cada conceito computacional (na respectiva coluna). Para cada trilha é apresentado um enunciado, com o objetivo que deve ser cumprido. No início da atividade, há vídeos tutoriais que orientam os aprendizes tanto quanto ao uso da plataforma e quanto aos conceitos de programação explorados nos próximos desafios. Em cada desafio, há dicas que contribuem para o cumprimento do objetivo proposto, pois os cenários, embora intuitivos, possuem

particularidades que podem aumentar a complexidade da tarefa (como esse é um aspecto subjetivo, não foi considerado nessa classificação). Três das atividades selecionadas contêm um desafio final (em geral um jogo), que pode ser compartilhado por meio de mídias sociais ou enviado para um celular. A complexidade dos desafios aumenta conforme a evolução da atividade.

Quadro 2 - Roteiro de atividades da plataforma Programaê!

Atividade \ Elemento de programação	Programação sequencial	Evento	Repetição	Seleção	Função	Total de trilhas*
1 Infinity Play Lab	2	5	3	-	-	10
2 Flappy Bird	-	10	-	-	-	10
3 Design Minecraft	1	6	2	-	-	12
4 Basquete NBA	-	8	-	-	-	8
5 Aventureiro Minecraft	4	-	6	4	-	14
6 Moana	5	-	8	5	-	19
7 Angry Birds	5	-	8	7	-	20
8 O Artista	3	-	4	-	3	10
9 Jornada do Herói Minecraft	2	-	5	-	4	12
10 Frozen	3	-	10	-	7	20

\*A soma dos desafios não corresponde ao total, pois há desafios finais que não se classificam nesses elementos

Fonte: Elaborado pela autora (2021)

As primeiras atividades elencadas foram aquelas com o maior número de desafios envolvendo o elemento evento (por ser mais intuitivo e simples). As estruturas de repetição variam entre repetir até um limite dado, repetir até que aconteça algo ou repetir infinitamente, sendo que em alguns desafios essas estruturas devem ser aninhadas ou utilizadas em conjunto com estruturas de seleção. Alguns desafios, como o ilustrado na Figura 4, exploram conhecimentos matemáticos quando exigem a definição de graus nos movimentos para criação de figuras geométricas; utilizam também o conceito de pixel e estruturas de repetição aninhadas. Quanto às funções, os desafios mais simples envolvem a utilização de funções prontas, chegando a desafios mais complexos que requerem que as funções sejam definidas. Na experiência de ensino o estudante explora o uso de funções prontas, uso de funções com

parâmetros, definição de funções e desafios que não necessariamente requerem, mas podem ser resolvidos com a definição de funções.

Figura 4 – Frozen: Trilha 11



Fonte: adaptado de programae.org.br.  
Acesso em janeiro de 2022

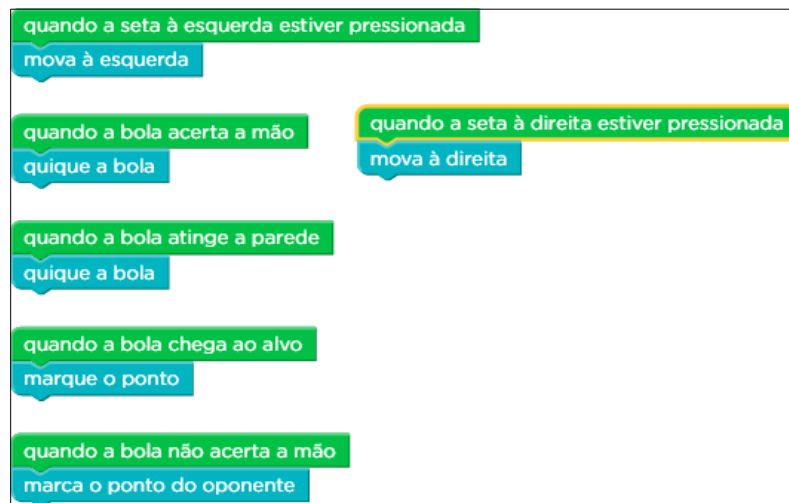
Quanto às contribuições do roteiro para o desenvolvimento das habilidades do PC, foi possível delinear algumas relações, tanto de forma mais geral (resolução de problemas a partir dos enunciados dos desafios) como mais específica (uso dos conceitos de programação):

- Decomposição: resolver o problema em partes, por exemplo, ao separar os comandos de eventos em direções (esquerda, direita, cima, baixo), decompondo a interação. Um exemplo prático encontra-se na Figura 5 sobre a trilha 6 da atividade Basquete NBA;
- Reconhecimento de padrões: identificar blocos que se repetem e utilizar uma estrutura de repetição ou uma função, por exemplo, quando são agrupados blocos para desenhar figuras, como nas trilhas 4 e 5 da atividade Frozen (Figuras 6 e 7);
- Abstração: esse pilar é transversal, aparece em todas as atividades. Entre as atividades propostas existe uma que se destaca por exigir com frequência o uso do mouse e estar atento aos movimentos do personagem na tela. Consiste em fazer com que o pássaro, chamado *Flappy Bird*, desvie os obstáculos e cumpra os desafios propostos para concluir e completar o jogo. Em vários momentos, em trilhas diferentes, exige-se fazer apenas o que se pede nas instruções, sem



que isso faça sentido para todo o jogo em si. Na trilha 5, ilustrada na Figura 8, o passarinho pode cair no chão, pode bater nos obstáculos que nada acontece. Esse é um exemplo de atividade que apresenta indícios do pilar abstração, pois é necessário abstrair a informação mais relevante do enunciado e focar no resultado;

*Figura 5 - Basquete NBA: Trilha 6 (Área de trabalho)*



Fonte: adaptado de [programae.org.br](http://programae.org.br) Acesso em janeiro de 2022.

*Figura 6 - Frozen: Trilha 4 (Bloco repita)*



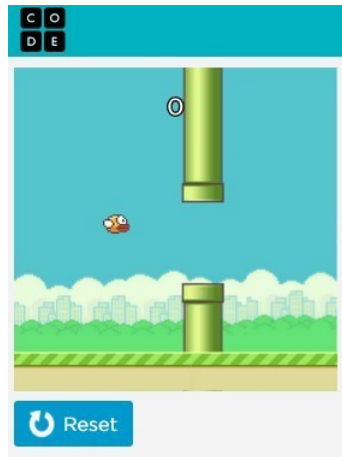
Fonte: [programae.org.br](http://programae.org.br) Acesso em janeiro de 2022.

*Figura 7 – Frozen: Trilha 5 (Bloco repita)*



Fonte: [programae.org.br](http://programae.org.br) Acesso em janeiro de 2022.

Figura 8 - Flappy Bird: Trilha 5



Fonte: [programae.org.br](http://programae.org.br) Acesso em janeiro de 2022.

- Algoritmo: utilizando o encaixe sequencial dos blocos, algoritmo consiste em construir a solução usando os elementos de programação (blocos). As figuras que ilustram os pilares anteriores, como por exemplo as Figuras 5, 6 e 7, são exemplos de algoritmos.

Para as atividades que constituem o Quadro 2 com o roteiro criado para aplicação da pesquisa, a habilidade de construção de algoritmos é a mais perceptível por definição, enquanto a abstração é a mais rara nesse tipo de situação, pois precisaríamos de um número maior de atividades envolvendo casos semelhantes a fim de organizar e classificar. Cabe ressaltar que essas relações ocorrem em outras atividades das plataformas *Programaê!* e *Code.org* que seguem o mesmo padrão: resolver um desafio, conforme o enunciado, utilizando programação em blocos.

### 3.7 Coleta e Análise dos dados

A coleta de dados ocorreu com base nos princípios da observação participante e na composição de um diário de bordo, considerando-se como instrumentos de coleta de informações: vídeos com captura dos momentos através da gravação dos encontros e produção escrita dos estudantes através de formulários aplicados após cada encontro. Seguindo as diretrizes da metodologia de análise de conteúdo, a pré-análise se configurou na preparação do material, que resultou no Quadro 1 sobre os pilares do PC (apresentado no Capítulo 1) e a segunda fase (exploração do material), se constituiu na elaboração do roteiro

de atividades da plataforma Programaê!, apresentado no Quadro 2 (que consta no Capítulo 2) bem como na definição das categorias de análise.

Na terceira fase da metodologia de análise de conteúdo, ou seja, no tratamento de resultados, as práticas de Bresnann e Resnick (2012) foram sintetizadas e adaptadas com a contribuição da pesquisa de Ponte (2020), e estão apresentadas no Quadro 3. As contribuições de Ponte (2020) apresentadas em seu artigo “Como desenvolver o raciocínio matemático em sala de aula”, traz um quadro para a análise dos processos essenciais do raciocínio matemático: conjecturar, generalizar e justificar e tabelou esses elementos, na sua forma e base. O modelo de Ponte (2020) tem por base:

o processo de realização de uma investigação ou resolução de um problema matemático, começando pela formulação de questões, passando para a formulação de conjecturas e estratégias de resolução (que podem envolver generalização), aplicando essas estratégias e testando as conjecturas, até ao processo de validação (através da justificação). A aula em três fases permite pôr em prática a abordagem de ensino exploratório (PONTE, 2020).

Assim, o diálogo estabelecido entre estratégias relacionadas ao raciocínio matemático e as práticas e pilares do PC, respectivamente de Ponte (2020) e Brennan e Resnick (2012), serviram de aporte teórico na compreensão dos dados, apresentado no Quadro 3.

*Quadro 3 – Estratégias do raciocínio matemático e Práticas do PC para análise dos dados*

<b>Estratégia</b>	<b>Prática</b>	<b>Pilar</b>
<ul style="list-style-type: none"> <li>• Formular uma estratégia para resolver um problema</li> <li>• Adaptar o processo</li> <li>• Mudar o plano em resposta à abordagem de uma solução em pequenas etapas</li> </ul>	Incremental e Iterativo	Decomposição
<ul style="list-style-type: none"> <li>• Conceituar o problema</li> <li>• Usar exemplos genéricos</li> <li>• Abstrair a informação mais relevante do enunciado</li> <li>• Focar no resultado</li> </ul>	Abstração e Modularização	Abstração
<ul style="list-style-type: none"> <li>• Reconhecer um padrão ou uma propriedade comum a um conjunto de objetos</li> <li>• Usar contraexemplos</li> </ul>	Reutilização e Remixagem	Reconhecimento de Padrões
<ul style="list-style-type: none"> <li>• Buscar a solução por tentativa e erro</li> <li>• Sequenciar instruções</li> <li>• Organizar uma ação que deve ser produzida</li> </ul>	Teste e Depuração	Algoritmo

Fonte: Adaptado de Bresnann e Resnick (2012) e Ponte (2012).

Na primeira coluna do Quadro 3 estão as possíveis estratégias utilizadas pelos

alunos para a resolução de problemas, com base nos pilares do PC. Exemplos dessas estratégias são: mudar o plano em resposta à abordagem de uma solução em pequenas etapas, abstrair a informação mais relevante do enunciado, reconhecer um padrão ou uma propriedade comum a um conjunto de objetos e buscar a solução por tentativa e erro. Na segunda coluna estão as práticas relacionadas com cada pilar: modularização, incremental, iteração, reutilização, remixagem, abstração e depuração. As práticas podem ser avaliadas como um todo, em qualquer atividade da plataforma. Na terceira coluna estão os pilares do PC, também chamados de dimensões: decomposição, abstração, reconhecimento de padrões e algoritmos.

Norteados pela metodologia de análise de conteúdo, o capítulo seguinte apresenta, de modo prático e objetivo, a produção de inferências no conteúdo da comunicação de um texto aplicável ao contexto de exploração do material elaborado, bem como o tratamento e a interpretação dos dados obtidos. A partir do uso dessa metodologia, percebemos o quão ela facilitou o desvendar dos objetivos da pesquisa, que foi aplicada aos discursos contidos nos textos, e mostrando o não aparente (FRANCISCO *et al.*, 2021, p.5). Para tanto foi realizada a articulação da experiência de ensino, através da plataforma Programaê!, com o desenvolvimento de habilidades de PC, descrevendo os processos de construção e o engajamento dos alunos ao criar seus projetos.

## **CAPÍTULO IV**

### **A EXPERIÊNCIA DE ENSINO ENVOLVENDO PC**

Consideramos que a experiência vivenciada com programação em blocos utilizando-se da plataforma Programaê! fornece um contexto e um conjunto de oportunidades potenciais para respondermos a seguinte pergunta de pesquisa: de que forma as atividades desenvolvidas estimulam o PC nos alunos, assim como contribuem para a superação de dificuldades na disciplina de Lógica Computacional?

Seguindo a metodologia de *Gamificação* descrita na seção 2.4 do Capítulo II, foi ofertado um curso com cinco encontros síncronos realizados no ambiente virtual *Google Meet*, totalizando uma carga horária de 20 horas.

Apesar da oferta para participação dos encontros ser para uma turma com 32 alunos, apenas 13 inscreveram-se. Durante a realização dos encontros, o número de alunos que participou efetivamente variou entre cinco e três alunos. Para manter o anonimato, nos dados aqui apresentados são utilizadas letras do alfabeto para identificar tais alunos, que interagiram durante a realização das atividades e manifestaram dúvidas que permitiram a análise dos aspectos relacionados ao PC.

O trabalho foi desenvolvido seguindo a proposta da plataforma utilizada, que funciona como um sistema tutor, em que o aprendiz não precisa de conhecimentos prévios em lógica de programação nem em relação ao manuseio da ferramenta. Na condução dos encontros não houve explicação dos objetivos de cada atividade, que era constituída por desafios ou dos conteúdos, envolvendo os elementos de programação. Os alunos deveriam resolver de maneira livre, individualmente e sem a necessidade de utilização de uma estratégia específica de resolução de problemas. A plataforma induz à resolução, apresentando o objetivo de cada desafio, dicas de resolução, blocos de programação conforme o objetivo do desafio e a quantidade máxima de blocos que deve ser utilizada.

Conforme os alunos iam resolvendo os desafios, já poderiam seguir adiante até concluírem a atividade sozinhos e, caso tivessem alguma dúvida ou dificuldade durante esse

processo, a professora estava à disposição para auxiliá-los. Os alunos não abriram as suas câmeras e permaneceram em silêncio na maioria do tempo, a menos que surgisse uma dúvida ou fossem indagados pela professora como estava o andamento da atividade. Nesses momentos, respondiam de forma oral e compartilhavam a tela, mostrando como estavam resolvendo o desafio.

As dúvidas foram utilizadas como indicativos de que novos elementos de programação, diferentes daqueles utilizados nas atividades anteriores, eram necessários para a resolução do desafio. Então essas interações foram registradas pelas falas dos alunos e captura de tela quando compartilhadas por eles, para a análise. Além das dificuldades explicitadas nas falas dos alunos, esses momentos foram identificados por observações da professora, com base nos recursos e na estrutura das atividades da plataforma.

A seguir são apresentadas algumas informações coletadas e os resultados referentes ao desenvolvimento de habilidades do PC, identificadas durante a experiência de ensino vivenciada e agrupadas conforme os pilares do PC. Para tanto, optou-se em utilizar como categorias de análise, os quatro pilares do Pensamento Computacional concluindo assim, a última fase da análise de conteúdo (inferências e interpretação dos dados). As atividades utilizadas nesta análise foram aquelas em que houve maior interação e que permitiram a coleta de dados diante do formato das aulas, sendo estas: Infinity Play Lab (trilha 10), Aventureiro Minecraft (trilhas 3 e 7), Frozen (trilhas 10, 13 e 17) e A Jornada do Herói Minecraft (trilha 10). Vale ressaltar que as categorias ora são mais evidenciadas em uma atividade ora em outras, desse modo, optamos em destacar a(s) categoria(s) que mais se evidenciam em determinada atividade, embora todas estejam presentes, pois se constitui em um único processo. Destacamos ainda, que a análise das atividades não seguem uma ordem de aplicação, mas está organizada conforme os pilares do Pensamento Computacional.

#### **4.1 As atividades realizadas e os Pilares do PC**

Conforme descrição anterior, consideram-se os pilares de decomposição, reconhecimento de padrões, abstração e algoritmo, no contexto do desenvolvimento do Roteiro de atividades da plataforma Programaê!.

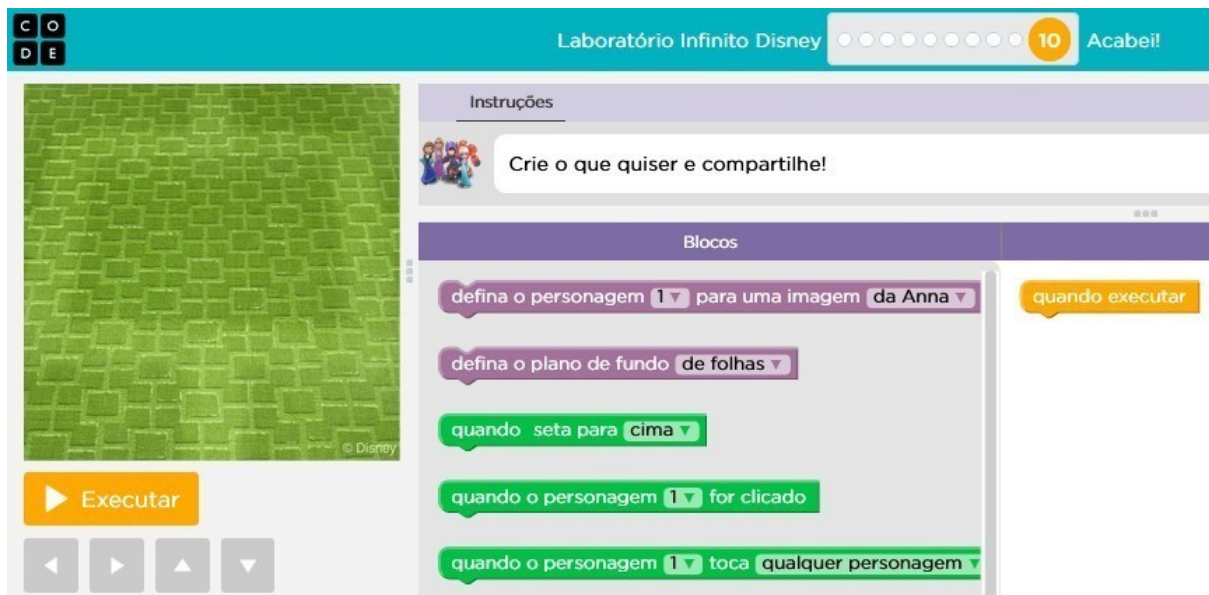
**Em relação à Decomposição**, tomamos o exemplo dado por Brackman (2017) ao relacioná-la a uma bicicleta. De acordo com o autor quando a decomposição é aplicada a um elemento físico, a manutenção desse aparelho fica mais fácil, pois se a bicicleta fosse fabricada em forma de uma única peça seu reparo por algum problema seria muito mais

difícil. Essa analogia também é válida para o desenvolvimento de programas de computadores.

O pilar decomposição envolve a capacidade de decompor ou “quebrar” um grande problema em problemas menores e assim ser capaz de solucioná-lo gradativamente. Desta forma, a atividade Infinity Play Lab (trilha 10) é um bom exemplo, pois os desafios são apresentados ao longo da atividade até chegar na resolução do problema final. Essa atividade possui vídeos introdutórios, utiliza personagens da Disney e aborda comandos que usam o conceito de *pixel*<sup>4</sup>. São 10 desafios ou trilhas envolvendo programação sequencial em dois momentos, eventos em cinco e estruturas de repetição em três momentos (conforme apontado no Quadro 2). Ao longo dos desafios, vão sendo apresentados os elementos de programação necessários para a realização do último desafio, cujo objetivo é criar seu próprio jogo.

As trilhas são as etapas de cada atividade e em cada etapa deve-se cumprir com o que está sendo pedido no enunciado da tarefa, que também são chamadas de desafios. A última trilha (trilha 10) da atividade tem como enunciado a criação de um jogo de forma livre. O que aparece do lado esquerdo da tela, em verde (área de execução) é apenas o plano de fundo padrão, não possui personagens e nada se movimenta (Figura 9).

Figura 9 - Infinity Play Lab: Trilha 10 - Tela inicial



Fonte: code.org, acesso em 08/02/2022.

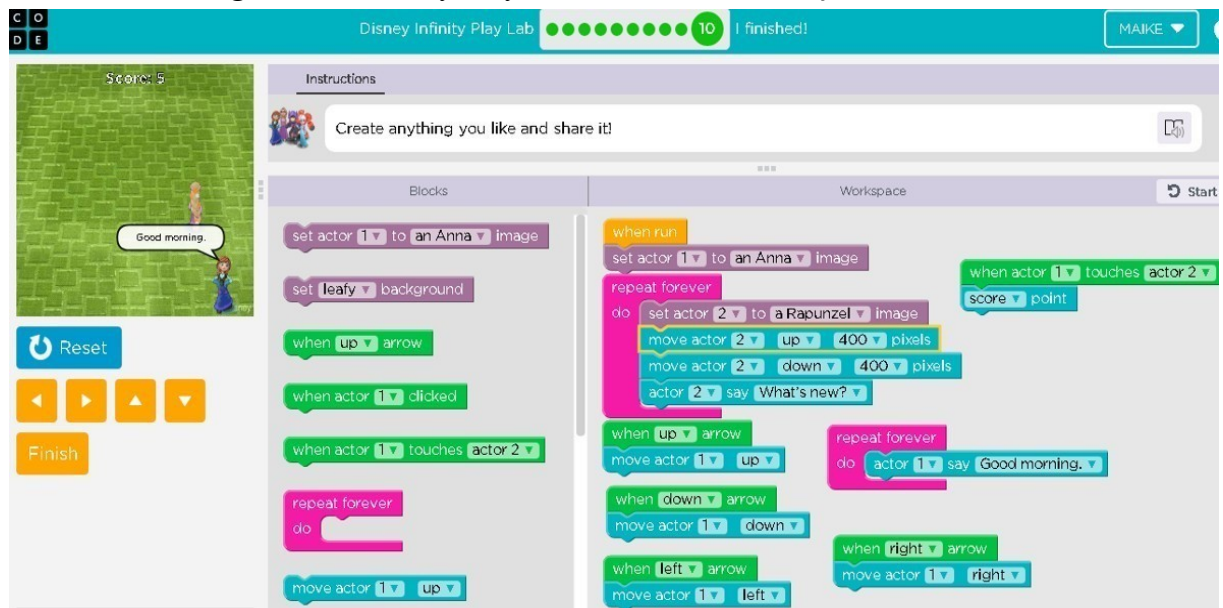
Nessa trilha, cabe ao aluno adicionar o código (conjunto de blocos) para que seu próprio jogo funcione. Conforme os blocos disponíveis, o aluno deve, minimamente, definir um personagem,

<sup>4</sup> *Pixel* é a menor área retangular resolvível de uma imagem, seja em uma tela ou armazenada na memória. Em uma imagem, cada pixel tem seu próprio brilho e cor, geralmente representados como um trio de intensidades de vermelho, verde e azul. Disponível em <http://foldoc.org/pixel> com acesso em 10/02/2022.

um plano de fundo e as ações para quando a seta for acionada. O aluno também pode incluir blocos para que aconteça algum evento quando o personagem for clicado ou quando ele encostar em outro personagem.

Como resultado da atividade proposta, temos a mesma tela (Figura 10) ampliada com a produção do aluno A, onde na área de trabalho ele acrescentou os blocos para construção do seu próprio jogo e compartilhou a tela para exibir sua produção final. Mesmo sendo indicado como usar em português, o aluno preferiu usar a plataforma em inglês.

Figura 10 - Infinity Play Lab: Trilha 10 - Produção do aluno A



Fonte: Adaptado de code.org por aluno A, 2021.

Podemos observar na Figura 10, na área de trabalho (lado direito da tela), que foram usados vários conjuntos de blocos, totalizando 21 blocos com comandos diferentes. O conjunto de todos resulta na resolução do problema. Os blocos são, por exemplo, “quando o personagem 1 toca o personagem 2” (bloco verde “*when actor 1 touches actor 2*”) e “mova o personagem 1 para cima” (bloco azul “*move actor 2 up 400 pixels*”). Esses dois blocos invocam o personagem para que algo interessante ocorra e aconteça a interação com o usuário. Os outros blocos de comando das funções são inserir personagem, inserir sons aos movimentos dos personagens, entre outros.

A estratégia utilizada pelo aluno foi de construir uma possível solução para um problema, inserindo os blocos aos poucos. Ao criar o próprio jogo o aluno não desenhou um plano sequencial, ele primeiro identificou os conceitos, em seguida, desenvolveu um plano para jogo e em seguida, implementou o código. Ele executou ciclos iterativos de imaginação e construção desenvolvendo um pouco, depois experimentando e desenvolvendo ainda mais,



com base em suas experiências e novas ideias. É um processo adaptativo, aquele em que o plano pode mudar em resposta à abordagem de uma solução em pequenas etapas.

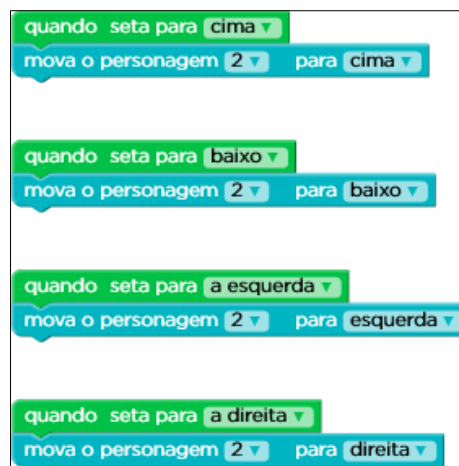
A modularização dos comportamentos dos personagens e objetos do jogo por meio de eventos, tornou mais fácil para o aluno pensar sobre as diferentes partes, e torna o código organizado caso outras pessoas queiram fazer a leitura dele.

Ao perceber que alguma ação era encaixada de forma inadequada e não funcionava como esperado, o aluno detectava o erro através da execução da atividade e concluía que para chegar ao resultado precisaria incluir ou trocar a ordem dos blocos.

Também conseguimos observar, no lado direito da tela (Figura 10), vários conjuntos de blocos separados, onde o aluno precisou cuidar que cada parte tivesse apenas um propósito, de modo que depois pudesse compor as partes de maneira eficiente para cumprir com o objetivo do desafio e, por consequência, da atividade. A predominância do elemento de programação Eventos foi constatada na produção final do aluno. Um evento pode desenvolver uma ação de maneira isolada, porém quando existem um conjunto de eventos, articulados em um mesmo cenário, eles contribuem para a resolução de um problema, nesse caso, a criação de um jogo interativo

Além disso, para atingir o objetivo e evitar falhas no algoritmo, o aluno usou como base a observação e na forma de um processo adaptativo, mudou de plano para que a solução acontecesse em pequenas etapas. Nesse sentido, o jogo apresentado pelo aluno pode ser decomposto em partes menores que envolvem a inclusão de blocos independentes para que possa atingir assim seu objetivo. A Figura 11 exibe um exemplo de blocos de comando de eventos que funcionam de forma isolada.

*Figura 11 - Infinity Play Lab: Trilha 10 - Blocos de eventos*



Fonte: code.org. Adaptado pela autora, 2022.

Ao observar a Figura 11 percebem-se blocos nas cores verde e azul. Os blocos verdes referem-se a comandos de sensores para o acionamento de teclas e os blocos azuis são códigos que indicam movimento. O encaixe das duplas de blocos indica que quando uma tecla de seta do teclado é pressionada, um evento acontece, movendo o personagem na direção da seta acionada. Os blocos determinam quando um evento acontece e assim, sinalizam ações dos usuários. Caso um personagem tenha a meta de percorrer todo o cenário, então o código que aparece na Figura 11 é satisfatório. Caso o personagem fosse uma raquete que precisa movimentar-se apenas para cima e para baixo no cenário, então o código da Figura 11 estaria em excesso, mas não incorreto. Cada um dos blocos referentes a um evento definido pelo aluno reflete a capacidade de decompor uma solução.

Vale ressaltar que a construção do conhecimento acontece de forma gradual; a medida que o aprendiz avança as etapas, a complexidade aumenta, motivando a superação dos desafios, conforme os pressupostos da gamificação.

A análise das informações considerou a organização disposta no Quadro 3, com as práticas e estratégias que os alunos dispuseram para atingirem os conceitos de programação, utilizando os elementos disponíveis.

Considerando que a prática de ser incremental e interativo se torna perceptível na programação por meio dos eventos, quando são usadas estratégias para resolução de problemas e a identificação de uma possível solução, a realização desse desafio permite inferir que o aluno A manifestou a habilidade de PC referente ao pilar decomposição. Conforme Brackman (2017), a decomposição, que favorece a solução de problemas por permitir a mudança do plano em resposta à abordagem de adaptação dos eventos em pequenas etapas. Isso pode ser observado na atividade realizada pelo aluno A.

**Em relação ao reconhecimento de Padrões**, a atividade Aventureiro Minecraft nos permite identificar com maior facilidade a presente categoria. Afinal, o reconhecimento de padrões está relacionado à associação de algum objeto (ou parte dele), tangível ou conceitual, com padrões familiares que permitam identificá-lo e classificá-lo (VICARI, 2018).

Para explicar o reconhecimento de padrões, Brackman (2017) usa o exemplo de identificação de similaridades entre raças de cachorros. Todos os cachorros possuem pêlos, rabo, quatro patas, porém suas características podem ser diferentes. Mas, quando se consegue um padrão de cachorro, todos os outros cachorros podem ser descritos seguindo o padrão alterando apenas as características.

No terceiro encontro foi desenvolvida a atividade Aventureiro Minecraft. Essa

atividade inicia com um vídeo introdutório, o qual explica que o aluno deve escolher um personagem e como funciona a tela separada em cenário (área de execução), blocos de comando e área de trabalho (Figura 12). A atividade é composta de 14 trilhas e tem como característica peculiar o cenário, que aparece em forma quadriculada como um plano cartesiano e indica cada passo que o personagem está dando. O cenário segue o padrão do jogo e, assim como no jogo, o que torna a atividade interessante é a exploração, fazer os personagens aventurarem-se por grutas e ver o que conseguem encontrar.

Figura 12 - Cenário Aventureiro Minecraft



Fonte: code.org acesso em 08/02/2022.

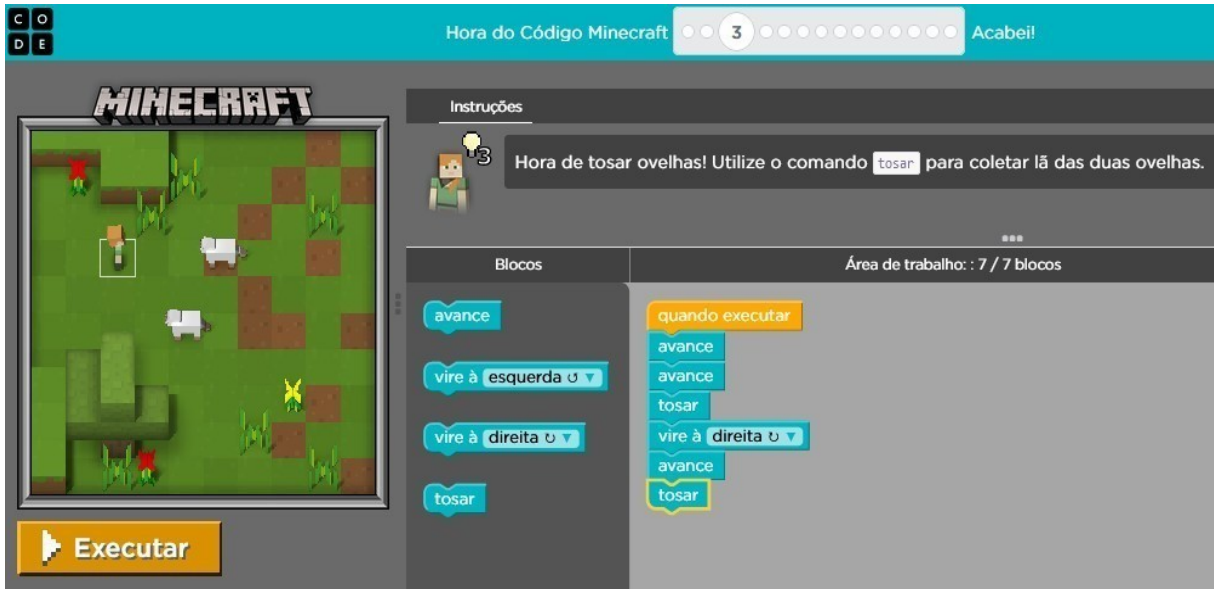
Os elementos de programação que são utilizados no percorrer da atividade são: de programação sequencial (quatro trilhas), repetições (seis trilhas) e decisão (seleção) (quatro trilhas), conforme levantamento realizado na fase de metodologia mostrados no Quadro 2.

O enunciado da trilha 3 pede que o aluno mova o personagem pelo cenário e tose as duas ovelhas presentes nele. Para isso o aluno deve inserir blocos com os comandos “avance”, que significa o mesmo que “ande”, “mova”; o bloco “tosar” e “vire à direita”. O desafio é concluído quando o personagem se aproxima e tosa as duas ovelhas. A atividade do Aventureiro Minecraft destacou-se por gerar um maior número de dúvidas e interação durante o encontro, o que tornou possível coletar os dados registrados através das falas dos alunos.

A fala do aluno B com relação a trilha 3 (Figura 13) foi a seguinte: “*Professora, esse eu não consegui fazer. Eu consegui fazer para chegar nas ovelhas, mas tem o limite de*

sete blocos, pode me mostrar como que faz?”

Figura 13 - Aventureiro Minecraft: Trilha 3



Fonte: code.org, acesso em 08/02/2022.

Posteriormente a fala do aluno B foi com relação a trilha 7 (Figura 14): “*Professora você pode falar os comandos que você colocou ali porque tá ruim de enxergar, o último quadradinho ficou faltando ele plantar.*”. A trilha 7 consiste em movimentar o personagem para que plante dos dois lados da água, como mostra o cenário na Figura 15. Durante os encontros, a professora compartilhava a tela e também ia resolvendo as atividades, por isso o aluno se queixa que está ruim de enxergar.

Figura 14 - Aventureiro Minecraft: Trilha 7 - Blocos



Fonte: programae.org.br Acesso em janeiro de 2022.

O aluno foi orientado que contasse, fizesse o cálculo dos quadradinhos que formam o cenário. A resposta dele foi:

Aluno B: “*Eu consegui professora mas usei 12 blocos.*”

Figura 15 - Aventureiro Minecraft: Trilha 7 - Cenário



Fonte: [programae.org.br](http://programae.org.br) Acesso em janeiro de 2022.

Nos dois casos o aluno conseguiu cumprir o desafio da trilha, foi eficaz, mas não foi eficiente. Houve falha na construção do algoritmo por não ser otimizado, porém o aluno usou a estratégia de resolução do problema com foco no enunciado, como consta no Quadro 3, que analisa o pilar Abstração com elementos de programação e trilhas de algumas das atividades.

Sempre que novas informações, olhares ou desafios surgirem, abstrações poderão ser revisadas ou reconstruídas. O problema enfrentado pelos alunos foi o de não se utilizarem de estruturas de repetição nos seus códigos. Ao utilizar tais estruturas você agrupa um conjunto de blocos “avance”, por exemplo, concentra-se no resultado e não nas linhas de código. Mas não se pode concluir que isso aconteça somente com estruturas de repetição, pode acontecer com estruturas de seleção. “Facilitar o processo” (como está no quadro), usando repetição, seria um indício da capacidade de abstração motivada pelo objetivo da trilha.

Dessa maneira, no caso da atividade apresentada, observou-se que em relação ao pilar abstração, que envolve manter a atenção a fatores relevantes ao objetivo proposto em detrimento a fatores irrelevantes, ao agrupar tais blocos em um bloco repetição, a realização do desafio permitiu observar que os alunos foram capazes de abstrair e mantiveram a atenção muito mais no enunciado do que no algoritmo.

**Em relação ao pilar Abstração**, Brackman (2017) utiliza o exemplo de um mapa de metrô, onde estão impressas todas as vias que o metrô percorre. Quando o passageiro ignora

todas as informações que não lhe são úteis e se concentra apenas nas informações necessárias que lhe levarão ao seu destino, está realizando uma abstração (BRACKMAN, 2017).

Assim, a atividade realizada foi Frozen, pois permite explorar a magia e a beleza do gelo com as personagens do filme da Disney, Anna e Elsa. Através do que se pede nos enunciados das vinte trilhas que compõem a atividade, ao inserir os blocos de comandos é possível criar retas, formas geométricas e por consequência, criar flocos de neve enquanto esquia com as personagens no cenário (Figura 16). Entre os elementos de programação predominantes nessa atividade, pode-se observar três trilhas sobre programação sequencial, dez trilhas sobre repetições e sete trilhas sobre funções.

Figura 16 – Frozen: Trilha 1



Fonte: adaptado de [programae.org.br](http://programae.org.br) Acesso em 30/01/2022.

O enunciado da trilha 10 da atividade *Frozen* pede o seguinte: “Use um bloco "Repita" em torno desses blocos (Figura 17) para criar um paralelogramo. É como um retângulo, mas tem ângulos diferentes. Esse tem ângulos de 60 graus e 120 graus em vez de todos os ângulos terem 90 graus”.

Figura 17 – Frozen: Trilha 10



Fonte: [programae.org.br](http://programae.org.br)

A informação no título da Área de trabalho sugere que são necessários 8 blocos de

código para cumprir o desafio, sendo que cinco já foram inseridos. Durante o encontro, ao chegar nessa trilha o aluno D relatou como que ele fez: “*nesse daí você coloca um bloco de repita duas vezes e daí vire à direita 60 graus, e depois 120 no de baixo.*”. A Figura 18 exemplifica como ficou a solução da trilha de acordo com a sugestão do aluno, que moveu os blocos pré-existentes para dentro de uma estrutura de repetição.

Figura 18 – Frozen: Trilha 10 – Resolução do aluno D

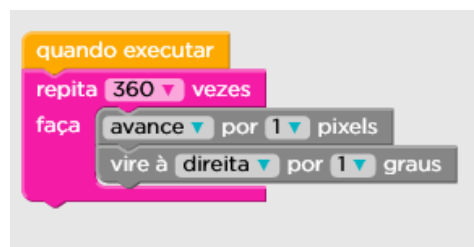


Fonte: programae.org.br

Ao ser questionado sobre como chegou ao resultado, o aluno D respondeu: “*Foi como fazer o retângulo*”. Na mesma atividade, na trilha 17 (Figura 19) o enunciado diz: “*Padrões de neve complexos podem ser criados com formas muito simples. Você pode criar um padrão repetindo 5 círculos de tamanho 5 e 5 círculos de tamanho 10?*”

Na atividade da *Frozen*, o elemento de programação função aparece com frequência (trilhas 14 a 20). A Figura 19 diz respeito a trilha 13 da atividade da *Frozen*, cujo objetivo é desenhar um círculo e usa um bloco de repetição para repetir 360 vezes o comando de “*avance*” e “*vire a direita*”. Na Figura 20 aparece uma função que desenha um círculo e recebe como parâmetro qual deve ser o tamanho desse círculo.

Figura 19 - Frozen: Trilha 13 - Criação do círculo

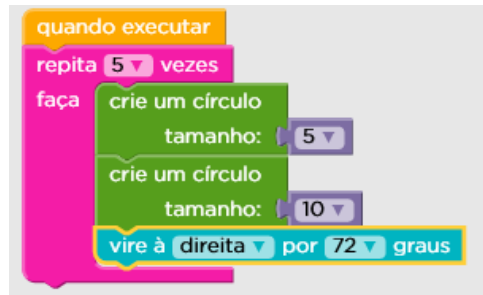


Fonte: programae.org.br

Na programação, às vezes é necessário executar um determinado bloco repetidas vezes. Nesse caso, é necessário criar um loop que repita uma determinada parte do processamento quantas vezes forem necessárias. Esses loops repetidos também são chamados de loops. A principal vantagem desta função é que o tamanho do programa é menor e sua

faixa de processamento pode ser aumentada sem alterar o tamanho do código de programação. (MANZANO; OLIVEIRA, 2005).

*Figura 20 - Frozen: Trilha 14 - Uso da função ‘círculo’*



Fonte: [programae.org.br](http://programae.org.br)

Quando um bloco de código é utilizado várias vezes como solução do problema, quer seja em uma estrutura de repetição, quer seja em pontos distintos do programa, o conjunto de código resultante pode constituir uma função. A vantagem da função é não precisar copiar o código todas as vezes que precisar executar aquela operação, além de deixar a leitura do código mais intuitiva. Uma função é definida como um conjunto de comandos identificado por um nome que pode ser usado em qualquer outra parte do programa. Ou seja, o bloco de instruções pode ser reaproveitado e a própria definição de função exerce a prática da reutilização, de acordo com o Quadro 3. A fala do aluno acima mostra que ele já sabia como fazer e o que mudava apenas eram os parâmetros (assim como as características do cachorro, exemplificado por Brackman, 2017, no início desta seção).

As práticas de reutilização e remixagem têm sido usadas há tempos na programação de computadores, pois reusar um trecho de código que já foi testado e está pronto permite chegar ao objetivo mais rapidamente.

As práticas de abstração e modularização também foram identificadas, quando o aluno usa blocos de repetição, o que leva a construção da função. Ou seja, ele reconheceu uma propriedade comum que aparecia na construção dos blocos, conforme avançava nas trilhas.

Uma atividade que apresenta bem o conceito de função é O Artista. Essa atividade foi desenvolvida no quarto encontro e consiste em movimentar um personagem denominado “Artista” que desenha por todo lugar que passa. Ao percorrer as 10 trilhas disponíveis nesta atividade, o aluno precisa decidir qual ângulo deverá usar para criar as formas que estão sendo pedidas nos enunciados. No caso da trilha 8, por exemplo, o enunciado apresenta o seguinte texto: “Uma função é mais uma ferramenta de programação para que você não tenha



que repetir as mesmas ações. Esta função (Figura 21) desenha uma flor, assim, você pode usá-la sempre que quiser fazer esse desenho. Use-a juntamente com o novo bloco "pule" para desenhar essas flores. Dica: as flores são separadas por 150 pixels". O padrão dos desenhos das flores aparece na Figura 23.

Na Figura 21 pode-se observar que todo o bloco “Função” recebe o nome “desenhe uma flor” e todos os seus seis comandos não precisam ser escritos novamente.

Para desenhar as três flores como diz no enunciado o aluno deve apenas arrastar para a área de trabalho apenas um bloco da função dentro de um bloco de repita 3 vezes, como mostra a solução da trilha na Figura 22.

Figura 21 - O Artista: Trilha 8 - Definindo uma função



Fonte: programae.org.br. Acesso em 12/04/2022.

Figura 22 - O Artista: Trilha 8 - Utilizando a função ‘flor’

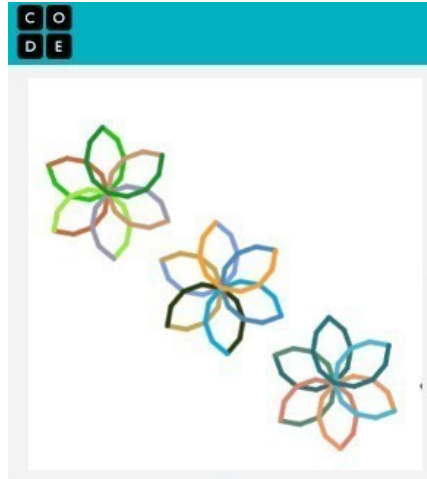


Fonte: programae.org.br. Acesso em 12/04/2022.

Em relação ao pilar reconhecimento de padrões, os alunos foram capazes de perceber que as regras para criar as formas durante as trilhas da atividade da *Frozen* tinham o mesmo padrão e valeram-se da estratégia do uso de contraexemplos e das propriedades comuns aos desenhos geométricos, que se repetiam a cada fase do jogo e que apenas acrescentavam-se

novas ações a cada fase. Essas estratégias constam no Quadro 3 de Análise do capítulo da metodologia.

*Figura 23 - O Artista: Trilha 8 - Padrão de flores*



Fonte: [programae.org.br](http://programae.org.br). Acesso em 12/04/2022.

Na medida em que os alunos se familiarizaram com a dinâmica das atividades, ou seja, repetir a sobreposição de figuras quase sempre da mesma forma, esses foram capazes de realizar as atividades de maneira colaborativa, mais rápida e com maior desenvoltura.

Na atividade Infinity Play Lab, já analisada sob o pilar Decomposição (item 4.1), o aluno, ao realizar a última trilha para criar seu próprio jogo, relatou o seguinte:

Aluno B – *“eu tentei fazer quando no início eu acerto o personagem, mas eu não achei a função pra isso, aí eu tentei encostar pra isso”*.

Na interação entre os personagens que ele criava para o seu jogo, o objetivo era encostar no outro personagem para vencer, como um “pega-pega”, porém só estava disponível o evento “ao clicar” e não o evento que o aluno estava buscando, como por exemplo “ao tocar”. Portanto, como o aluno B não encontrou o comando que desejava, substituiu por outro semelhante. Aqui o aluno mostrou a prática de reuso e remixagem.

**O pilar Algoritmo**, última categoria de análise, é o elemento que agrega os demais elementos de programação e pode ser explicado como sendo o conjunto de passos ou regras simples, criados para resolver os problemas encontrados. Utilizado na resolução de problemas complexos eficientemente, os algoritmos podem ser compreendidos por sistemas computacionais e seus passos ou regras não servem apenas para criar um código, também é possível ser usado no dia a dia, na elaboração de uma receita culinária, na operação de um eletrodoméstico seguindo o manual de instruções, enfim, independentemente da carreira profissional que o estudante deseja seguir (BRACKMAN, 2017, p.33).

Na elaboração de um algoritmo, é importante usar técnicas que facilitem o processo de construção, que ajudem a testar se o algoritmo obtido resolve o problema e, ainda, que permitam aferir o desempenho da solução.

Na elaboração de um algoritmo, é importante usar técnicas que facilitem o processo de construção, que ajudem a testar se o algoritmo obtido resolve o problema e, ainda, que permitam aferir o desempenho da solução. Para tanto, a atividade desenvolvida no curso foi A Jornada do Herói. Para completar essa atividade, o aluno precisa escolher um personagem (Alex ou Steve) e escrever os códigos para programar “O Agente”. O aluno realiza as trilhas em conjunto com “O Agente” para limpar quaisquer obstáculos em seu caminho, para que possa coletar os itens que precisará para a jornada. Apenas “O Agente” pode colocar e quebrar blocos, e apenas o personagem que o aluno comanda pode coletar itens. “O Agente” pode andar, virar e ativar placas de pressão. Ele também pode destruir blocos e colocar blocos. Quando ele coloca um bloco, como por exemplo, os trilhos de carrinho de mineração, os coloca debaixo dele mesmo (Code.org, 2022). A Figura 24 apresenta a tela inicial dessa atividade que é composta de 12 trilhas.

Figura 24 - A Jornada do Herói Minecraft: Tela inicial



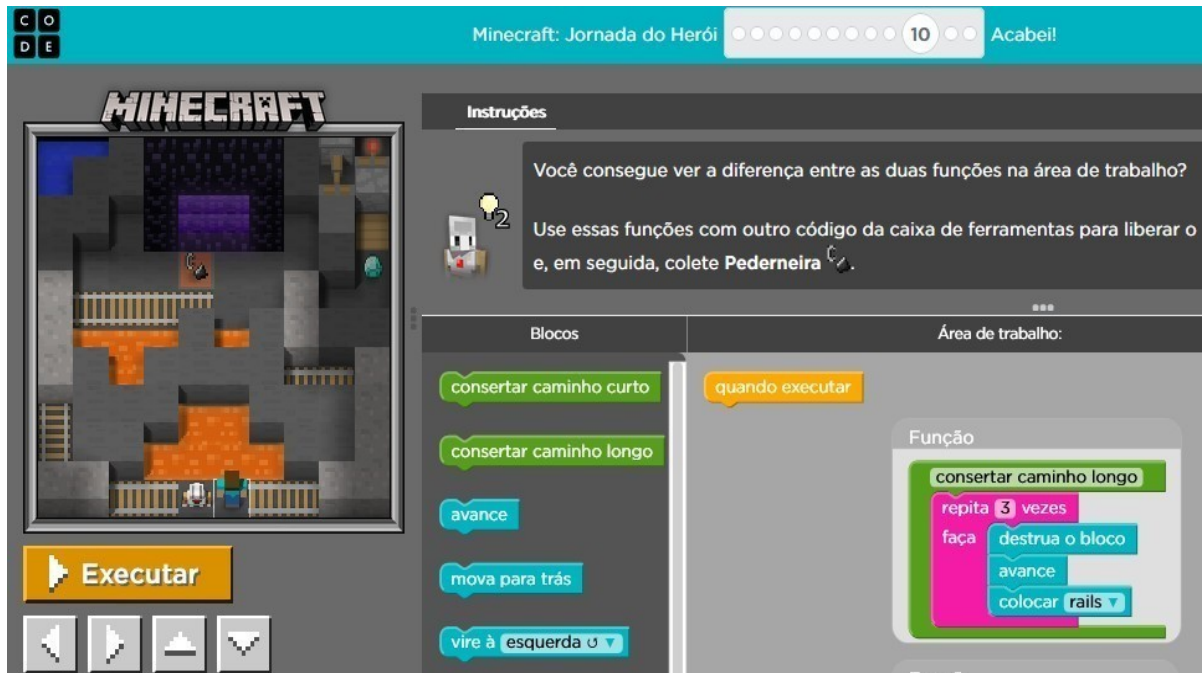
Fonte: code.org. Acesso em 17/03/2022.

A atividade A Jornada do Herói Minecraft possui duas trilhas que abordam programação sequencial, e cinco e quatro trilhas sobre os elementos de programação repetição e funções, respectivamente.

Na trilha 10, o enunciado pergunta: “Você consegue ver a diferença entre as duas

funções na área de trabalho?” e em seguida direciona a instrução: “Use essas funções com outro código da caixa de ferramentas para liberar o caminho até o Carrinho de Minas e, em seguida, colete a Pederneira”. A Figura 25 apresenta o cenário, onde pode-se observar o personagem e “O Agente”, os trilhos, os obstáculos e a Pederneira.

Figura 25 - Jornada do Herói Minecraft: Trilha 10



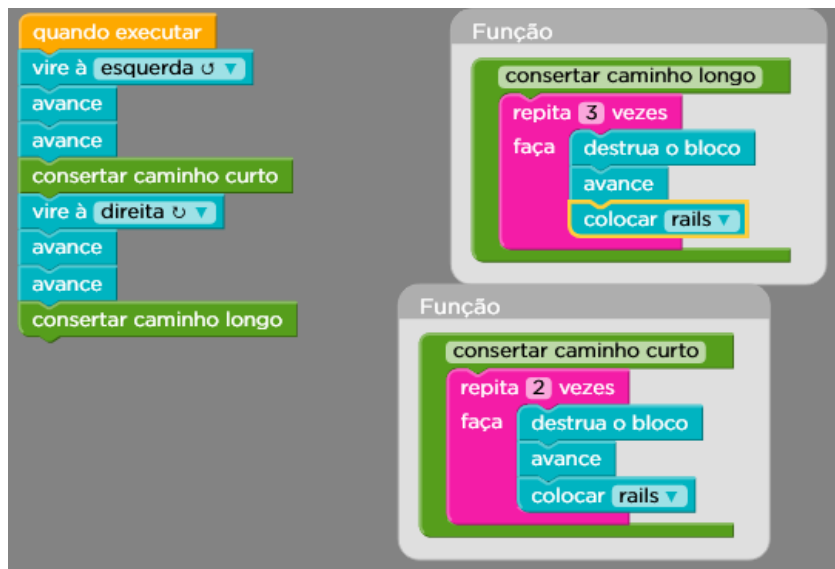
Fonte: [programae.org.br](http://programae.org.br).

A tarefa deve ser iniciada com um comando que mude a direção do Agente para a esquerda. O aluno deve notar quantas repetições possui dentro de cada função, pois dentro dela existe um comando de “Avance”, mostrando quantos passos o Agente precisa andar para consertar o caminho todo. O número de passos deve ser contado através da imagem. As setas (Figura 25) servem para movimentar o personagem para que ele saia do carrinho e conclua a atividade. A Figura 26 mostra a resolução do desafio sem o uso do comando de repetição e a Figura 27 mostra a resolução do desafio com o uso do elemento de programação repetição.

Na trilha 10 (Figura 26) o aluno B relatou que não usou nenhum “repita”: “*Eu consegui fazer mas eu não entendi como que faz.*”

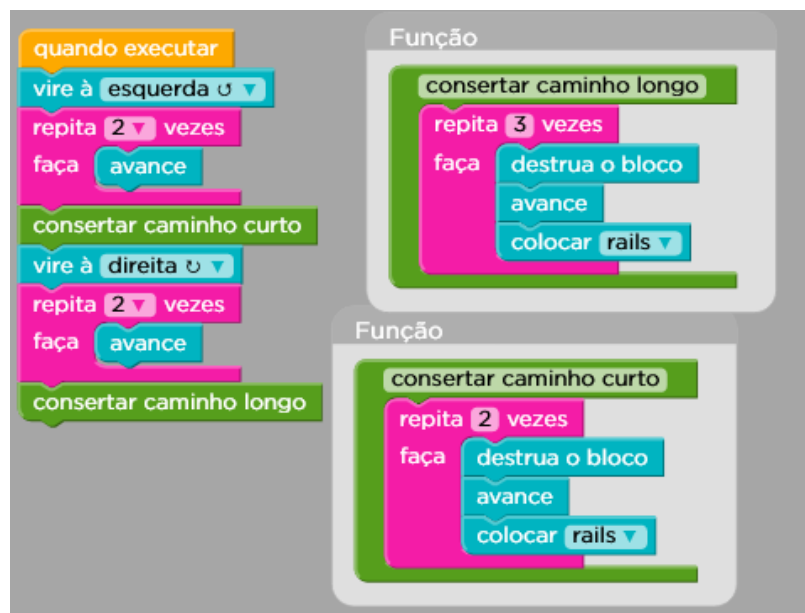
Ao chegar na solução da trilha, fez-se necessárias algumas tentativas por parte do aluno, pois apesar de concluir o que se pedia, o personagem ficava ‘batendo na parede’, ou seja, estava sobrando código.

Figura 26 - Jornada do Herói Minecraft: Trilha 10 - Área de trabalho



Fonte: programae.org.br

Figura 27 - Jornada do Herói Minecraft: Trilha 10 - Uso do bloco 'repita'



Fonte: programae.org.br

Em termos de programação, isso significa que com a lógica utilizada aliada aos comandos adequados, um programador pode modelar qualquer problema passível de ser resolvido através de algoritmos em uma linguagem de programação atual. Com isso também se aborda outro processo recorrente na programação, o processo de *sequenciação* de comandos válidos que como o próprio nome já deixa claro, trata-se de uma sequência de comandos para executar determinada ação (SANTOS *et al.*, 2016).

A elaboração de um algoritmo é um passo importante na resolução de um problema,

portanto é importante testá-lo. Assim, após elaborar um algoritmo, antes de executá-lo, deve-se simular a sua execução para adquirir confiança de que ele produzirá o resultado esperado.

Esse processo de teste, conhecido como depuração, pode revelar erros e o algoritmo deve então ser corrigido. Realizar a depuração do algoritmo consiste num processo de ajuste dos códigos para localizar e remover os defeitos desse algoritmo (TICON, 2020). Em geral, para um mesmo problema existem diversos algoritmos que produzem resultados corretos.

O pilar algoritmo contempla todas as práticas: incremental e iterativo, abstração e modularização, reutilização e depuração. Contudo existe uma intensidade maior quando se trata da prática da depuração, pois para os programadores é fundamental desenvolverem estratégias para lidar com - e antecipar – problemas.

O pilar Algoritmo contempla também todos os elementos de programação: programação sequencial, eventos, estruturas de repetição e seleção, e funções. Com isso, é possível obter um algoritmo, ou seja, uma sequência finita e ordenada de regras (instruções) para realizar uma tarefa pré-definida.

O elemento de programação que está diretamente relacionado com o pilar Algoritmo é o processo de sequenciação de comandos válidos, que como o próprio nome já deixa claro, trata-se de uma sequência de comandos válidos para executar determinada ação (TICON, 2020). No desenvolvimento da atividade esse elemento consolida-se no encaixe das ações ou comandos na sequência correta para que “O Agente” destrua os obstáculos e libere a passagem para a coleta da ferramenta e conclusão com sucesso da trilha 10, citada como exemplo.

Além disso, o pilar Algoritmo abrange os demais pilares do PC. Um algoritmo pode ser projetado para solucionar um grande problema e solucionar dentro deste uma instância menor do mesmo problema. Assim, um problema complexo vai sendo resolvido de forma gradual, em partes menores (TICON, 2020). A manipulação de variáveis para armazenamento de valores, gera o pensamento algébrico abstrato e o aproveitamento de códigos com o uso de funções remete ao reconhecimento de padrões.

## **4.2 Síntese dos Resultados**

A partir da análise dos dados coletados através das interações com os alunos, somos capazes de responder a seguinte questão de pesquisa: de que forma as atividades desenvolvidas estimulam o PC nos alunos, assim como contribuem para a superação de dificuldades na disciplina de Lógica Computacional?

Com os dados coletados, foi possível identificar que nessas atividades os alunos adotaram as práticas conforme o Quadro 4. De forma concreta, foi possível avaliar o uso dos elementos de programação para a realização das atividades. A partir destes elementos, convalidamos as as práticas observadas por Brennan e Resnick (2012).

*Quadro 4* - Elementos e práticas de programação nas atividades do Roteiro

<b>Atividade – Trilha</b>	<b>Elemento predominante</b>	<b>Prática</b>
Infinity Play Lab – Trilha 10	Evento	Incremental e iterativo
Aventureiro Minecraft – Trilhas 3 e 7	Repetição	Modularização e abstração
Frozen – Trilha 10	Repetição/Função	Reutilização e remixagem
Jornada do Herói – Trilha 10	Repetição	Teste e depuração

Fonte: elaborado pela autora.

As poucas interações que os alunos realizaram durante os encontros propiciaram analisar algumas práticas e estratégias utilizadas por eles para atingir os objetivos das atividades e, através delas, desenvolver algumas habilidades de PC. Essas interações aconteceram nas trilhas listadas na primeira coluna do Quadro 4. Durante a elaboração do roteiro, cuja ordem das atividades constam no Quadro 2 desse capítulo, podemos identificar quais são os elementos de programação predominantes em cada uma delas. Esses elementos estão listados na segunda coluna do Quadro 4. Na terceira coluna estão as práticas utilizadas pelos alunos que foram identificadas na fase de análise dos dados. As práticas com programação em blocos se concentram no processo de raciocínio e aprendizado, indo além do "do que você está aprendendo" para "como você está aprendendo" (BRENNAN; RESNICK, 2012).

No caso específico do pilar decomposição, o aluno manifestou a habilidade de PC referente a esse pilar, pois consideramos que a prática de ser incremental e interativo se torna perceptível na programação por meio dos eventos, quando são usadas estratégias para resolução de problemas e a identificação de uma possível solução. O aluno também construiu uma possível solução para o problema, inserindo os blocos aos poucos. Ao criar o próprio jogo o aluno executou ciclos iterativos de imaginação e construção desenvolvendo um pouco, depois experimentando e desenvolvendo ainda mais, com base em suas experiências e novas

ideias. Foi um processo adaptativo, aquele em que o plano pode mudar em resposta à abordagem de uma solução em pequenas etapas. A modularização dos comportamentos dos personagens e objetos do jogo por meio de eventos, facilitou para que o aluno pensar sobre as diferentes partes, e torna o código organizado.

Com relação ao pilar abstração, o aluno conseguiu cumprir o desafio da trilha porém houve falha na construção do algoritmo por não ser otimizado. Notamos através da análise que o aluno usou a estratégia de resolução do problema com foco no enunciado e percebemos que o problema enfrentado pelo aluno foi o de não utilizar estruturas de repetição nos seus códigos. Mas não se pode concluir que isso aconteça somente com estruturas de repetição, pode acontecer com estruturas de seleção. “Facilitar o processo” (como está no quadro de análise), usando repetição, seria um indício da capacidade de abstração motivada pelo objetivo da trilha.

No caso da atividade apresentada, observou-se que em relação ao pilar abstração, que ao agrupar tais blocos em um bloco repetição, a realização do desafio permitiu que os alunos desenvolvessem o pilar abstração e assim, foram capazes de abstrair e mantiveram a atenção muito mais no enunciado do que no algoritmo.

A utilização do elemento de programação “função” estimula o aluno no desenvolvimento do PC com relação ao pilar reconhecimento de padrões. As práticas de reutilização e remixagem têm sido usadas há tempos na programação de computadores, pois reusar um trecho de código que já foi testado e está pronto permite chegar ao objetivo mais rapidamente. Portanto, em relação ao pilar reconhecimento de padrões, os alunos foram capazes de perceber que as regras para criar as formas durante as trilhas da atividade da Frozen tinham o mesmo padrão e valeram-se da estratégia do uso de contraexemplos e das propriedades comuns aos desenhos geométricos, que se repetiam a cada fase do jogo e que apenas acrescentavam-se novas ações a cada fase. Na medida em que os alunos se familiarizaram com a dinâmica das atividades, ou seja, repetir a sobreposição de figuras quase sempre da mesma forma, esses foram capazes de realizar as atividades de maneira colaborativa, mais rápida e com maior desenvoltura

Quanto ao pilar algoritmo, o alunos precisou de algumas tentativas, pois apesar de concluir o que se pedia, o personagem ficava ‘batendo na parede’, ou seja, estava sobrando código. Isso significa que com a lógica utilizada aliada aos comandos adequados, o educando pode modelar qualquer problema passível de ser resolvido através de algoritmos em uma linguagem de programação atual. Percebemos que o pilar algoritmo contempla todas as



práticas: incremental e iterativo, abstração e modularização, reutilização e depuração. E contempla também todos os elementos de programação: programação sequencial, eventos, estruturas de repetição e seleção, e funções. Em resposta a nossa pergunta, no desenvolvimento da atividade esse elemento consolida-se no encaixe das ações ou comandos na sequência correta para que “O Agente” destrua os obstáculos e libere a passagem para a coleta da ferramenta e conclusão com sucesso da trilha, manifestando no aluno habilidade de PC. Além disso, concluímos que o pilar Algoritmo abrange os demais pilares do PC.

No ano seguinte, os alunos que participaram dos encontros possuem a disciplina de Linguagem de Programação. Para esses três alunos foi perguntado se a realização das atividades contribuíram no aprendizado de Linguagem de Programação e as respostas dos alunos A e B foram as seguintes:

Aluno A: *“O curso que a professora deu ano passado foi com o intuito de usar a lógica como base para resolver os problemas propostos. Agora em 2022 que tenho aulas de programação, sei que saí na frente de meus colegas por ter feito o curso, pois me deu uma base estruturada. No curso eu aprendi lógica de programação.”*

Aluno B: *“O curso com o code.org me auxiliou na matéria de linguagem de programação no curso de informática. Sim, me ajudaram a ter noção básica de lógica de programação que influencia na linguagem de programação. Auxiliou na lógica e no raciocínio, não só na matéria, mas como lógica básica inclusa no pensamento humano, fundamental esse conhecimento para as pessoas que querem ingressar na linguagem de programação.”*

Tais relatos contribuem com o resultado da pesquisa e vai além, pois eles perceberam que a ferramenta não estimula somente o raciocínio lógico para um fim específico, mas para uma formação geral de aprendizado do indivíduo.

Apoiados no trabalho de Martinelli (2017) pode-se deduzir o seguinte:

- a) Houve entendimento significativo dos conceitos de PC a partir dos elementos de programação apoiados pela plataforma Programaê! de acordo com a sequência de atividades organizadas para alunos da 1º série do curso Técnico em Informática;
- b) Os alunos envolvidos apresentaram-se satisfeitos e motivados pelas atividades propostas na plataforma Programaê!

Talvez fosse necessário um tempo de pesquisa maior tendo em vista que no ano seguinte a turma dos participantes dará início a disciplina de Linguagem de Programação e não se sabe até que ponto eles entendem os conceitos de programação associados a blocos específicos. Mas foi possível identificar que, ao longo das atividades, os alunos empregaram os conceitos de programação e obtiveram os resultados esperados concluindo os desafios propostos nas trilhas.

## CAPÍTULO V

### CONSIDERAÇÕES FINAIS

A presente pesquisa surgiu da necessidade de atenuar as dificuldades observadas na disciplina de Lógica de Programação pelos alunos do curso Técnico em Informática. Em busca de uma metodologia alternativa que contribuísse para diminuir o problema, iniciamos uma investigação sobre o Pensamento Computacional com o uso da plataforma *Programaê!*

Assumi-se nessa dissertação de mestrado como objetivos específicos: apresentar o Pensamento Computacional com base na literatura vigente, utilizar a plataforma *Programaê!* nas aulas de Lógica Computacional como forma de introdução ao PC, e desenvolver um material didático voltado ao trabalho com o PC em sala de aula. A partir de tais objetivos derivou-se a seguinte questão de pesquisa: de que forma as atividades desenvolvidas estimulam o PC nos alunos, assim como contribuem para a superação de dificuldades na disciplina de Lógica Computacional?

A pesquisa bibliográfica sobre PC no âmbito educacional mostrou que a informática está saindo da superficialidade e adentrando às entranhas da Computação, para que tão logo usuários passem a ser desenvolvedores de tecnologia, atendendo às demandas profissionais do século XXI. Mais que conceituar PC, existe uma problematização por não existir uma definição única, mas sim consensos do que vem a ser o PC. Respaldados pela BNCC, documento base que rege a educação no país, nesse trabalho o PC se organiza em quatro aptidões ou quatro pilares: a decomposição, o reconhecimento de padrão, a abstração e o algoritmo. Encontramos diferentes abordagens educacionais que envolvem o PC em variados níveis de ensino, porém minguadas quando se trata da educação profissional.

Nesse âmbito, foi elaborado um produto educacional, cuja implementação em sala de aula é uma exigência. Dez atividades disponíveis na plataforma *Programaê!* foram selecionadas, a fim de constituir um roteiro introdutório de apoio à disciplina de Lógica Computacional. O produto desenvolvido possui formato de *e-book*, e caracteriza-se como um guia didático para utilização em sala de aula, contendo atividades com foco no desenvolvimento do PC através de uma abordagem prática com programação em blocos.

O professor tem um papel muito importante no processo de ensino e aprendizagem e

pode valer-se da abordagem de gamificação para desenvolver o PC em sala de aula. A metodologia de gamificação apresenta algumas linhas gerais que se assemelham aos pilares do PC, como decomposição e depuração.

Acompanhar as ações desenvolvidas pelos estudantes de forma remota, através dos encontros online, prejudicou a pesquisa ao falhar na observação no local em vários aspectos como: não ter acesso às telas dos alunos, não poder observar as interações entre eles, suas reações quando completavam os desafios, o seus comportamentos quando erravam ou acertavam, o tempo do percurso de cada trilha e para analisar quais componentes do PC estavam sendo explorados por intermédio de quais elementos de programação.

Pode-se concluir que as atividades se relacionam com o PC tanto na sua forma com a construção de jogos/programas (passo a passo para encaixe e ordem dos blocos) quanto no seu método de resolução de problemas (decomposição, abstração, reconhecimento de padrões e algoritmo).

As práticas de abstração e modularização, ser incremental e iterativo e reutilização se aplicam a todas as atividades. Os elementos de programação estão diretamente relacionados com os pilares do PC: algoritmo (ordem dos comandos), decomposição (eventos) e reconhecimento de padrões (funções). O pilar Abstração foi diagnosticado em todas as atividades, pois a abstração que dá base para o pensamento. Abstração envolve capacidade de leitura. Se um aluno não é apto a ler e compreender o que lê, isso compromete a capacidade de abstração e, por conseguinte, de desenvolver PC em contextos como esse.

Não há um método simples que permita avaliar se os aprendizes desenvolveram habilidades de PC; no entanto, a verificação das atividades realizadas através das práticas, permite inferir que, a partir do uso dos conceitos (elementos de programação), pode-se concluir que as ações desenvolvidas nessa pesquisa que os pilares do PC foram consolidados através das atividades propostas, pois os alunos reconhecem que essas atividades contribuem para o aprendizado de Lógica Computacional. A superação da dificuldade nessa disciplina prevê um desempenho satisfatório na disciplina de programação de computadores, a qual estudarão na série seguinte. A programação de computadores está voltada a prática da resolução de problemas, sendo uma ciência que envolve um conjunto de princípios, técnicas e formalismos que visam a produção de software de qualidade. Para Brennan e Resnick (2012) a programação é um cenário valioso para o desenvolvimento de habilidades para o PC.

Em 2022 o Estado do Paraná iniciou o Novo Ensino Médio e incluiu na matriz curricular duas aulas semanais de PC nas turmas da 1ª série do Ensino Médio Regular,

reconhecendo a importância desse assunto na formação dos jovens. Porém, reduziram o termo PC nessa disciplina, abordando em seu conteúdo o ensino de lógica de programação em Javascript e programação Web com HTML e CSS. Esse fato confronta os princípios do PC que envolve um conceito mais amplo, como uma habilidade de resolução de problemas inerente ao raciocínio humano entre outras características como construção de conceitos antes de programação, redução de problemas grandes e complexos em problemas menores e mais simples, combinação de habilidades matemáticas e técnicas, geração de ideias antes dos artefatos e elaboração de abstrações (SANTOS, 2018).

Ainda em 2022 com a implementação de forma efetiva do Novo Ensino Médio nos colégios do Estado do Paraná, o curso técnico em Informática iniciou um novo formato, passou a denominar-se Curso Técnico em Desenvolvimento de Sistemas e as disciplinas técnicas acontecem de forma remota. Apesar das mudanças na matriz curricular, permanece a disciplina de Lógica Computacional, no entanto, de forma remota.

Como extensão deste estudo para trabalhos futuros, destaca-se aqui a necessidade de acompanhamento dos alunos que frequentaram os encontros para uma investigação se o conhecimento prévio adquirido é um diferencial no seu desempenho construção de algoritmos. Outra sugestão, como análise futura, fazer uma comparação com pré teste e pós teste, ou com turmas de controle, com o intuito de comparar se houve avanço no uso dos conceitos e práticas de programação.

Outra necessidade é preparar os professores para ensinar o PC, por isso sugere-se a oferta de oficinas de formação em PC para os professores e a disseminação do produto educacional que foi desenvolvido como resultado desta pesquisa.

## REFERÊNCIAS

- ARAGÃO, R. M. R. **Teoria da aprendizagem significativa de David P. Ausubel**: sistematização dos aspectos teóricos fundamentais, 1976. 97f. Tese de Doutorado. Programa de Pós-graduação em Educação. Universidade de Campinas, Campinas, 1976.
- ARAÚJO, D. *et al.* O ensino de computação na educação básica apoiado por problemas: Práticas de licenciandos em computação. In: **Anais do XXIII Workshop sobre Educação em Computação**. SBC, 130-139, 2015.
- AZEVEDO, Samuel *et al.* Introdução a Robótica Educacional. 1999.
- BARCELOS, T. S.; SILVEIRA, I. F. Pensamento computacional e educação matemática: Relações para o ensino de computação na educação básica. In: XX Workshop sobre Educação em Computação, Curitiba. Anais do XXXII CSBC p. 23. 2012.
- BARDIN, L. **Análise de Conteúdo**. São Paulo: Edições 70, 2011.
- BARROS, L., Ribeiro, S. P. S., Oeiras, J.. “**Projeto de Extensão Universitária para apoio e realização da Olimpíada Brasileira de Informática em Escolas**”. In: XXIX Congresso da SBC - XVII WEI, Bento Gonçalves. 2009
- BBC Learning, B. *What is computational thinking?* Disponível em: <<http://www.bbc.co.uk/education/guides/zp92mp3/revision>> . Acesso em 11 de janeiro de 2020.
- BELL, T., WHITTEN, I., FELLOWS, M. *Computer Science Unplugged*. Universidade de Canterbury, Nova Zelândia. 2007 Disponível em <http://csunplugged.org>
- BELL, T., WITTEN, I. H., FELLOWS, M.. *Computer science unplugged*– Ensinando ciência da computação sem o uso do computador. Tradução coordenada por Luciano Porto Barreto, 2011. Páginas 3–45.
- BLIKSTEIN, P. **O pensamento computacional e a reinvenção do computador na educação**. **Education & Courses**, 2008. Disponível em: <http://www.blikstein.com/paulo/documents/online/>
- BOGDAN, R. C.; BIKLEN, S. K. *Investigação qualitativa em educação: uma introdução à teoria e aos métodos*. Portugal: Porto Editora, 1994
- BORBA, M. C.; DA SILVA, R. S. R.; GADANIDIS, G. **Fases das tecnologias digitais em educação matemática**. Belo Horizonte: Autêntica, 2016.
- BRACKMANN, C. P. **Desenvolvimento do Pensamento Computacional Através de Atividades Desplugadas na Educação Básica**, 2017. 204f. Tese de Doutorado. Programa de Pós-Graduação em Informática na Educação. Universidade Federal do Rio Grande do Sul, Porto Alegre 2017.
- BRASIL. Ministério da Educação. Secretaria da Educação Básica. **Base Nacional Comum Curricular**. Brasília, DF, 2017. Disponível em <http://basenacionalcomum.mec.gov.br/> Acesso em 11 de abril de 2020.
- BRENNAN, K.; RESNICK, M. *New frameworks for studying and assessing the development of computational thinking*. In: **Proceedings of the 2012 annual meeting of the American educational research association**.

Vancouver, Canada. 2012. p. 25.

BUNDY, A. *Computational thinking is pervasive*. *Journal of Scientific and Practical Computing*, 1(2), 67-69, 2007.

BUSARELLO, R. I., ULBRICHT, V. R. E FADEL, L. M. **A gamificação e a sistemática de jogo: conceitos sobre gamificação como recurso motivacional**. In: Fadel, L. M. *et al.* (Org.). *Gamificação na Educação* (pp. 11-37). São Paulo, Pimenta Cultural, 2014.

CASTRO, A. de *et al.* **O uso da programação Scratch para o desenvolvimento de habilidades em crianças do ensino fundamental**. 2017. Dissertação de Mestrado. Universidade Tecnológica Federal do Paraná.

CASTRO, G. A. M. *et al.* Desafios para o professor de ciências e matemática revelados pelo estudo da BNCC do ensino médio. **Revista Eletrônica de Educação Matemática**, v. 15, n. 2, p. 1-32, 2020.

CAVALCANTE, A.; COSTA, L.; ARAUJO, A. Um estudo de caso sobre competências do pensamento computacional desenvolvidas na programação em blocos no Code. Org. In: **Anais dos Workshops do Congresso Brasileiro de Informática na Educação**. 2016. p. 1117.

CIEB, **Centro de Inovação para a Educação Brasileira. Currículo de Referência em Tecnologia e Educação**. Disponível em: <<http://curriculo.cieb.net.br/>>. Acesso em: 11 de janeiro de 2020.

CODE. 2016. CODE. Disponível em: <https://code.org/>. Acesso em: 20 de set. 2020.

COSTA, T. R., BATISTA, A., MAIA, M., ALMEIDA, L. e FARIAS, A. “Trabalhando Fundamentos de Computação no Nível Fundamental: experiência de licenciandos em Computação da Universidade Federal da Paraíba”. XX Workshop de Educação em Computação (WEI'2012). Curitiba, PR, Brasil.

CSTA – *Computer Science Teachers Association* “*Computational thinking: teacher resources*.”2.ed. Disponível em <<https://www.csteachers.org/>>. Acesso em: 15 de junho de 2020.

CUNHA, F.; NASCIMENTO, C. Ribeiro. Uma abordagem baseada em robótica para ensinar fundamentos da computação na educação básica. In: **Anais do Workshop de Informática na Escola**, v. 25, n. 1, p. 735-743, 2019.

DA SILVA TICON, S. C.; DE ABREU MÓL, A. C.; LEGEY, A. P. Atividades plugadas e desplugadas na educação infantil no desenvolvimento do pensamento computacional. **Dialogia**, n. 40, p. 21751, 2022.

DAL BEN, M. de S.; SANTOS, C. P. Tecendo Espaços e Experiências no Campo da Robótica Educacional. **Vivências**, v. 16, n. 30, p. 173-191, 2020.

DALLA VECCHIA, R. A modelagem matemática e a realidade do mundo cibernético. Tese de Doutorado, Unesp, Rio Claro. 2012.

DE ALMEIDA, M. E. B.; VALENTE, J. A. Integração currículo e tecnologias e a produção de narrativas digitais. **Currículo sem fronteiras**, v. 12, n. 3, p. 57-82, 2012.

FARDO, M. L. A gamificação aplicada em ambientes de aprendizagem. **Renote**, v. 11, n.1, 2013.

FIORENTINI, D.; LORENZATO, S. Investigação em educação Matemática: percursos teóricos e

metodológicos. Campinas: Autores Associados, 2006.

FIQUEIREDO, M. PAZ, T., JUNQUEIRA, E. Gamificação e educação: um estado da arte das pesquisas realizadas no Brasil. *In: Anais dos Workshops do Congresso Brasileiro de Informática na Educação*. 1154, 2015.

FRANÇA, RS d *et al.* **A disseminação do pensamento computacional na educação básica: lições aprendidas com experiências de licenciandos em computação.** *In: Anais do XXII Workshop sobre Educação em Computação (WEI-CSBC)*. sn. 2014.

FRANÇA, S. R., SILVA, W. C. e AMARAL, H. J. C. “Ensino de Ciência da Computação na Educação Básica: Experiências, Desafios e Possibilidades”. XX Workshop de Educação em Computação (WEI’2012). Curitiba, PR, Brasil.

FRANCISCO, D. J. *et al.* **Análise de Conteúdo: como podemos analisar dados no campo da educação e tecnologias.** *In: PIMENTEL, Mariano; SANTOS, Edméa. (Org.) Metodologia de pesquisa científica em Informática na Educação: abordagem qualitativa. (Série Metodologia de Pesquisa em Informática na Educação, v. 3)*. Porto Alegre: SBC, 2021.

FUTSCHEK, G. MOSCHITZ, J.: Learning Algorithmic Thinking with Tangible Objects Eases Transition to Computer Programming. *In: ISSEP 2011: 155- 164. 5th International Conference on Informatics in Schools: Situation, Evolution and Perspectives*, Bratislava, Slovakia, p. 26-29, 2011

GAL-EZER, J.; HAREL, D. *Curriculum and course syllabi for a high-school CS program. Computer Science Education*, v. 9, n. 2, p. 114-147, 1999.

GARCIA, R. E.; CORREIA, R. C. M.; Shimabukuro, M. H. “Ensino de Lógica de Programação e Estruturas de Dados para Alunos do Ensino Médio”, *In: Anais do XXVIII WEI, CSBC’2008*, Belém/PA.

GARLET, D.; BIGOLIN, N. M.; SILVEIRA, S. R. Uma proposta para o ensino de programação de computadores na educação básica. Departamento de Tecnologia da Informação, Universidade Federal de Santa Maria, RS. 2016. Acesso em: 02 de setembro de 2019.

GIL, A. C. **Como elaborar projetos de pesquisa**. 4. ed. São Paulo: Atlas, 2002

GOMES, A.; GOMES, C. R. A. **Classificação dos tipos de pesquisa em Informática na Educação.** *In: JAQUES, P. A.; PIMENTEL, M.; SIQUEIRA, S.; BITTENCOURT, Ig. (Org.) Metodologia de Pesquisa em Informática na Educação: Concepção da Pesquisa*. Porto Alegre: SBC, 2019.

GROVER, S.; PEA, R. *Computational Thinking in K-12: A Review of the State of the Field. Educational Researcher*, 42(1), 38–43, 2013.

GUIMARÃES, P. **Desenvolvimento do Pensamento Computacional no ensino da Matemática com estudantes autistas.** 2022. Tese de Mestrado do Programa de Pós Graduação em Ensino de Ciências Naturais e Matemática. Unicentro. Guarapuava, PR. (no prelo)

KAFAI, Y. B. BURKE, Q. *Computer Programming Goes Back to School. In: Education Week*, set. (2013).



- KAPP, K. *The Gamification of Learning and Instruction: Game-based Methods and Strategies for Training and Education*. Pfeiffer, 2012.
- LIUKAS, L. **Hello Ruby**: adventures in coding. Feiwei & Friends, 2015.
- MANZANO, J. A. N. G.; OLIVEIRA, J. F. de. **Algoritmos**: lógica para desenvolvimento de programação de computadores. São Paulo: Érica, 2012.
- MARTINELLI, S. R. **Nova metodologia para a aplicação e avaliação de práticas de pensamento computacional no ensino fundamental I**. Itapetininga, 2017. 159 f. Trabalho de Conclusão de Curso (Especialização em Informática Aplicada à Educação) - Instituto Federal de Educação, Ciência e Tecnologia de São Paulo, Câmpus Itapetininga, Itapetininga, 2017. Disponível em: <<https://drive.ifsp.edu.br/s/6FN90II8DIDl6>>. Acesso em: dezembro de 2021.
- MARTINS, A. **O que é Robótica**. São Paulo: Editora Brasiliense, 2006
- MEDEIROS, I.; RABELO, H. Relato de Experiência: O Pensamento Computacional por meio da aplicação de Atividades Desplugadas em uma turma de 5º Ano - Natal/RN. **VI Conedu**. 2019.
- MESTRE, P. *et al.* **Pensamento Computacional**: Um estudo empírico sobre as questões de matemática do PISA. *In*: Anais dos Workshops do Congresso Brasileiro de Informática na Educação. 1281, 2015.
- NUNES, D. J. Ciência da Computação na Educação Básica. **Jornal da Ciência**, 2011.
- NUNES, N. B.; KOLOGESKI, A. L. Síntese de iniciativas que envolvem Lógica de Programação para o Ensino-aprendizagem de alunos de Ensino Médio. 2019.
- OLIVEIRA, E. J. S. de. **Pensamento computacional e robótica**: Um estudo sobre habilidades desenvolvidas em oficinas de robótica educacional. 2016, 82f. Trabalho de Conclusão de Curso. Licenciatura em Ciências da Computação. Universidade Federal da Paraíba. 2016.
- P21 (2015) <https://www.battelleforkids.org/networks/p21> acesso em 22/11/2021
- PAPERT, S. **A Máquina das Crianças**: Repensando a Escola na Era da Informática. Porto Alegre: Artmed, 2008.
- PAPERT, S. A. *Mindstorms: Children, computers, and powerful ideas*. Basic books, 2020.
- PARANÁ. Secretaria do Estado da Educação. - **Diretrizes da Educação Profissional: fundamentos políticos e pedagógicos**. 2006, Disponível em [http://www.educadores.diaadia.pr.gov.br/arquivos/File/diretrizes/diretriz\\_educacao\\_profissional.pdf](http://www.educadores.diaadia.pr.gov.br/arquivos/File/diretrizes/diretriz_educacao_profissional.pdf)
- PARANÁ. SEED - Portal Dia a Dia Educação. 2013. Disponível em: <http://www.gestaoescolar.diaadia.pr.gov.br/modules/conteudo/conteudo.php?conteudo=337>
- PEREIRA JUNIOR, J. C. R., RAPKIEWICZ, C. E., DELGADO, C., XEXEO, J. A. M. Ensino de Algoritmos e Programação: Uma Experiência no Nível Médio. *In*: **XIII Workshop sobre Educação em Computação**. WEI'05, São Leopoldo - RS. 2005.
- PEREIRA, L. Escolas Defendem Ensino de Programação a Crianças e Adolescentes. 2013.

Disponível em: <<http://olhardigital.uol.com.br/noticia/-escolas-defendem-ensino-de-programacao-acriancas-e-adolescentes/35075>>

PONTE, J. P. da; QUARESMA, M.; MATA-PEREIRA, J. Como desenvolver o raciocínio matemático na sala de aula? **Educação e Matemática**, n. 156, p. 7-11, 2020.

PONTES, H. P. Desenvolvimento de jogos no processo de aprendizado em algoritmos e programação de computadores. **Proceedings of the XII Simpósio Brasileiro de Games e Entretenimento Digital (SBGames)**. São Paulo, 2013.

**PROGRAMAÊ!**: um guia para construção do pensamento computacional. Fundação Telefônica Vivo; Fundação Lemann. - São Paulo : Fundação Telefônica Vivo; Fundação Lemann, 223, 2018.

PROJETO POLÍTICO PEDAGÓGICO. Colégio Estadual Professor Francisco Carneiro Martins. Guarapuava. 2021.

QUINTANILHA, L. Irresistível robô. Revista ARede, São Paulo, ano 3, n. 34, p.10-17, mar. 2008

RAABE, A. *et al.* Um Instrumento para Diagnóstico do Pensamento Computacional. Anais dos Workshops do Congresso Brasileiro de Informática na Educação, [S.l.], p. 1172. 2017. Disponível em: <<https://www.br-ie.org/pub/index.php/wcbie/article/view/7506/5301>>. Acesso 22 de dezembro de 2019

REICHERT, J. T.; BARONE, D. A. C.; KIRST, M. Pensamento Computacional Na Educação Básica: Análise Com Discentes Do Curso De Licenciatura Em Matemática. *Ensino da Matemática em Debate*, v. 6, n. 3, p. 63-83. 2019.

RESNICK M. *All I really need to know (about creative thinking) I learned (by studying how children learn) in kindergarten. In Proceedings of the 6th ACM SIGCHI USA*. ACM. 2007.

SANCHIS, I. P., MAHFOUD, M. Construtivismo: desdobramentos teóricos e no campo da educação. **Revista Eletrônica de Educação**. 4(1), 18-33, 2010.

SANTOS, C. F. R. A robótica educacional como recurso de mobilização e explicitação de invariantes operatórios na resolução de problemas. 2018. 189f. Tese de Doutorado. Doutorado em Ensino de Ciência e Tecnologia. Universidade Federal Tecnológica do Paraná, Ponta Grossa, 2018.

SANTOS, E. R. dos; *et al.* **Estímulo ao Pensamento Computacional a partir da Computação Desplugada: uma proposta para a Educação Infantil**. *Relatec - Revista Latinoamericana de Tecnologia Educativa*. V.15(3), p. 99-112. 2016.

SENAI, GIZ. Profissões emergentes na era digital: oportunidades e desafios na qualificação profissional para uma recuperação verde, 2021.

SHUTE, V. J. *Who is likely to acquire programming skills? Journal of Educational Computing Research*, 7(1), 1-24, 1991.

SHUTE, V. J., MASDUKI, I., DONMEZ, O. *Conceptual framework for modeling, assessing, and supporting competencies within game environments. Technology, Instruction, Cognition, and Learnin*. 8, 137-161, 2010.

- SHUTE, V. J.; SUN, C.; ASBELL-CLARKE, J. *Demystifying computational thinking*. *Educational Research Review*, v. 22, p. 142-158, 2017.
- SILVA, L. C. L. da. **A relação do Pensamento Computacional com o ensino de Matemática na Educação Básica**. 2019, 131f. Dissertação de mestrado. Programa de Pós-Graduação em Matemática. Universidade Estadual Paulista. 2019.
- SOUSA, R. V. de; BARRETO L. P; ANDRADE, A; ABDALLA, D. Ensinando e aprendendo conceitos sobre a ciência da computação sem o uso do computador: Computação *Unplugged!*. **Práticas em Informática na Educação: Minicursos do Congresso Brasileiro de Informática na Educação**, v. 1, n. 1, 2010.
- VALENTE, J. A. Integração do pensamento computacional no currículo da educação básica: diferentes estratégias usadas e questões de formação de professores e avaliação dos alunos. *Revista E-curriculum*, v. 14, n.3, p. 864-897.
- VICARI, R. M.; MOREIRA, A. F.; MENEZES, P. F. B. Pensamento computacional: revisão bibliográfica. 2018.
- WERBACH, K., HUNTER, D. **For The Win: How Game Thinking Can Revolutionize Your Business**. Filadélfia, Pensilvânia: Wharton Digital Press, 2012.
- 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*. *Communications of the ACM*. March, Vol. 49, No. 13. 2006.
- WING, J. M. *Computational Thinking: What and Why?*, 17. out. 2010. Disponível em: <<http://www.cs.cmu.edu/~CompThink/resources/TheLinkWing.pdf>>.
- ZANETTI, H. *et al.* Proposta de ensino de programação para crianças com Scratch e Pensamento Computacional. **Tecnologias, Sociedade e Conhecimento**, v. 4, n. 1, p. 43- 58. 2017. Acesso em janeiro de 2020.
- ZANETTI, H.; OLIVEIRA, C. Práticas de ensino de Programação de Computadores com Robótica Pedagógica e aplicação de Pensamento Computacional. Anais dos Workshops do Congresso Brasileiro de Informática na Educação, [S.l.], p. 1236, out. 2015. ISSN 2316-8889. Disponível em: <<https://br-ie.org/pub/index.php/wcbie/article/view/6268>>. Acesso em: 16 fev. 2020.
- ZANETTI, H; BORGES, M.; RICARTE, I. Pensamento Computacional no Ensino de Programação: Uma Revisão Sistemática da Literatura Brasileira. **SBIE**, [S.l.], p. 21. 2016. Disponível em: <<https://www.br-ie.org/pub/index.php/sbie/article/view/6677/4566>>. Acesso em 04 de janeiro de 2020.
- ZICHERMANN, G.; CUNNINGHAM, C. **Gamification by Design. Implementing Game Mechanics in Web and Mobile Apps**. Canada: O'Reilly Media, 2011.